Quality Assessment of Generative Adversarial Networks

Project Report

Gaganjot Singh (20133821)

Department of Electrical and computer engineering

Queen's University

This report is submitted for the degree of

Master of Engineering

Abstract

Generative Adversarial Networks (GANs) have been considered as one of the promising approaches in deep learning, it generate fake images with the same statistics as the training set. Although different GANs have been proposed for different image analysis tasks, including image generation, semi-supervised learning, and image-to-image translation, their quality assessment has remained challenging and so there is no way to compare their visual performance. In this context, this report conducts the experiments using six different GANs on six different databases to generate fake images. The generated images are then assessed using several no-reference image quality metrics to compare the performance of the GANs in generating fake images. The experimental results also reveal interesting properties of the examined GAN, notably regarding the different datasets and metric functions.

Acknowledgement

I would like to express my deep and sincere gratitude to my research supervisor, Ali Etemad, Assistant Professor at the Department of Electrical and Computer Engineering, Queens University (Kingston, Canada), and director of the Ambient Intelligence and Interaction (Amii) lab, for allowing me to do research and provide invaluable guidance throughout this research. His vision, sincerity, and motivation have deeply motivated me.

It is my absolute honor to record to Alireza Sepas-Moghaddam, my best praises and deepest appreciation for his wise and precious guidance, both of whom have been instrumental in my research toward growth as an engineer. It was a great privilege and honor to work and study under their supervision. I am extremely thankful for what they offered me.

Gaganjot singh

Contents

List of Figures

1 Introduction

1.1 Background

In recent years, generative adversarial networks (GANs) [1] have seen exponential growth. In addition to offering amazingly plausible fake images [2], they were also performed innovatively in, for instance, semi-supervised learning, image-to-image translation [3], and image refinement simulated. Regardless of how many GAN models are available, to find the better GAN we need to evaluate generated images. However, their evaluation is still predominantly subjective and frequently reflects the manual examination of the visual fidelity of generated images

Image quality is a critical index for estimating the efficiency of the video and image processing schemes. Therefore, the assessment of image quality is of great importance. The easiest way for an image to be assessed is to demonstrate it to a skilled human observer. However, for each person, human perceptions may differ. This issue can be addressed by taking various opinions from distinct people and processing the outcomes statistically; this is called *subjective quality assessment*. However, this is a long and inaccurate quality assessment procedure. In the same way that participants were selected, all their information, skills, accessibility, bias, interpretations are subjective and qualitative. Due to the intrinsic constraints of subjective assessments, the development of GANs needs appropriate quantitative measures to regulate the structure of better models.

Contrary to many image analysis tasks, such as image compression, where both reference and reconstructed images are available, the images generated by the GANs may not have any reference image. Therefore, to find the better GAN one needs an automated system for no-reference quantitative evaluation of GANs images. The problem can thus be represented as *objective quality assessment of GAN*.

1.2 Objective

The main objective of this report is on objective quality evaluation of image generated by GANs using Image Quality Assessment (IQA). The goal is to find the best performing GAN using No-Reference Image Quality Assessment metric(NR-IQA). The major segment of this project entails (i) reviewing and categorizing the available GANs; (ii)choosing the best GANs for generating fake images; (iii) using IQA for measuring quality of the generated images; and finally (iv) comparing various IQA for finding best performing GAN. The outcome of this report assess which GAN produces better images using NR-IQA.

1.3 Scientific Summary

As the goal of the project is to find the GAN that can generate best fake images and this is done by quality evaluation of these images by NR-IQA metrics. In the project, 6 databases were used by 6 GANs to generate images and then 3 no-reference image quality assessment metrics were used to evaluate the images this can be seen in figure 1.
The 6 database includes:

The 6 database includes-

- MNIST: A digit database including a training set of 60,000 examples and a test set with 10,000 examples.[17]

- CelebA: Large-scale facial data collection of over 200 K famous images, each with 40 characteristics. [18]

- Stl10: An image recognition dataset with 10 class and 10000 unlabeled images [19]. 10 classes include airplane, bird, car, cat, deer, dog, horse, monkey, ship, and truck.

- Lsun: It contains around 1 million labeled images for each of 10 scene categories and 20 object categories. [20] The dataset contains bike, bird, bedroom, bus, chair, car, table images, among many.

- Maps: A dataset consisting of 1000 images of land in both satellite and normal views.
- Facade: It include a data set of facade images from 606 rectified images of facades from different sources. [22]

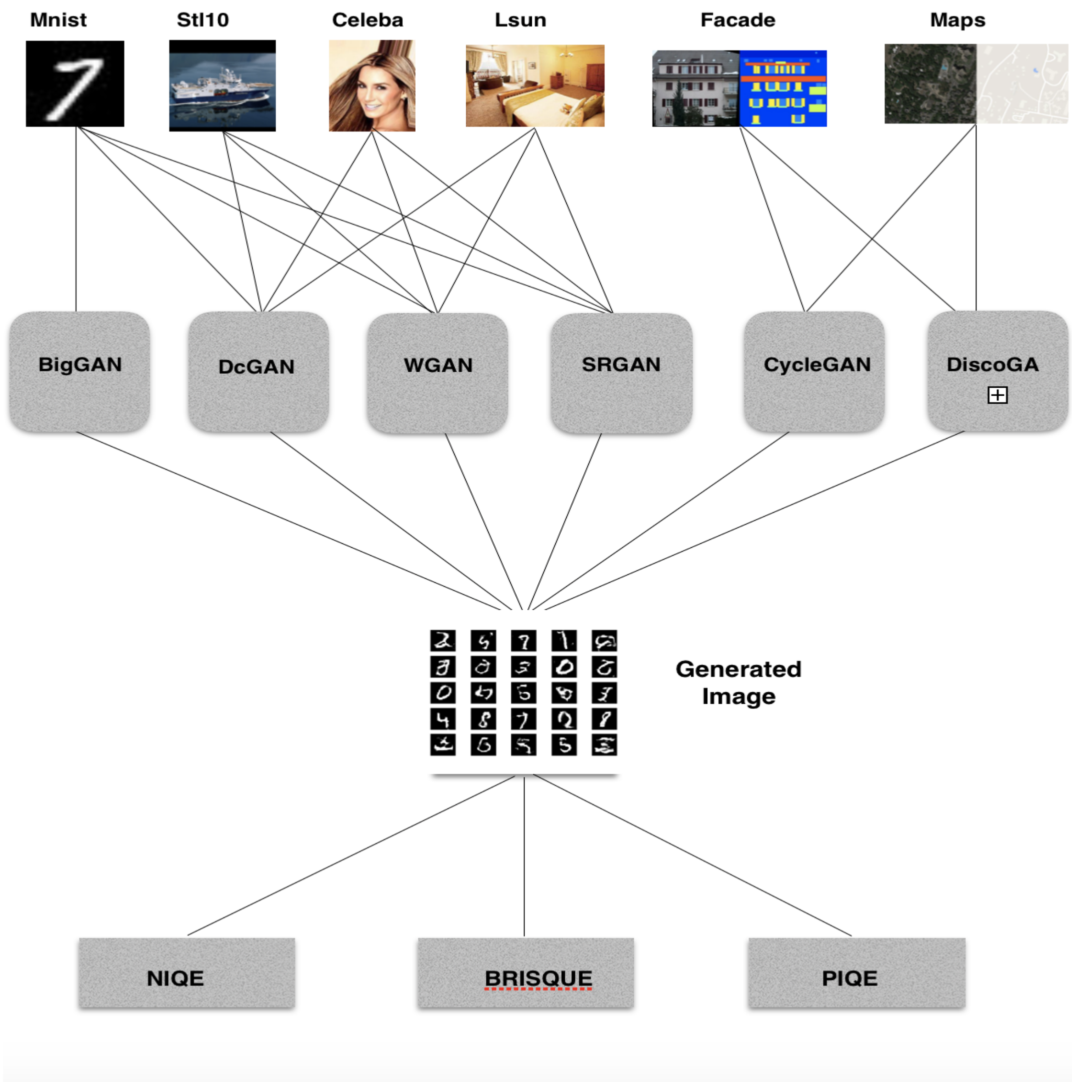


Figure 1: Summary of the projects contributions

Six GANs that used these datasets are (more details will be provided in Section 2.2) :

- DcGAN - Deep Convolutional GAN [21] as name suggest uses convolutional layer at place of fully connected and max-pooling layer.

- WGAN - Wasserstein GAN [23] uses Wasserstein distance to estimate the variance between the model and target distributions.
- BigGAN -[27] It uses deep neural network with very large batch size and high number of model parameters to produce high resolution images.
- SrGAN - Super Resolution GAN [26]uses deep network with combination of adversary network to produce super-resolved images.
- DiscoGAN -Discovery GAN [25] performs cross-domain image to image translation using deep convolutional network.
- CycleGAN - Cycleconsistent GAN [24] performs unpaired image-to-image translation using cycle-consistent adversarial networks.

IQA has a broad range of applications [4][5][6] and could be arranged into three categories: Full-Reference IQA (FR-IQA) (FR-IQA)[7][8][9], No- Reference IQA (NR- IQA)[11][12] [13][14][15] And Reduced- Reference IQA (RR-IQA) [10]. Further, three no-reference image quality assessment metrics used in this research are:

- PIQE - Perception based Image Quality Evaluator (PIQE) [29] is opinion-unaware and unsupervised metric. PIQE measures the local variability of the perceptibly distorted blocks and assesses block distortion.
- NIQE - Natural Image Quality Evaluator (NIQE) [28] model has been trained on a primitive image database, measuring the quality of images with fading distortion.
- BRISQUE- Blind/Reference less Image Spatial Quality Evaluator (BRISQUE) [30] is trained on an image database with distinct distortions, measuring the quality of images with the same distortion type.

2 Methodology

2.1 Theory

2.1.1 GAN

Deep learning is used to discover rich, progressive models that show probability distribution over data such as natural language processing, speech in the form of waveforms, and natural images that are mainly experienced in artificial intelligence. In 2014, Ian Goodfellow and his peers [1] suggested, a deep-learning-based generative model. Generative modeling is an unsupervised learning task in machine learning, involving the automated detection and learning of the regularities or patterns of the input data, so that new examples can be generated or produced that can be plausibly taken from the original dataset. GANs are an intelligent method of training a generative model by dividing the problem with two sub models as a supervised learning problem: the generator model that we train to produce new examples, and the discriminator model which attempts to classify the instance as either real (from the domain) or fake (generated).

Primarily, the generative model in GAN is put against an enemy: a discriminatory model, which decides whether an instance comes from data sample or models. The basic idea of GANs is to establish a game among two players. One is called the generator. The generator produces the samples which are intended to be comparable to the training data. A discriminator is the other player. The discriminator inspects tests to determine whether they are real or fake. The discriminator has the desire to use traditional controlled methods of learning, to isolate contributions into two classes (real or fake). The generator is willing to deceive the discriminator. The generative model can be considered as like a group of forgers, try to provide and use counterfeit funds without identification, while the discrimination model is almost like the police, seeking to recognize the counterfeit cash. In this match, rivalry leads both groups to enhance their strategies until the fakes are indistinguishable from the certified data.

- **The Generator** - The generator is essentially a differentiable function *G* represented by multilayer perceptron with parameter *g*. For the start, input noise *z* is tested from some basic prior distribution,*G(z)* produces an output *x*. The input of the first layer of the deep neural net does not need to be same as the input to function *G*.

- **The Discriminator** - There is also a second multilayer perceptron *D* which gives a single scalar output. *D(x)* represents the probability that x comes from the generator or the data.

- **Training Process** - In training, both discriminator *D* and Generator *G* play minimax game in two-player with value function *V(G, D)*. To maximize the accurate label to both samples generated from *G* and training examples we train *D*, simultaneously we train *G* to decrease *log(1-D(G(z)))*.

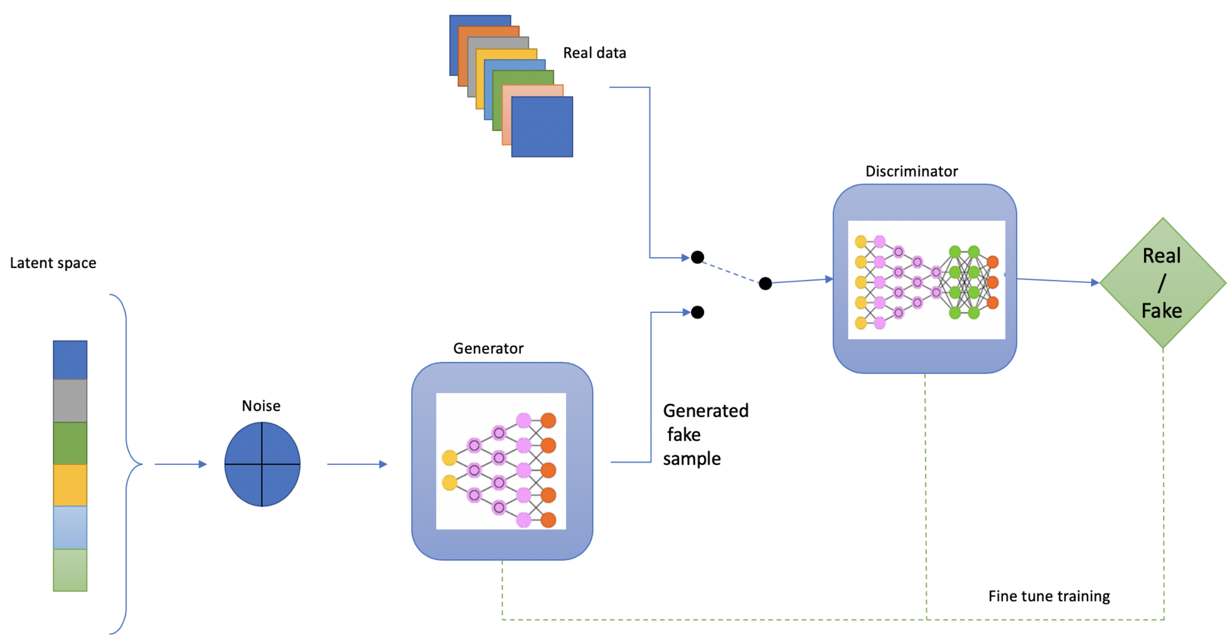


Figure 2: Gan Working

At the beginning of the learning phase, *G* is poor so *D* is very sure about rejecting the samples produced, as they clearly are not as good as the training data. Therefore, we can increase *log*

D(G(z) because of the *log(1-D(G(z))* at the training location G, in order to reduce *log(1-D(G(z)))*. This feature provides the same set objective of *G* and *D* dynamics but makes the beginning of the learning much stronger.

$$\min_{G}\max_{D} V(D,G) = E_{x\sim p_{data}(x)}[logD(x)] + E_{z\sim p_z(z)}[\log(1 - D(G)z)))] \tag{1}$$

where *D* is Discriminator, *G* is generator, *z* is input noise and *x* is output of the generator. Even though GANs have shown some real achievements in realistic image generation, but the training is unstable and slow. Many GAN models suffer the following major problems:

1. **Nash equilibrium** - GAN is a non-cooperative zero-sum game. One models try to win my maximizing its own actions and another model action is to reduce them. In game theory, the GAN model converges when the discriminator and the generator reach a Nash equilibrium. Nash equilibrium is when one model will not change regardless of what another model does.

2. **Vanishing gradient** - If the discriminator behavior is bad, the generator has no precise feedback or reality, and the loss function cannot represent it. If the discriminator operates admirably, the loss function rate decreases to close to zero and the learning is too mild or even stuck.

3. **Mode Collapse** - The generator can fall into a position during the training, in which it usually produces the same outputs. For GANs, generally referred to as Mode Collapse, this is a typical disappointed situation. Although the generator may most probably mislead the related discriminator, it does not understand how the complex information from the true globe are represented, as they stand in a small area with a very low variety.

4. **No proper evaluation metric** - Without a decent assessment metric, it resembles working in obscurity. GAN do not provide good sign to advise when to stop and when to look at the performance of various models.

Several methods to improve GANs have been discussed in [60]. Some of them are mentioned below:

1. **Feature matching** - A new objective for a generator to tackle the instability of GAN's is to avoid overtrainings of the existing discriminator function. Feature matching offers the objective of optimizing the discriminator to check whether the output equivalents of the generator require initial sample statistics. The fresh generator loss feature i

$$E_{x\sim p_{data}} f(x) - E_{z\sim p_z(z)} f(G(z))||^2 \tag{2}$$

   where *f(x)* denote activations on an intermediate layer of the discriminator,*z* is input noise and *x* is output of the generator..

2. **Minibatch discriminator** - with mini batch discriminator instead of processing each point independently the discrimination can analyse the correlation between training data points in line batch. The concept of discrimination by the minibatch is very common: any model of discrimination that examines the different patterns in sequence and not in solitude could likely assist to prevent the generator's collapse. The discrimination in minibatches allows us to generate very quickly visually attractive examples, and this view is preferred to include matching. In one mini batch, the overall summary of one data point is comprehended by seeing how near it is to another samples in same batch.

3. **Historical Averaging** - When implementing this procedure, we transform each players cost to include a term

$$||\theta = \frac{1`}{t} \sum_{i=1}^{t} \theta[i]||^2 \tag{3}$$

   where $\theta[i]$ is the value of the parameters at past time *i* and *t* are the number of models

4. **One-sided label smoothing** - Label smoothing, substitutes the 0 and 1 targets for a classifier with softened values, like .9 or .1, and was lately conferred to lessen the vulnerability of neural networks to adversarial example

5. **Virtual batch normalization** - The reference batch is determined once at the origin and stays identical within the training.Each data sample is normalized based on an established batch (reference batch) of data preferably than within its minibatch.

With the improvements in GANs, the number of GANs have increase exponential and can be classified into 4 major categories-

1. **Single/multiple class generation**- They are the type of generative adversarial networks that take in noise and generates the images from them with in single or multiple domains

2. **Intra-domain** - the type of GANs that translate images within a domain.

3. **Inter-domain** - the type of GANs that translate images from one domain to another domain.

4. **Super-resolution** - There are some GANs that perform super-resolution.

2.1.2 No-reference image quality metrics

Estimation of noise content of an image and its subsequent removal is a very important area of research. Human intelligence has until these years been considered as the only instrument to sensor signal-image noise. The simplest way to evaluate the quality of an image is to show it to an expert human observer. However, for each person, human understanding may be distinct. This issue can be addressed by taking various opinions and statistically processing the outcomes from distinct people. This is known as a subjective evaluation of image quality. However, it is a long and inaccurate quality assessment method. In fact, everything is subjective and quality from the choice of the participants, their knowledge, expertise, accessibility, seriousness prejudice, interpretations. An automated system is therefore needed for quantitative image assessment. A simplified and effective objective assessment scheme can be used, giving quality ratings to images in accordance with a subjective human quality assessment. The objective Image Quality Assessment (IQA) refers to the challenging task of automatically predicting the perceptual quality of a deformed image. Objective image quality assessment can be accomplished in three ways-

1. **Full reference image quality assessment (FR-IQA)** - This is the evaluation of image by comparing the quality of the distorted image with the initial version of the image, which is thought to be undistorted. By measuring the difference of distorted image from the reference image, the amount of distortion is calculated. The easiest way of measuring image quality is to calculate the Peak Signal to Noise Ratio(PSNR), but PSNR doesnt always correlate with human visual perception and image quality [31]. Additional parameters were suggested for addressing the restriction of PSNR metric. Structural Similarity Index (SSIM) [32], Visual Fidelity (VIF) [39], Fast SSIM (FSIM) [33], Multi-Scale Structural Similarity (M-SSIM), Weighted four-component Structural Similarity [35] are parameters which correlate well with human perception. These parameters give the extent of deviation of a distorted image from the reference image. The requirement for a quality assessment reference image limits the use of the following parameters and algorithms for quality assessment.

2. **Reduced reference image quality assessment (RR-IQA)** - In RR-IQA Instead of a full image, algorithms only use restricted characteristics from reference images to assess the quality. The restrictions of FR-IQA remain in RR-IQA, meaning that quality assessment is required with characteristics obtained from the reference image. Despite all the constraints of the RR-IQA technology, satellite and remotely sensed image quality assessment is commonly used.

3. **No reference image quality assessment (NR-IQA)** -These algorithms gives image quality without the need for a reference or its characteristics which is the main reason of using NR-IQA for evaluating image generated by GANs. The NR-IQA issue is more severe than the two issues mentioned above. Due to the lack of a reference image it is necessary to model the reference image statistics, the nature of the human visual system and the impact of distortions

in unmonitored image statistics. The efficacy of a measure of quality with a certain distorted image in the lack of a reference image can also be assessed very difficultly. No-reference algorithms use statistical features of the input image to evaluate the image quality. Since there is no reference image for the fake images that produced by the GANs no-reference quality metrics are the only available options. These no-reference algorithms include:

- **Blind/Reference less Image Spatial Quality Evaluator (BRISQUE)** - Mittal et al. [49] suggested image quality metrics in special domain. This algorithm utilizes locally normalized luminance [30], that is, Mean Subtracted Normalized Contrast (MSCN) image and is calculated as shown in (9).

$$I'(i,j) = \frac{I(i,j)-\mu(i,j)}{\sigma(i,j)+1} \tag{4}$$

  where *'μ'* and *'σ'* are the mean and standard deviation in a 3x3 window, *I* is the Intensity image and $i \in 1,2M$, $j \in 1,2N$ are spatial indices. The approach is based on the principle that natural images possess certain regular statistical properties that are measurably modified by the presence of distortions. It utilizes a Natural Scene Statistics. (NSS) model framework of locally normalized luminance coefficients and quantifies 'naturalness' using the parameters of the model. Despite its simplicity, BRISQUE is statistically better than the full-reference peak signal-to-noise ratio and the structural similarity index, and is highly competitive with respect to all present-day distortion-generic NR-IQA algorithms. BRISQUE has very low computational complexity, making it well suited for real time applications.

- **Natural Image Quality Evaluator (NIQE)** is based on the construction of a quality aware collection of statistical features based on a simple and successful space domain natural scene statistic (NSS) model. Although a NIQE system is trained on an image database, NIQE can evaluate the image quality arbitrarily. NIQE is unconscious of opinion and uses no subjective quality ratings. The difference is, the NIQE image score may not correlate with the human sense of quality as well as the BRISQUE score. IQA model is founded on perceptually relevant spatial domain NSS features extracted from local image patches that effectively capture the essential low-order statistics of natural images. The NSS features used in the NIQE index are similar to those used in a prior IQA model called BRISQUE. However, NIQE only uses the NSS features from a corpus of natural images while BRISQUE is trained on features obtained from both natural and distorted images and also on human judgments of the quality of these images. Once the image coefficients are computed, the image is partitioned into *PxP* patches. Specific NSS features are then computed from the coefficients of each patch.Given a collection of natural image patches selected as above, their statistics are characterized by quality aware NSS features computed from each selected patch. A simple model of the NSS features computed from natural image patches can be obtained by fitting them with a Multivariate gradient (MVG) density. The new IQA index, called NIQE, is applied by computing the 36 identical NSS features from patches of the same size *PxP* from the image to be quality analyzed, fitting them with the MVG model, then comparing its MVG fit to the natural MVG model.

- **Perception based Image Quality Evaluator (PIQE)** The PIQE algorithm is opinion-unaware and unsupervised, which means it does not require a trained model. PIQE is capable of measuring arbitrary image quality and, in most cases, is NIQE like. In order to calculate quality, PIQE assesses block distortion and measures the local variability of the perceptibly distorted blocks. This method is inspired from two principles about how humans perceive image quality. Firstly, humans visual attention is highly directed towards salient points in an image or spatially active regions. Secondly, local quality at block/patch level adds up to the overall quality of an image as perceived by humans. The first pre-processing step that is performed on the input test image is local mean removal and divisive normalization. This step extracts Natural Scene Statistics (NSS) features from the grayscale image. In the method, spatially active blocks, are analyzed for two types of distortion criteria, namely, noticeable distortion criterion and additive white noise criterion. A distorted block is then assigned a distortion score based on the type of distortion.

2.2 Related Work

2.2.1 GAN

Since the invention of GAN, the amount of GAN that have come up is incredible and the GAN Zoo keeps on expanding. Now, there are more than 400 GANs with different feature available. Some of the most popular GANs are

1. **DcGAN (Deep convolutional generative adversarial network)**-1. DCGAN is one of the successful and popular network design for GAN. It is a direct extension of the GAN described in Section 2.1.1, except it uses convolutional-transpose and convolutional layer in the generator and discriminator, respectively. It was first defined by Radford et. al. [21]. The discriminator, D, is made up of batch norm layers, strided convolutional layers, and LeakyRelu activations. The input of discriminator is 3x64x64 input image and output is a scalar probability that input is from the real data distribution. The generator, G, is designed to map a latent vector, z, which is drawn from a standard normal distribution and output is 3x64x64 RGB image. This is accomplished through a series of strided convolutional transpose layer, each paired with 2D batch norm layer and a ReLU activation which can be seen in figure 3.

   It has also been mentioned that using strided convolutional layer is a better practice than using pooling to downsample, as it lets the network to learn its own pooling function. Also, healthier gradient flow is promoted my batch norm and leaky ReLU which is critical for the learning process of both G and D. The author specify that all model weight should be randomly initialized from a normal distribution with stdev=0.02, mean =0. All models were trained with mini-batch stochastic gradient descent (SGD) with a mini-batch size of 128. In the leakyReLU, the slope of the leak was set to 0.2 in all models

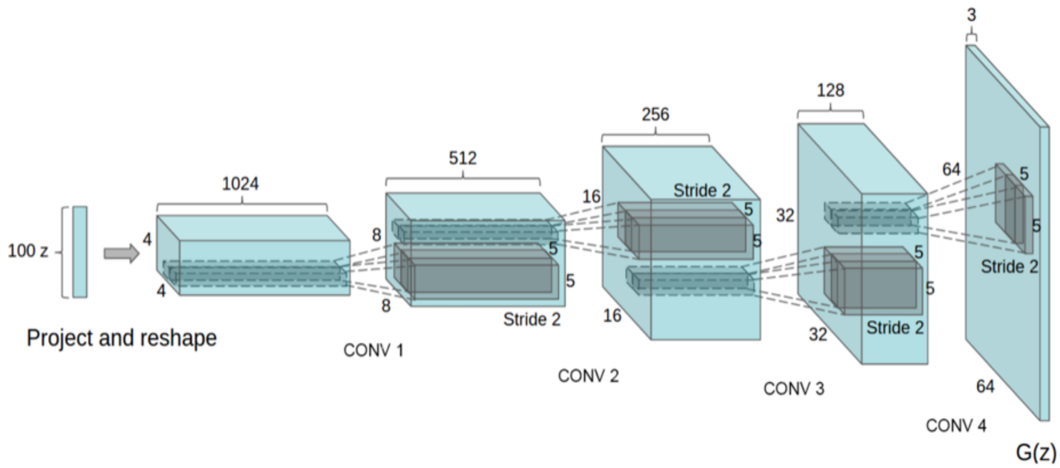


Figure 3: DcGAN architecture.

2. **WGAN (Wasserstein GAN)-** WGAN is a GAN variant which uses the 1-Wasserstein distance, introduced by Martin Arjovsky, et al [21]. WGAN proposes a new cost function using Wasserstein distance that has a smoother gradient. It changes or replaces the discriminator mode with the critic that scores the realness or fakeness of a given image instead of classifying or predicting the probability of generated images as real or fake. The development of the WGAN has a dense mathematical motivation, although in practice requires only a few minor modifications to the established standard deep convolutional generative adversarial network(DCGAN). The authors proposed a smart transformation of the formula based on Kantorovich-Rubinstein duality [24], according to equation (5)

$$L(p_r, p_g) = \min_{w \in W} E_{x \sim p_r}[f_w(x)] - E_{z \sim p_r(z)}[f_w(g_\theta(z))] \tag{5}$$

where z is input noise, x is output generated. In the modified Wasserstein-GAN, the discriminator model is used to learn w to find a good fw and the loss function is configured as measuring the Wasserstein distance between Pr and Pg. The discriminator is trained to learn k-Lipschitz continuous function to help compute Wasserstein distance instead of labeling the fake samples apart from the real ones. Wasserstein distance gets smaller as loss function decrease in the training and the generator mode's output grows closer to real data distribution. Figure 4 shows the architecture of WGAN.

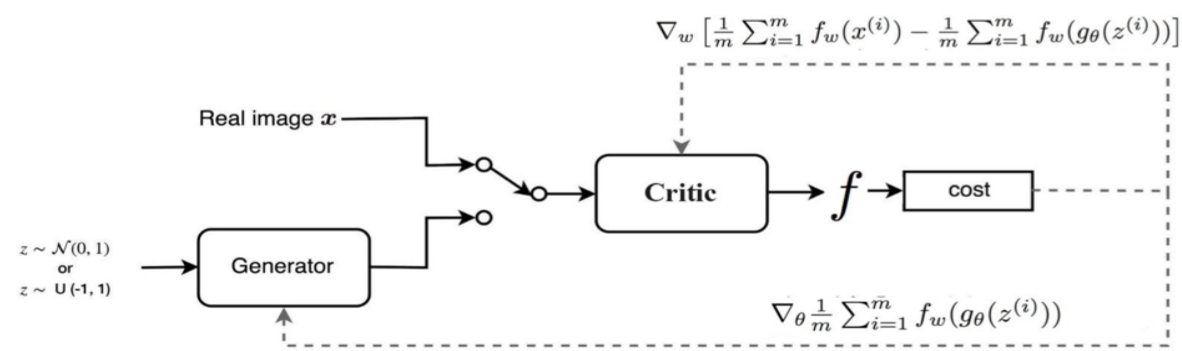


Figure 4: WGAN architecture.

3. **SrGAN (Super Resolution GAN)** - SrGAN produces higher resolution images by applying a deep network in combination with an adversary network. Training begins with down-sampling high-resolution image to low-resolution image. The low-resolution images are passed through a generator to perform up-sampling. High-resolution images are distinguished by discriminator and the GAN loss is back-propagated to train discriminator and generator. The network design is mostly composed of convolution layers, batch normalization and parameterized ReLU. The convolutional layer stands for 3x3 kernel filters outputting 64 channels with stride 1, as shown in figure 5. Generator loss function composes of adversarial loss and content loss. To keep perceptual similarity instead of pixel-wise similarity content loss is used. The adversarial loss pushes the solution to natural image manifold using discriminator network.

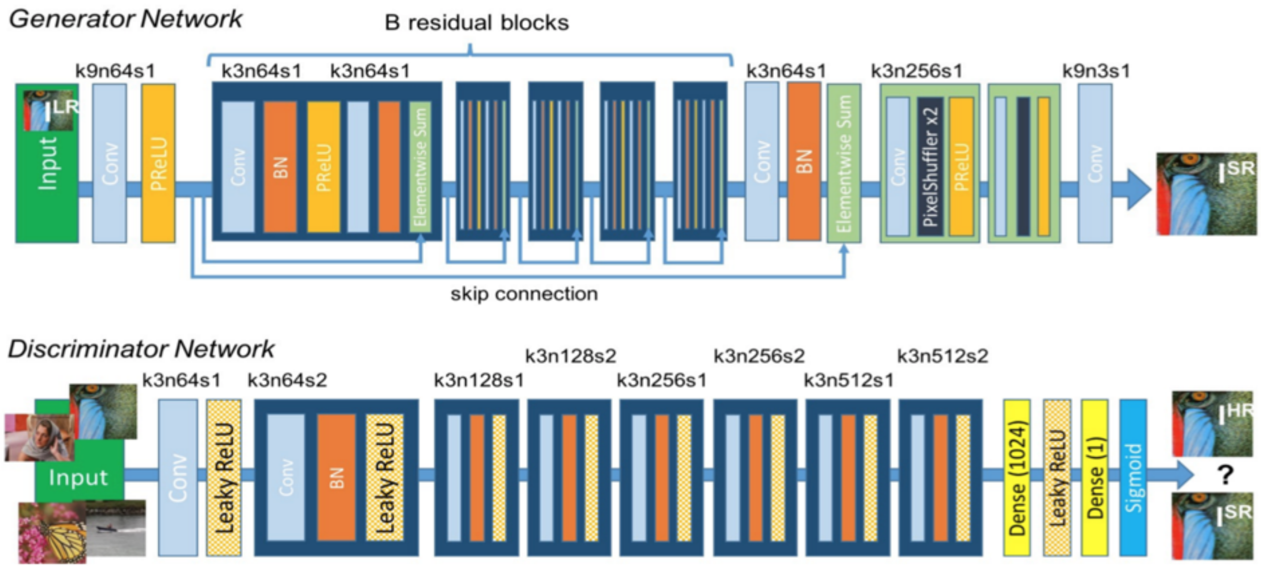


Figure 5: SrGAN architecture.

4. **CycleGAN** - It is an approach to train a deep convolutional neural network for image-to-image translation tasks. The image-to-image translation is the task of transforming an image from one domain to another. Datasets for training model for image-to-image translation requires large paired examples which can be difficult to get. However, CycleGAN is a technique that uses unpaired collections of images from two different domains for training unsupervised image translation models via the GAN architecture.

   CycleGAN architecture involves simultaneous training of two discriminator models and two generator models. The input of the first domain is taken as input by one generator and generated output is of other domain, and the other generator takes the second domain as input and produces images form the first domain. Discriminator models are then used to determine how plausible the generated images are and update the generator models accordingly. The discriminator and generator models are trained in an adversarial zero-sum process

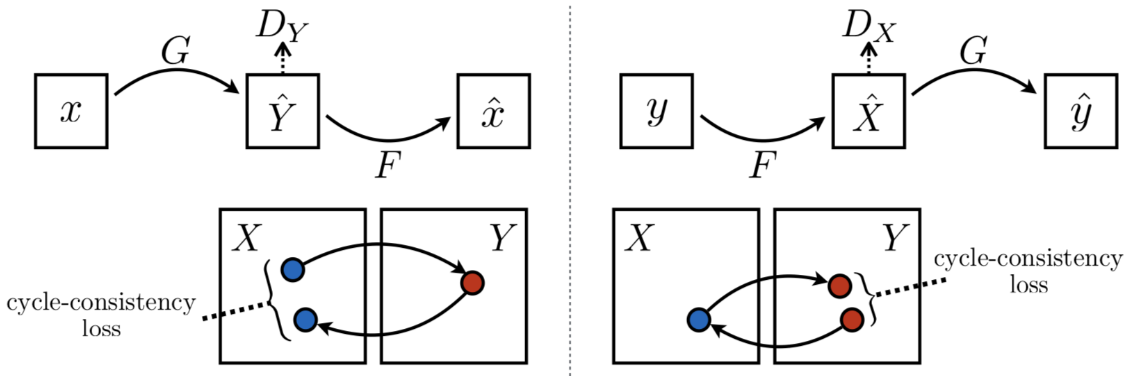


Figure 6: CycleGAN architecture.

Additionally, an extension is used in CycleGAN called cycle consistency. This is a technique to use the output of one generator as the input of other generator and then comparing the image generated with the original input of the first generator. This can be seen above in figure 6. It enforces that *F(G(x)) = x* and *G(F(y)) = y*.

$$Loss_{cyc}(G,F,X,Y) = \frac{1}{m}\sum_{i=1}^{m}\left[F\left(G(x_i)\right) - x_i\right] + \left[G\left(F(y_i)\right) - y_i\right] \quad (7)$$

where *G* tries to generate images *G(x)* that looks similar to images from domain *y*, while Dy aims to distinguish between translated samples G(x) and real samples *y*. *G* aims to minimize this objective against an adversary *D* that tries to maximize it.

5. **DiscoveryGAN** - DiscoGAN generates images of products in one domain when given an image of another domain. The architecture of DiscoGAN is similar to that of CycleGAN, it also has two generators and two discriminators. Generator and discriminator perform the same function as performed by CycleGAN. There is just one change in CycleGAN, it has an additional feature that contributes to cycle consistency loss in the overall loss function.

   A generator G_{AB} translates input imageX_A from domain *A* into X_{AB} in domain *B* as shown in figure 7. The generated image is then translated into a domain *A* image X_{ABA} to match the original input an image (Equation 8, 9). The generator G_{AB} receives two types of losses a reconstruction loss L_{CONSTA} (Equation 10) that measures how well the original input is reconstructed after a sequence of two generations, and a standard GAN generator loss L_{GAN_B} (Equation 11) that measures how realistic the generated images is in domain B.

$$x_{AB} = \mathbf{G}_{AB}(x_A) \tag{8}$$
$$x_{ABA} = \mathbf{G}_{BA}(x_{AB}) = \mathbf{G}_{BA} \circ \mathbf{G}_{AB}(x_A) \tag{9}$$
$$L_{CONST_A} = d(\mathbf{G}_{BA} \circ \mathbf{G}_{AB}(x_A), x_A) \tag{10}$$
$$L_{GAN_B} = -\mathbb{E}_{x_A \sim P_A}\left[\log \mathbf{D}_B(\mathbf{G}_{AB}(x_A))\right] \tag{11}$$

$$L_{G_{AB}} = L_{GAN_B} + L_{CONST_A} \tag{12}$$

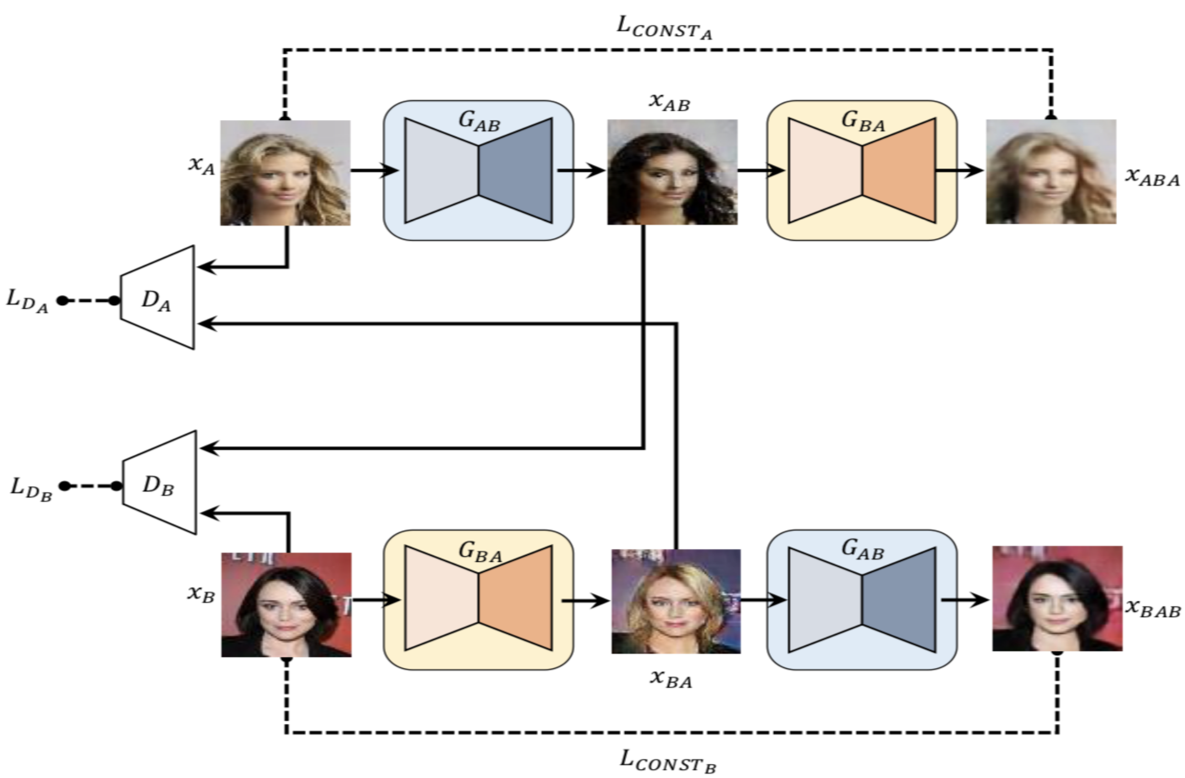


Figure 7: DiscoGAN architecture.

6. **BigGAN** - BigGAN is an approach to brings together a series of recent best practices in the training GAN's and increases the batch size and number of model parameters. The result is the generation of high-resolution images (large) and high-quality images (high-fidelity).
   It uses model architecture with attention modules from Self-attention GAN[SAGAN] [62], as can be seen in figure 8 and is trained via hinge loss. This involves introducing an attention map that is applied to feature maps, allowing the generator and discriminator models to focus on different parts of the image. Its architecture also introduces a *"truncation trick"* that results in an improvement in image quality.

$z \in \mathbb{R}^{128} \sim \mathcal{N}(0, I)$
dense, $4 \times 4 \times 16 \cdot ch$
ResBlock up $16 \cdot ch$
ResBlock up $8 \cdot ch$
ResBlock up $8 \cdot ch$
ResBlock up $4 \cdot ch$
Non-Local Block (64×64)
ResBlock up $2 \cdot ch$
ResBlock up $1 \cdot ch$
ResBlock up $1 \cdot ch$
BN, ReLU, 3×3 conv 3
Tanh

(a) Generator

RGB image $x \in \mathbb{R}^{512 \times 512 \times 3}$
ResBlock down $1 \cdot ch$
ResBlock down $1 \cdot ch$
ResBlock down $2 \cdot ch$
Non-Local Block (64×64)
ResBlock down $4 \cdot ch$
ResBlock down $8 \cdot ch$
ResBlock down $8 \cdot ch$
ResBlock down $16 \cdot ch$
ResBlock $16 \cdot ch$
ReLU
Global sum pooling
Embed(y)$\cdot \boldsymbol{h}$ + (dense $\rightarrow$ 1)

(b) Discriminator

Figure 8: BigGAN architecture.

Class-conditional batch normalization is used to provide class information to the generator. It transforms a layer's activation *x* into normalized activation *z*. In the paper, the technique is referred to as *conditional instance normalization*. Special normalization was used to normalize weight in the generator. Specifically, it involves normalizing the spectral norm of the weight matrix. BigGAN discriminator model is updated twice before updating generator. Orthogonal initialization is used to initialize the model weight. This involves setting the weights to be a random orthogonal matrix. The number of model parameter and batch size was dramatically increased. To directly connect the input latent point to specific layers deep in the network skip connections were added.

2.2.2 No-reference image quality metrics

Evaluation of the GANs has been one of the major concern for many years. Therefore, a lot of efforts has been done to find the best method to evaluate the images. In 2018, Ali Borji reviews more than 24 quantitative and 5 qualitative measures for evaluating generative models with a particular emphasis on GAN derived models. The paper discussed image quality measures have a high probability for finding Perceptual Judgments, Sensitivity to Distortions and Sample Efficiency. Vipin kambal [47] surveyed all the NR-IQA available and used that on the dataset that already had subjective evaluation done. However, these metrics have never been tested on the images generated by GANs.

Therefore, in our approach, we have decided to use no-reference image quality assessment on different types of GANs to find the which GAN produces best images.

2.3 Approach

In this subsection, we discuss the procedure followed for evaluation of the images generated by GANs using No-reference Quality Metric. The project can be divided into 5 steps:

1. **Reviewing available GANs**: First and foremost, we start by reviewing the most important GANs that have gained most popularity in last 5 years. Twenty-seven GANs were reviewed based on their type, dataset used and year they were published

GAN Name	Year	Dataset	Task
Auxiliary classifier GAN [51]	2017	Imagenet samples	Single-/multi- class generation
BeGAN [52]	2017	Facial Images	Single-/multi- class generation
Bidirectional GAN [53]	2017	Imagenet samples	Single-/multi- class generation
Big GAN [27]	2018	Imagenet sample	Single-/multi- class generation
Boundary-Seeking GAN [54]	2018	Digit generation	Single-/multi- class generation
CGAN [3]	2014	Digit generation	Single-/multi- class generation
Context conditional GAN [55]	2016	Multiple objects	Inpainting
Context Encoder [56]	2016	Multiple objects	Inpainting
CO GAN [57]	2016	Imagenet samples	Single-/multi- class generation
Cycle GAN [24]	2018	Multiple objects	Single-/multi- class generation, inter domain adaptation
DC GAN [21]	2016	Digit generation, facial images	Single-/multi- class generation
DiscoGAN [25]	2017	Multiple objects	Intra-domain and Interdomain adaptation
Dual GAN [40]	2018	Photo->sketch	Intra-domain adaptation
EBGAN [41]	2017	Imagenet samples	Single-/multi- class generation
GAN [1]	2014	Digit generation, facial images	Single-/multi- class generation
Info GAN [42]	2016	Digit generation, facial images	Single-/multi- class generation
LAPGAN [43]	2015	Imagenet samples	Single-/multi- class generation
LOGAN [44]	2018	Logo generation	Single-/multi- class generation
LSGAN [45]	2017	Indoor/outdoor scene	Single-/multi- class generation
Pix2pix [46]	2018	Multiple objects	Intra-domain and Interdomain adaptation
Pro GAN [57]	2018	Facial images	Super resolution, inter-domain
Semi-supervised GAN [58]	2016	Digit generation	Single-/multi- class generation
Stack GAN [59]	2016	Image generation from text	Single-/multi- class generation
Star GAN [63]	2017	Multiple objects	Image -image translation
Style GAN [64]	2019	Facial images	Single-/multi- class generation
Super-resolution GAN [26]	2017	Facial images	Super resolution
W GAN [23]	2017	Multiple objects	Single-/multi- class generation

2. **Selecting GANs for image Generation**: Before starting to generate images first we need to select the GANs available, out of the 27 GANs that were reviewed we choose 6 GANs based on the types of GANs. Selection was done keeping in mind to at least take one GAN from each type of GAN. We selected DcGAN, WGAN, BigGAN from Single/multiple class generation, SRGAN from Resolution type and DiscoGAN and Cycle GAN were used for Intra and inter domain

3. **Generating Images using GANs**: After the selection of GAN, we used 6 datasets to run these GANs. While generating images we changed some of the hyper-parameters in the GANs to produce better quality images. In case of DcGAN, new layer of convolution-transpose was also added to get better images of range 128X128. All GANs images for all databases that were generated were 128X 128 in size to maintain their equality

4. **Evaluating fake images with NR-IQA**: Fake images generated by different GANs are then passed through three NR-IQA metrics: NIQE, PIQE, BRISQUE. Score was computed for all image datasets and even score was computed for the highest 100 images in the dataset.

5. **Finding best NR-IQA** : We analyze the above data to find the best metrics that can be used for evaluating images generated by GANs.

2.4 Milestone and deliverables

Six project stages were chosen before the start of the project.

- Literature review- For the first 2 weeks, it was decided to learn about GAN and gather as much information as we can find on the image assessment quality metrics.

- Understanding GANs Code - GANs can have more than one type of structure, so it was necessary to understand the type and structure of the GANs.

- Image generation- Nearly by the end of first month, we started generating images. The dataset that we decided to use was mnist as it was small.

- Tuning GANs hyper-parameter-After about a month, we started to change the parameter of GANs, including learning rate, batch size and in one case changing the architecture of the neural networks, i.e., adding another convolutional layer to DcGAN.

- Image quality evaluation - After the generation of images, the next part was to run the experiments to assess using no-reference quality metrics.

- Report work- Last 20 days were devoted to write report.

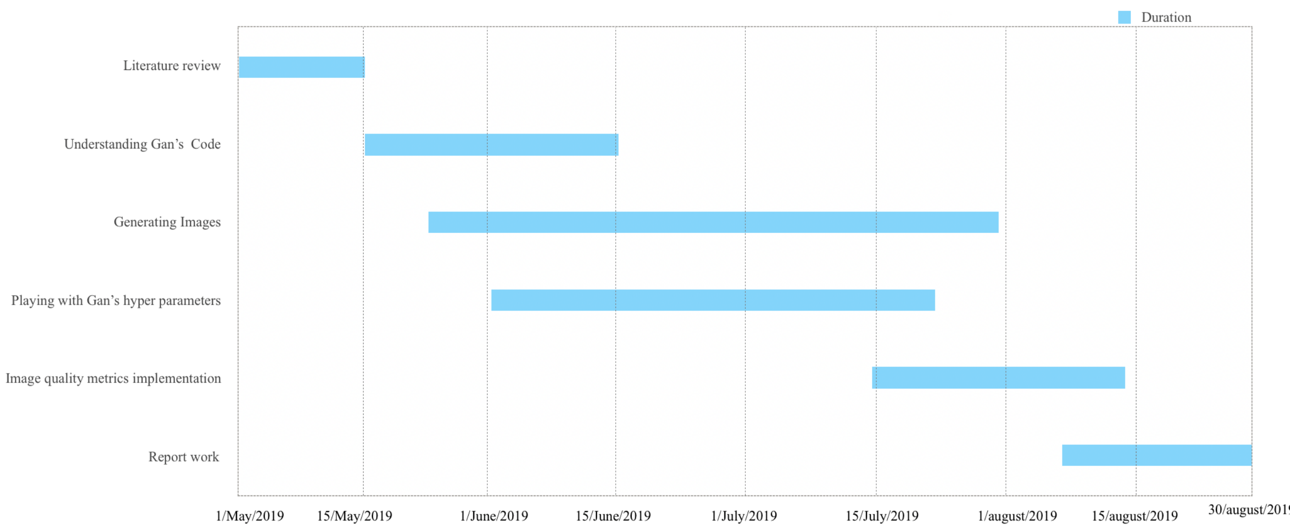


Figure 9: Milestone

3 Result

As stated in objective, entire project is divided into 4 steps. After completion of first step to review and select the GANs, we generated of the fake images using the selected GANs We used Mnist, celeba, stl10 and Lsun on DcGAN, WGAN and SRGAN. One of the GAN chosen was BigGAN, providing state of art result but requiring a large amount of computing power; due to the GPU limitation, we were only able to examine a small dataset, i.e., mnist, using that GAN. For DiscoGAN and CycleGAN we used Faade and Maps dataset to convert from one domain to another. This section illustrates some sample images generated by WGAN (Figure 10), DcGAN (Figures 11, 12, and 13), Disco GAN (Figure 14), and CycleGAN (Figure 15).

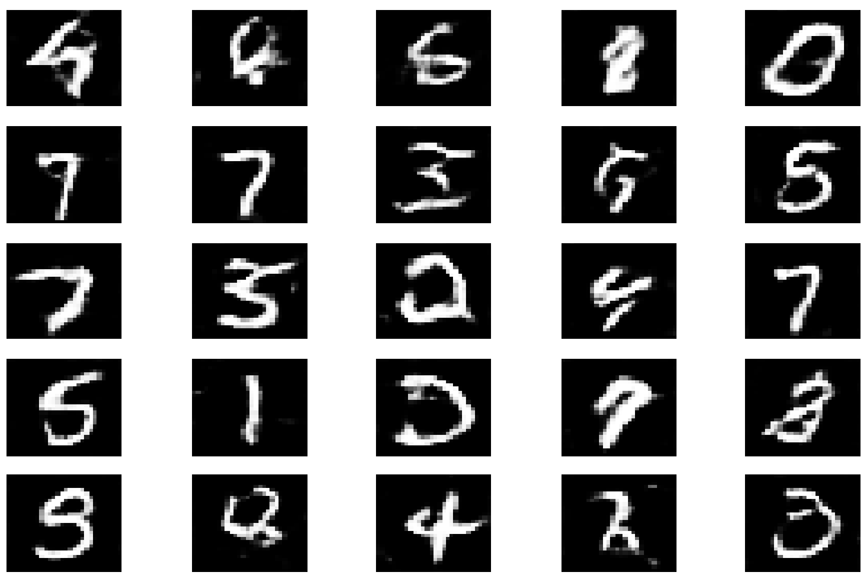


Figure 10: Images generated by WGAN using Mnist dataset

Figure 11: Images generated by DcGAN using Celeba dataset

Figure 12: Images generated by DcGAN using stl10 dataset

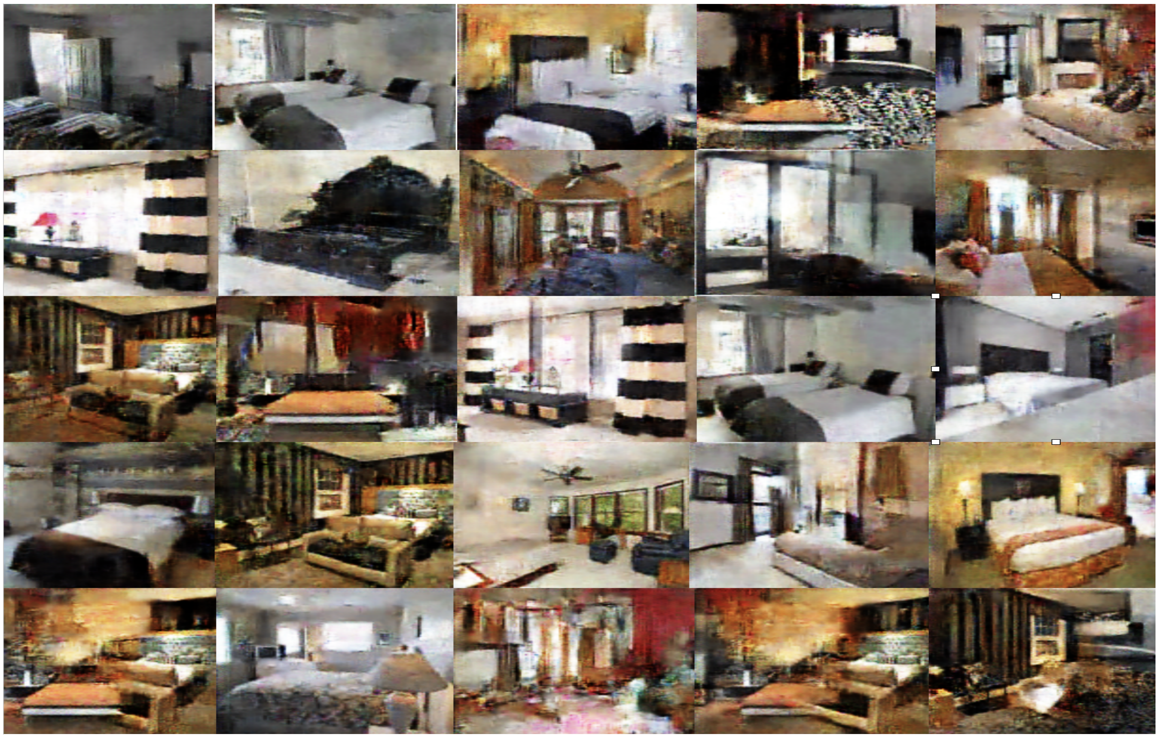


Figure 13: Images generated by DcGAN using lsun(bedroom) dataset

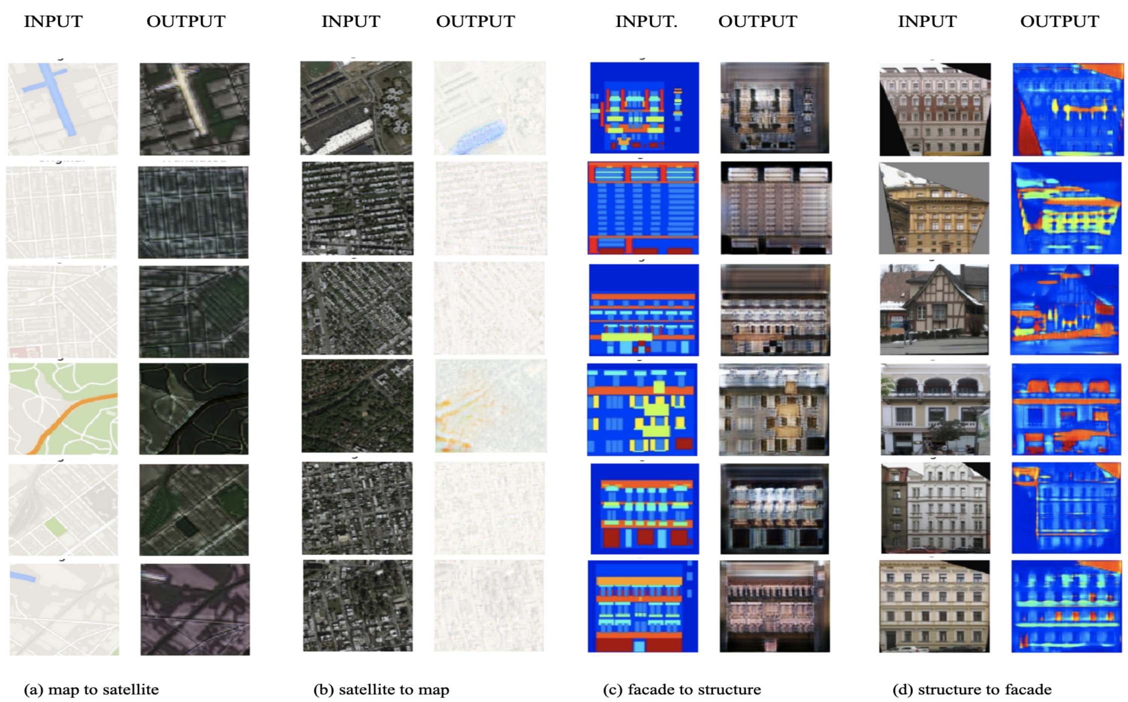
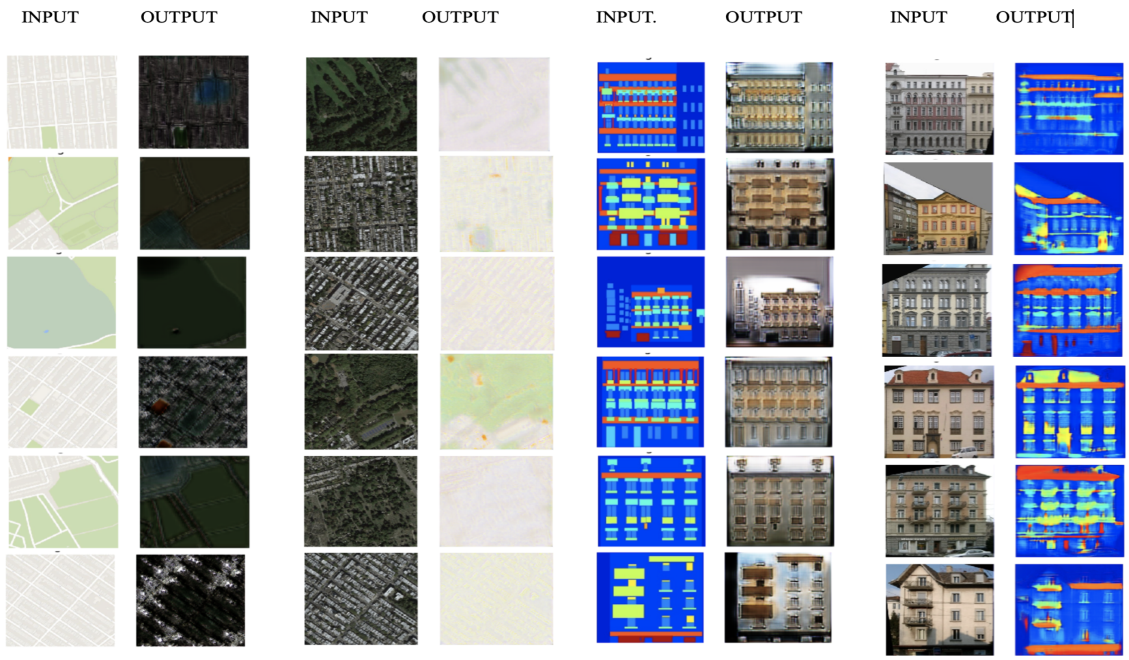


Figure 14: Images generated by CycleGAN using maps and Facade dataset



(a) map to satellite (b) satellite to map (c) facade to structure (d) structure to facade

Figure 15: Images generated by DiscoGAN using maps and Facade dataset

The quality assessment results using NIQE, BRISQUE, and PIQE metrics on the generated fake images can be seen in the table on next page. In this table, we can see the quality assessment values for each dataset for all the used GANs. Each cell includes 2 values, where the left value indicates the performance obtained for the whole generated images and the right one reports the performance for the best 100 images that are manually selected from the generated images. We have also reported the quality assessment values for the original datasets used for the training that can be useful to also examine the differences in quality. It should be noted that a lower number indicates a better quality for all the used metrics.

Dataset	GAN's	NIQE	BRISQUE	PIQE
Mnist	Original	18.6587/ 18.5643	38.1266 / 32.1167	85.0243 / 75.0907
	Dcgan	18.8709 / 18.8673	38.2955 / 28.5808	87.0434 / 64.1869
	Wgan	18.8707 / 18.6872	38.6311 / 28.9912	83.9722 / 58.3731
	Srgan	8.3448 / 7.2105	49.9061 / 45.3297	99.8027 / 97.6729
	BIgGan	18.8721 / 18.8580	46.2334 / 37.6596	80.0703 / 29.7837
CelebA	Original	5.3933 / 2.4375	32.27787 / 2.0380	45.7386 / 21.5171
	Dcgan	18.8774 / 18.8623	19.7617/ 5.4323	26.50765/22.3476
	Wgan	18.8797 / 18.8727	29.9385 / 9.8264	29.9614 / 7.8910
	Srgan	4.8667 / 3.9650	51.8695 / 40.0724	77.5656 / 57.3768
Stl10	Original	16.8780 / 16.8643	21.5078 / 0.1793	10.3388 / 2.1182
	Dcgan	18.8795 / 18.8671	23.5778 / 2.4235	18.1842 / 2.7069
	Wgan	18.8815 / 18.8721	26.2802 / 6.0918	18.7242 /2.3942
	Srgan	6.6345 / 4.6539	56.8309 / 45.4594	88.6827 / 68.3377
Lsun(Bedroom)	Original	4.4129 / 3.0070	23.9201 / 11.7412	48.4080 / 64.1869
	Dcgan	18.8770 / 18.8657	30.0018 / 0.72515	42.3246 / 6.7021
	Wgan	18.8797 / 18.8711	23.3678 / 4.8763	19.9786 / 3.4431
	Srgan	3.7431 / 3.0400	39.0657 / 23.9480	64.9338 / 42.6421
Facade	Original	7.1336 / 4.8051	35.9817 / 23.8571	52.1947 / 41.3197
	DiscoGan	9.1752 / 9.0952	34.2779 / 32.4753	43.4911 / 37.4132
	Cyclegan	18.8805 / 18.8716	37.6413 / 27.1612	48.8984 / 36.1898
Maps	Original	3.0765 / 2.2415	24.9968 / 10.9307	20.4085 / 13.1029
	DiscoGan	6.6111/5.3052	35.3532/26.4389	20.7796 / 10.2828
	Cyclegan	18.8805 / 18.7368	29.0221 / 14.6162	20.2697 / 8.8283

It can be observed from Figure 16 that NIQE values for the original dataset is lower than all the examined GANs, except SrGAN. It is expected as the super-resolution GAN improve the image quality than the original low-resolution dataset. It should be noted that we may not compare SrGAN with other GANs as it purpose is to generate high resolution image rather than generating fake images. For four datasets including Mnist, CelebA, Lsun and Stl10, we can see that DcGAN performs better than WGAN in term of NIQE values. Concerning other two datasets, including Facade and Maps, DiscoGAN performs better than Cycle GAN.
Regarding BRISQUE metric, as Table presents, the results are not stable. However, by considering the BRISQUE values for the best 100 images, Mnist, CelebA, and Stl10 datasets. It means that among the large number of images generated by DcGAN, there are some images with very good quality. The same can be concluded for CycleGAN as its BRISQUE results show a better performance than DiscoGAN.

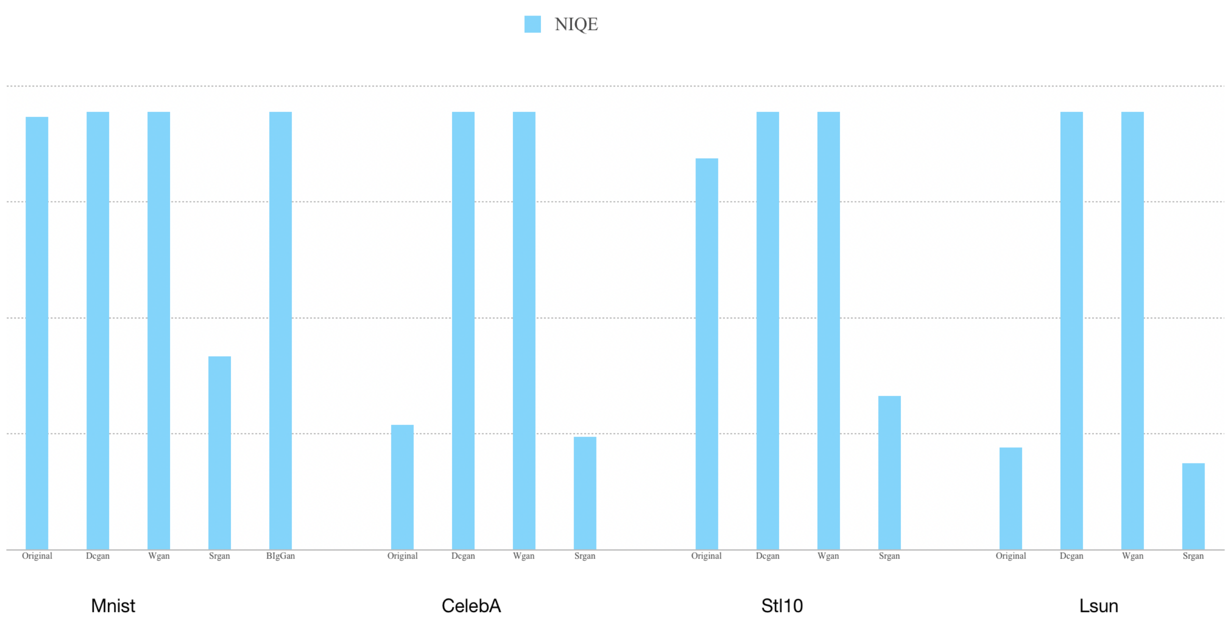


Figure 16: NIQE value for differet Gan's and dataset

Concerning PIQE, we can see that WGAN achieves the lowest values for 4 datasets, including Mnist, CelebA, Stl10 and Lsun. In some cases, it can be observed that the PIQE performance for WGAN is even less (better) than that for the original dataset. When comparing DiscoGAN and CycleGAN for facade and Maps, we can see that cycle GAN images have better quality than DiscoGAN.

4 Conclusion

This report assessed the quality of the image generated by 6 different GANs using three different no-reference image quality assessment metrics. The experiments were conducted on six different databases, showing that DcGAN and CycleGAN are the best performing GANs, respectively in fake image generating from noise and translating images from a domain to another domain (inter-domain adaptation). This report also studied the most important GANs that have gained most popularity in the recent years. In includes reviewing and categorizing twenty-seven GANs based on their type, dataset used and year they were published.

As the results showed, the objective evaluation provided us with the opportunity to understand which GAN can generate better images. However, the results of the different metrics do not always show the same behavior. Therefore, as future work, we will perform a subjective test and then compare objective and subjective assessments results to find the best no-reference assessment metrics that can additionally help to select the best performing GAN.

References

[1] *Goodfellow, I.J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A.C., Bengio, Y. (2014). Generative Adversarial Nets. NIPS.*

[2] Makhzani, A., Shlens, J., Jaitly, N., Goodfellow, I.J. (2015). Adversarial Autoencoders. ArXiv, abs/1511.05644.

[3] Isola, P., Zhu, J., Zhou, T., Efros, A.A. (2016). Image-to-Image Translation with Conditional Adversarial Networks. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 5967-5976.

[4] Guo, J., Chao, H. (2017). Building an End-to-End Spatial-Temporal Convolutional Network for Video Super-Resolution. AAAI. ./

[5] Liu, Y., Yan, J., Ouyang, W. (2017). Quality Aware Network for Set to Set Recognition. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 4694-4703/

[6] Zhang, K.R., Zuo, W., Gu, S., Zhang, L. (2017). Learning Deep CNN Denoiser Prior for Image Restoration. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2808-2817.

[7] Zhang, L., Li, H. (2012). SR-SIM: A fast and high performance IQA index based on spectral residual. 2012 19th IEEE International Conference on Image Processing, 1473-1476.

[8] Wang, Z., Bovik, A.C. (2006). Modern Image Quality Assessment. Modern Image Quality Assessment.

[9] Kim, J., Lee, S. (2017). Deep Learning of Human Visual Sensitivity in Image Quality Assessment Framework. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 1969-1977.

[10] Golestaneh, S.A., Karam, L.J. (2016). Reduced-Reference Quality Assessment Based on the Entropy of DWT Coefficients of Locally Weighted Gradient Magnitudes. IEEE Transactions on Image Processing, 25, 5293-5303.

[11] Ye, P., Kumar, J., Doermann, D.S. (2014). Beyond Human Opinion Scores: Blind Image Quality Assessment Based on Synthetic Scores. 2014 IEEE Conference on Computer Vision and Pattern Recognition, 4241-4248.

[12] Kang, L., Ye, P., Li, Y.C., Doermann, D.S. (2014). Convolutional Neural Networks for No-Reference Image Quality Assessment. 2014 IEEE Conference on Computer Vision and Pattern Recognition, 1733-1740

[13] Tang, H., Joshi, N., Kapoor, A. (2014). Blind Image Quality Assessment Using Semi-supervised Rectifier Networks. 2014 IEEE Conference on Computer Vision and Pattern Recognition, 2877-2884.

[14] Zhang, Peng, WenGANg Zhou, Lei Wu and Houqiang Li. SOM: Semantic obviousness metric for image quality assessment. 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2015): 2394-2402

[15] Xu, J., Ye, P., Li, Q., Du, H., Liu, Y., Doermann, D.S. (2016). Blind Image Quality Assessment Based on High Order Statistics Aggregation. IEEE Transactions on Image Processing, 25, 4444-4457

[16] Lin, K., Wang, G. (2018). Hallucinated-IQA: No-Reference Image Quality Assessment via Adversarial Learning. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 732-741.

24. 25. 26. 27.

[17] LeCun, Y., Cortes, C. (2005). The mnist database of handwritten digits.

[18] Celeba dataset: http://mmlab.ie.cuhk.edu.hk/projects/CelebA.html

[19] Stl10 dataset https://cs.stanford.edu/ acoates/stl10/

[20] Yu, F., Zhang, Y., Song, S., Seff, A., Xiao, J. (2015). LSUN: Construction of a Large-scale Image Dataset using Deep Learning with Humans in the Loop. ArXiv, abs/1506.03365.

[21] Radford, A., Metz, L., Chintala, S. (2015). Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks. CoRR, abs/1511.06434.

[22] Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., Schiele, B. (2016). The Cityscapes Dataset for Semantic Urban Scene Understanding. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 3213-3223.

[23] Arjovsky, M., Chintala, S., Bottou, L. (2017). Wasserstein GAN. ArXiv, abs/1701.07875.

[24] Zhu, J., Park, T., Isola, P., Efros, A.A. (2017). Unpaired Image-to-Image Translation Using Cycle-Consistent Adversarial Networks. 2017 IEEE International Conference on Computer Vision (ICCV), 2242-2251.

[25] Kim, T., Cha, M., Kim, H., Lee, J.K., Kim, J. (2017). Learning to Discover Cross-Domain Relations with Generative Adversarial Networks. ICML.

[26] Ledig, C., Theis, L., Huszar, F., Caballero, J., Cunningham, A., Acosta, A., Aitken, A., Tejani, A., Totz, J., Wang, Z., Shi, W. (2016). Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 105-114.

[27] Brock, A., Donahue, J., Simonyan, K. (2018). Large Scale GAN Training for High Fidelity Natural Image Synthesis. ArXiv, abs/1809.11096.

[28] Mittal, A., Soundararajan, R., Bovik, A.C. (2013). Making a Completely Blind Image Quality Analyzer. IEEE Signal Processing Letters, 20, 209-212.

[29] Perception based IQA https://www.mathworks.com/help/images/ref/piqe.html

[30] Mittal, A., Soundararajan, R., Bovik, A.C. (2013). Making a Completely Blind Image Quality Analyzer. IEEE Signal Processing Letters, 20, 209-212.

[31] Huynh-Thu, Q., Ghanbari, M. (2008). Scope of validity of PSNR in image/video quality assessment.

[32] Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P. (2004). Image quality assessment: from error visibility to structural similarity. IEEE Transactions on Image Processing, 13, 612.

[33] Chen, M., Bovik, A.C. (2010). Fast structural similarity index algorithm. 2010 IEEE International Conference on Acoustics, Speech and Signal Processing, 994-997.

[34] Wang, Z., Simoncelli, E.P., Bovik, A.C. (2003). Multiscale structural similarity for image quality assessment. The Thrity-Seventh Asilomar Conference on Signals, Systems Computers, 2003, 2, 1398-1402 Vol.2.

[35] Li, C., Bovik, A.C. (2010). Content-partitioned structural similarity index for image quality assessment. Sig. Proc.: Image Comm., 25, 517-526.

[36] Kang, L., Ye, P., Li, Y.C., Doermann, D.S. (2015). Simultaneous estimation of image quality and distortion via multi-task convolutional neural networks. 2015 IEEE International Conference on Image Processing (ICIP), 2791-2795.

[37] Bianco, S., Celona, L., Napoletano, P., Schettini, R. (2016). On the use of deep learning for blind image quality assessment. Signal, Image and Video Processing, 12, 355-362

[38] Zeng, H., Zhang, L., Bovik, A.C. (2017). A Probabilistic Quality Representation Approach to Deep Blind Image Quality Prediction. ArXiv, abs/1708.08190.

[39] Sheikh, H.R., Bovik, A.C. (2006). Image information and visual quality. IEEE Trans. Image Processing, 15, 430-444.

[40] Yi, Z., Zhang, H., Tan, P., Gong, M. (2017). DualGAN: Unsupervised dual learning for image-to-image translation. In Proceedings of the IEEE international conference on computer vision (pp. 2849-2857).

[41] Berthelot, D., Schumm, T., Metz, L. (2017). BeGAN: Boundary equilibrium generative adversarial networks. arXiv preprint arXiv:1703.10717.

[42] Chen, X., Duan, Y., Houthooft, R., Schulman, J., Sutskever, I., Abbeel, P. (2016). InfoGAN: Interpretable representation learning by information maximizing generative adversarial nets. In Advances in neural information processing systems(pp. 2172-2180).

[43] Lai, W. S., Huang, J. B., Ahuja, N., Yang, M. H. (2017). Deep laplacian pyramid networks for fast and accurate super-resolution. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 624-632).

[44] Ridley, B. K. (1996). The LO phonon lifetime in GAN. Journal of Physics: Condensed Matter, 8(37), L511.

[45] Zhu, X., Liu, Y., Li, J., Wan, T., Qin, Z. (2018, June). Emotion classification with data augmentation using generative adversarial networks. In Pacific-Asia Conference on Knowledge Discovery and Data Mining (pp. 349-360). Springer, Cham.

[46] Hesse, C. Github tensorflow implementation of isola et al.s pix2pix model, 2017. URL https://github. com/affinelayer/pix2pix-tensorflow.

[47] Kamble, V.M., Bhurchandi, K.M. (2015). No-reference image quality assessment algorithms: A survey.

[48] 22-C edric Villani. Optimal Transport: Old and New. Grundlehren der mathema- tischen Wissenschaften. Springer, Berlin, 2009.

[49] Mittal, A., Moorthy, A.K., Bovik, A.C. (2012). No-Reference Image Quality Assessment in the Spatial Domain. IEEE Transactions on Image Processing, 21, 4695-4708.

[50] Salimans, T., Goodfellow, I.J., Zaremba, W., Cheung, V., Radford, A., Chen, X. (2016). Improved Techniques for Training GANs. ArXiv, abs/1606.03498.

[51] Odena, A., Olah, C., Shlens, J. (2016). Conditional Image Synthesis With Auxiliary Classifier GANs. ICML.

[52] Berthelot, D., Schumm, T., Metz, L. (2017). BEGAN: Boundary Equilibrium Generative Adversarial Networks. ArXiv, abs/1703.10717.

[53] Seo, M.J., Kembhavi, A., Farhadi, A., Hajishirzi, H. (2016). Bidirectional Attention Flow for Machine Comprehension. ArXiv, abs/1611.01603.

[54] Hjelm, R.D., Jacob, A.P., Trischler, A., Che, G., Cho, K., Bengio, Y. (2018). Boundary Seeking GANs. ICLR.

[55] Denton, E., Gross, S., Fergus, R. (2016). Semi-supervised learning with context-conditional generative adversarial networks. arXiv preprint arXiv:1611.06430.

[56] Pathak, D., Krahenbuhl, P., Donahue, J., Darrell, T., Efros, A. A. (2016). Context encoders: Feature learning by inpainting. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 2536-2544).

[57] Karras, T., Aila, T., Laine, S., Lehtinen, J. (2017). Progressive growing of GANs for improved quality, stability, and variation. arXiv preprint arXiv:1710.10196..

[58] Odena, A. (2016). Semi-supervised learning with generative adversarial networks. arXiv preprint arXiv:1606.01583.

[59] Zhang, H., Xu, T., Li, H., Zhang, S., Wang, X., Huang, X., Metaxas, D. N. (2017). StackGAN: Text to photo-realistic image synthesis with stacked generative adversarial networks. In Proceedings of the IEEE International Conference on Computer Vision (pp. 5907-5915).

[60] Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A., Chen, X. (2016). Improved techniques for training GANs. In Advances in neural information processing systems (pp. 2234-2242).

[61] Wang, Z., Li, Q. (2010). Information content weighting for perceptual image quality assessment. IEEE Transactions on Image Processing, 20(5), 1185-1198.

[62] Zhang, H., Goodfellow, I., Metaxas, D., Odena, A. (2018). Self-attention generative adversarial networks. arXiv preprint arXiv:1805.08318.

[63] Choi, Y., Choi, M., Kim, M., Ha, J. W., Kim, S., Choo, J. (2018). StarGAN: Unified generative adversarial networks for multi-domain image-to-image translation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 8789-8797).

[64] Karras, T., Laine, S., Aila, T. (2019). A style-based generator architecture for generative adversarial networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 4401-4410)