Enhancing Pixelwise Generative Adversarial Networks through Advanced Loss Functions and Regularization Techniques

Vinjamuri Stanley Gabriel, Chris Thomas
Virginia Tech
Blacksburg, VA
{stanleygabriel97, chris}@vt.edu

Abstract

In this work, we successfully integrated focal loss, a novel loss function that concentrates training on hard misclassified examples, into various generative adversarial network (GAN) architectures for image synthesis tasks. Focal loss was applied to the objective tasks of unconditional image generation, conditional image generation, and the pixel-wise image-to-image translation task. The primary goal was to mitigate common issues faced by GANs, such as mode collapse and training instability, thereby improving the quality and stability of pixel-level image synthesis. We systematically evaluated the performance of focal loss against traditional losses like cross-entropy, mean squared error, and Wasserstein loss, along with various regularization techniques. Extensive experiments were conducted across the aforementioned generative tasks, analyzing convergence behavior through loss curves and qualitatively assessing generated image quality. Using focal loss led to a smoother and more stable loss curve as compared to vanilla GAN loss, while LSGAN and WGAN loss still performed better in terms of convergence and training stability. In regards to image quality, focal loss showed promising image generation result at par with LSGAN for similar epoch durations. A thorough analysis was done comparing multiple architectures, loss functions and robust datasets.

1. Introduction

Generative Adversarial Networks (GANs) have revolutionized the field of generative modeling, offering powerful tools for generating high-fidelity, realistic images. **Pixelwise GANs** take this a step further by focusing on generating images where each pixel is adjusted individually, enabling finer control and higher quality in tasks such as image-to-image translation, super-resolution, denoising and more [4] [15] [7] [2]. However, the training stability and output quality of GANs can be significantly affected by the choice of loss functions and regularization techniques. This paper explores innovative adaptations in the training of GANs, focusing on integrating varied loss functions and advanced regularization methods to enhance image synthesis quality and model robustness.

The researchers implemented traditional GAN loss [10], Least Squares GAN (LSGAN) loss [13], Wasserstein GAN (WGAN) loss along with gradienty penalty [1] [3], and Focal Loss to compare image-to-image translation for every loss function with different GAN architectures. The novel approach implemented in this project was to find how focal loss gives enhanced results to different image-to-image translations. In this project, we explore the integration of Focal Loss, a loss function initially proposed for dense object detection [5] [16], with GANs to mitigate these challenges and improve the performance of image generation and image-to-image translation tasks. By emphasizing hard-to-classify examples and downweighting well-classified ones, Focal Loss encourages the model to focus on challenging cases, potentially leading to better generalization and convergence. In this study, we investigate the impact of incorporating Focal Loss into the adversarial training process of GANs. Specifically, we apply Focal Loss to the discriminator network, aiming to improve its ability to distinguish between real and generated samples. By dynamically adjusting the loss contribution of each sample, we hypothesize that the discriminator can better guide the generator towards producing more realistic and diverse outputs, ultimately enhancing the quality of generated images and the performance of image-to-image translation tasks.

2. Related Work

To some extent, classic GANs use a binary cross-entropy loss function to compete against each other in the minimax game. However, this loss function has been proven not to work well enough because of mode collapse that traps generation at a local optimum and produces a small set of sam-

ples [18] [19] [17] [11]. It is because the discriminator's unbalanced loss leads it to saturate once it identifies between genuine and counterfeit instances with confidence. Alternative to this, newer cost functions like least-squares, earth-mover's distance were implemented to counter this problem.

Similar to this study, introducing focal loss into Generative Adverserial Networks was performed by Gao et al. [9] in 2020 for image to image generation scenario. The work introduced focal losses for both generator and discriminator along with an enhanced self attention mechanism which claims to improve representational capacity of features in the generator. The work has promising results in terms of effectiveness of addition of focal loss as the cost function when compared to other loss functions.

The project also investigated into spectral normalization as a technique to improve the training stability of Pixelwise GANs. This technique tackles the issue of exploding gradients in deep neural networks by normalizing the weight matrices in the discriminator using their spectral norms. This normalization implicitly [6] enforces a Lipschitz constraint, producing smoother gradients and more stable training.

3. Methodology

3.1. Architecture

The researchers used Deep Convolutional Generative Adversarial Networks (DCGAN) architecture which introduces several deep convolutional layers to the discriminator and generator models of the GAN architecture [14]. This structure enhances the quality of the generated images for the generator model and stabilizes the training process. The convolutional architecture of DCGAN unlike any other standard GANs, uses CNNs that are more effective in handling image data. The activation functions for DCGAN incorporate the ReLU activation function for the generator model to promote non-linearity for all hidden layers, and the tanh function is used in the output layer to scale the image pixel values. The discriminator model uses LeakyReLU to provide a pathway for gradients in layers where the neuron would otherwise completely shut off, preventing the vanishing gradient problem. The generator takes a noisy latent vector as the input which is generated from a normal distribution sample space and goes through layers of transpose convolution, ReLU, and batch normalization to generate a color image of required dimension. The discriminator takes real images from the dataset and fake images which were created by the generator as its input. These inputs are then passed through convolutional layers, LeakyReLU, and batch-normalization and the discriminator produces a final output deciding whether the input is real or fake. Figure 1 shows the DCGAN architecture introduced by Radford et al. in 2015.

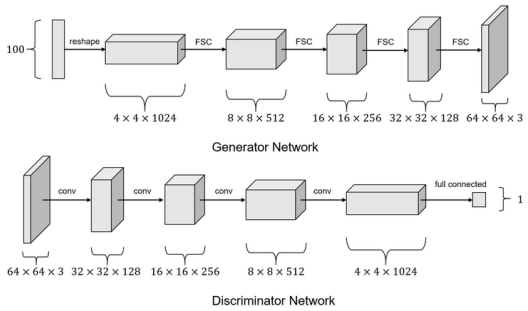

Figure 1. Generator and Discriminator architecture for DCGAN [8]

3.2. Loss functions

The researchers aimed to enhance the performance of Pixelwise Generative Adversarial Networks (GANs) by integrating a sophisticated blend of loss functions and regularization methods. The loss functions included in this study are:

1. **Vanilla GAN Loss** [10]: The original GAN loss function, which aims to minimize the Jensen-Shannon divergence between the real and generated data distributions.
$\min_G \max_D V(D,G) = \mathbb{E}_{x\sim p_{\text{data}}(x)}[\log D(x)] + \mathbb{E}_{z\sim p_z(z)}[\log(1-D(G(z)))]$

2. **Least Squares GAN (LSGAN) Loss** [13]: A modification of the vanilla GAN loss that replaces the cross-entropy loss with a least-squares loss, which has been shown to stabilize training and improve the quality of generated images.
$\min_G \max_D V(D,G) = \frac{1}{2}\mathbb{E}_{x\sim p_{\text{data}}(x)}[(D(x)-1)^2] + \frac{1}{2}\mathbb{E}_{z\sim p_z(z)}[D(G(z))^2]$

3. **Wasserstein GAN (WGAN) Loss** [1]: A loss function based on the Wasserstein distance, utilizing the Earth-Mover's distance to provide a more stable and meaningful training process, represented mathematically as $\min_G \max_{D\in\mathcal{D}} \mathbb{E}_{x\sim\mathbb{P}_r}[D(x)] - \mathbb{E}_{\tilde{x}\sim\mathbb{P}_g}[D(\tilde{x})]$, where $\mathcal{D}$ is the set of 1-Lipschitz functions, and $\mathbb{P}_r, \mathbb{P}_g$ are the data and model distributions, respectively. In the original Wasserstein GAN work, the Lipschitz constraint was employed using weight clipping which was deemed as an incorrect way of enforcing the constraint. Improved training and loss function was introduced by Guljrani et al. which introduced a regularization gradient penalty term along with the original critic loss. The new loss is given by $\mathbb{E}_{\hat{\mathbf{x}}\sim\mathbb{P}_g}[D(\tilde{\mathbf{x}})] - \mathbb{E}_{\mathbf{x}\sim\mathbb{P}_r}[D(\mathbf{x})] + \lambda\mathbb{E}_{\hat{\mathbf{x}}\sim\mathbb{P}_{\hat{\mathbf{x}}}}\left[(||\nabla_{\tilde{\mathbf{x}}} D(\tilde{\mathbf{x}})||_2 - 1)^2\right]$ where the second term is the gradient penalty.

4. **Focal Loss** [5]: Traditionally expressed as $FL(p_t) = -\alpha_t(1-p_t)^\gamma \log(p_t)$, where p_t is the model's estimated probability for the class label t, α_t is a weighting factor for the class, and γ is the focusing parameter. This loss is tailored to address the imbalance in the pixelwise generation task, focusing the model's learning effort on hard-to-generate pixels. Implementing this loss function for the discriminator and generator was performed using following equation $\min_G \max_D V(D,G) = \mathbb{E}_{x\sim p_{\text{data}}(x)}[1 - D(x)]^\gamma[\log D(x)] + \mathbb{E}_{z\sim p_z(z)}[D(G(z)]^\gamma[\log(1 - D(G(z))]$

For regularization in losses where Lipschitz constraint is not followed, they used spectral normalization, enforcing a Lipschitz constraint by normalizing the weight matrices W in the discriminator, thus controlling the Lipschitz constant. This is mathematically denoted as $\bar{W} = \frac{W}{\sigma(W)}$, where $\sigma(W)$ represents the spectral norm of W. This technique ensures the stability of the discriminator's training process, promoting a balanced adversarial game. Through this multifaceted approach, combining targeted loss functions with robust regularization, the researchers had significant improvements in the pixelwise synthesis accuracy, stability, and overall generative quality of the GAN model.

3.3. Pix2Pix

In addition to the DCGAN architecture for image generation, we also explore the Pix2Pix architecture [4], which has been specifically designed for supervised image-to-image translation tasks. The Pix2Pix architecture is based on a conditional Generative Adversarial Network (cGAN) that is tailored for image-to-image translation tasks where there is a paired dataset containing corresponding input and output images. This supervised approach differs significantly from unconditioned or traditional GANs that generate data from a random noise vector.

3.3.1 Generator and Discriminator Architecture

Pix2Pix uses a U-Net-based architecture for the generator, which incorporates skip connections that bypass the encoder-decoder layers, allowing direct flow of information from the input to the output. This helps preserve high-frequency details by passing content that does not need to be altered directly across the network, thereby aiding in more precise structural transformations between the input and output images.

The discriminator, often termed PatchGAN, operates on different patches of the image. It assesses whether small sections of the image are real or fake, rather than the entire image at once. This approach focuses the training on getting the structure correct at the scale of patches and is computationally more efficient because it reduces the number of parameters and stabilizes training.

3.3.2 Training and Objective Function

The training of the Pix2Pix model involves alternating between training the discriminator and the generator. The discriminator learns to differentiate between real and fake image pairs, while the generator tries to fool the discriminator. The objective function is a combination of a traditional GAN loss and an L1 loss, where the GAN loss encourages fooling the discriminator, and the L1 loss encourages the generator to be near the ground truth in a mean absolute error sense. This hybrid loss function helps the system to generate sharp and realistic images while staying close to the target distribution.

In terms of performance, Pix2Pix generally produces results that are both visually convincing and highly accurate in terms of content preservation when compared to the ground truth. The effectiveness of the model in producing high-quality results across diverse tasks highlights its utility in practical applications where paired training data is available.

This architecture can be particularly useful in tasks where detailed and precise translation between images is required, making it a valuable tool for both academic research and industrial applications in fields such as medical imaging, photo editing, and computer graphics.

4. Experimental Setup

The researchers conducted experiments on various image generation tasks, including:

1. Unconditional Image Generation: Generating realistic images from random noise vectors using the DCGAN architecture with different loss functions.

2. Conditional Image Generation: Generating images conditioned on specific attributes or labels using the DCGAN architecture with different loss functions.

3. Image-to-Image Translation: Translating images from one domain to another, such as converting sketches to photorealistic images or day scenes to night scenes, using the Pix2Pix architecture with the vanilla GAN loss, LSGAN loss, WGAN-GP loss, and Focal loss.

The dataset used for testing the architecture was Cat faces dataset for generative models, this dataset has 64x64 resized RGB images of cat faces to test the generative models implemented within the experiment. The researchers then moved to CIFAR-10 dataset for image generation. The CIFAR-10 dataset consists of 60000 32x32 color images in 10 classes, with 6000 images per class. For Pix2Pix architecture the researchers used the Facades Dataset. The

dataset consists of 506 Building Facades & corresponding Segmentations split into train and test subsets.

All models are implemented using the PyTorch deep learning framework. For image generation tasks, the researchers used the Adam optimizer with a learning rate of 0.0001 and beta values of 0.5 and 0.999. The models were trained for a fixed number of epochs which were 100 for unconditonal and conditional GAN's, and the generated images were saved at regular intervals for evaluation. For implementation of WGAN loss, n_critic values was fixed as 5 and weight clipping was 0.01. For focal loss, α and γ were selected as 1 and 2 respectively. The batch size was kept as 256 and latent noise dimenstion was 10 for all variations.

5. Evaluation and Results

The study for image generation was conducted for two architectures viz. unconditional and conditional GANs. CIFAR-10 dataset [12] was used for the training and inference in this method. The uncommon hyperparameters used for each of the different loss functions were kept consistent with the original papers. In contrast, common hyperparameters like learning rate, optimizer, their variables, and batch size were kept the same for all the models in order to have a fair evaluation.

As can be observed from the following loss figures which shows the average loss curves per epoch for both generator and discriminator for all variations. Focal loss consistently performs than vanilla GAN in terms of convergence and stability. As can be observed, the variation in vanilla GAN's Discriminator is visibly unstable as the network keeps learning. Focal loss had similar performance to MSE loss, with MSE loss having an overall better stability. As seen by the images generated by these models in Table 1 and Table 2, the quality of the images are comparable to other loss functions. Since, this is a subjective matter and the resolution of generated images are quite low, it is quite difficult to comment quantitatively the difference in the outputs. For unconditional image generation, since there is no labeling involved it is hard to point out the output image but focal loss seems to be generating horses quite well. As with conditional implementation, the generated images are following the basic schema of their labels.

In assessing the performance of our Pix2Pix models, we employed the Peak Signal-to-Noise Ratio (PSNR) as a quantitative metric to measure the fidelity of generated images against real images in our test set. Using the PSNR formula, we calculated the image quality metrics across various test cases, which provided us with insights into the stability and efficiency of our model under different loss conditions. The equations used for these calculations are included to ensure reproducibility and clarity in our evaluation methods.

$$\text{PSNR} = 10 \cdot \log_{10} \left(\frac{\text{MAX}_I^2}{\text{MSE}} \right)$$

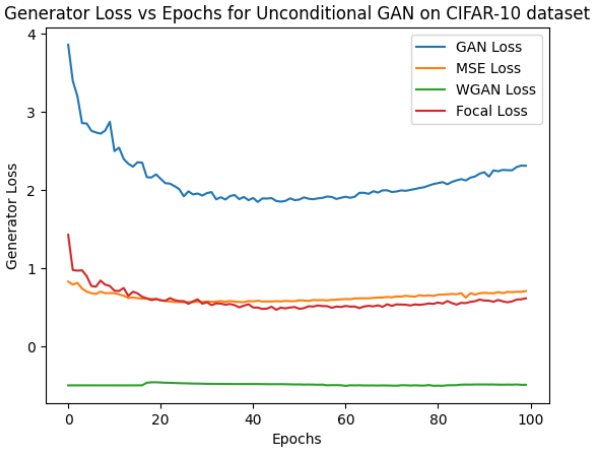

Figure 2. Generator loss for unconditional GAN's on CIFAR-10 dataset

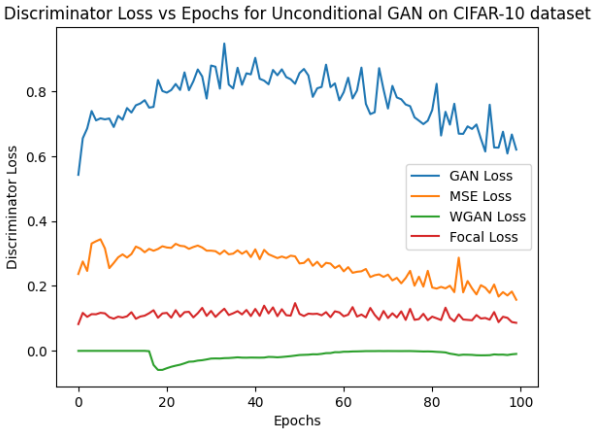

Figure 3. Discriminator loss for unconditional GAN's on CIFAR-10 dataset

6. Conclusion

The goal of the research was to experiment and analyse different methods on improving GAN's via modifying the loss functions. The research diverted from solving denoising problem to test the changes in simpler image generation and image to image translation tasks. The addition of focal loss shows promising performance in comparison to traditional GAN loss. The analysis was done on conditional CIFAR-10, unconditional CIFAR-10 and Pix2Pix. There is a requirement on detailed study of quantitative metrics of the generated images. Another future avenue would be to use focal loss for dense image generation tasks drawing parallel from its original use case of dense object detection.

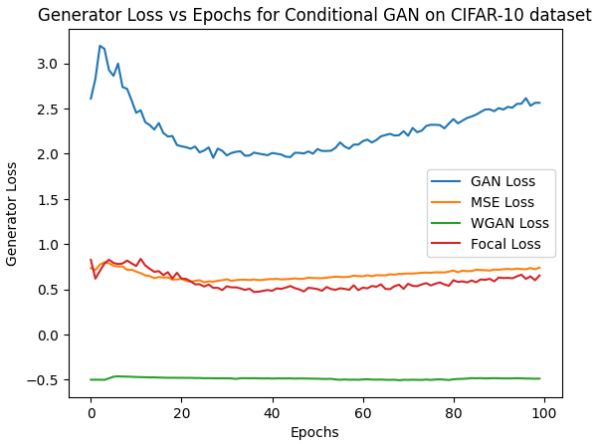

Figure 4. Generator loss for conditional GAN's on CIFAR-10 dataset

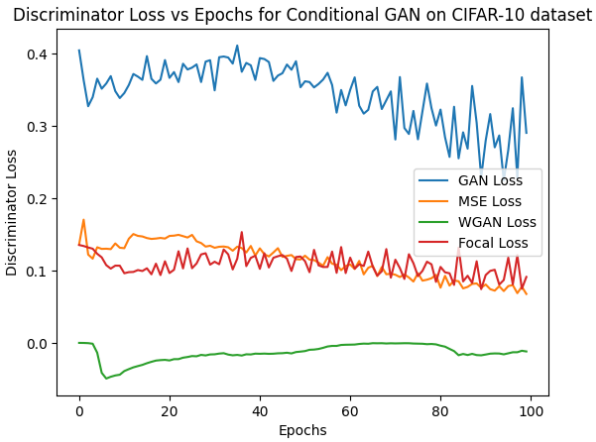

Figure 5. Discriminator loss for conditional GAN's on CIFAR-10 dataset

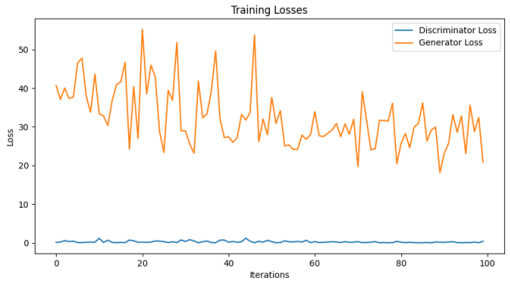

Figure 6. Discriminator and Generator loss for Pix2Pix with vanilla GAN

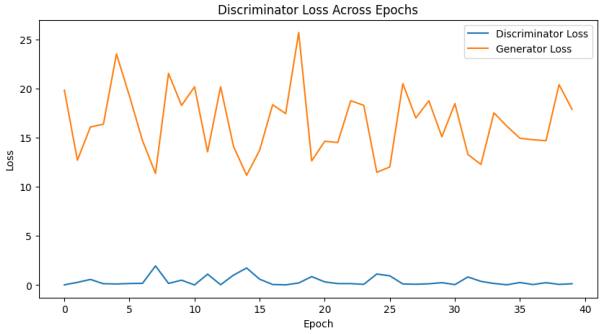

Figure 7. Discriminator and Generator loss for Pix2Pix with Focal Loss

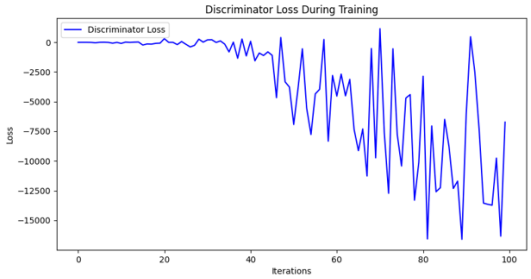

Figure 8. Discriminator loss for Pix2Pix with WGAN

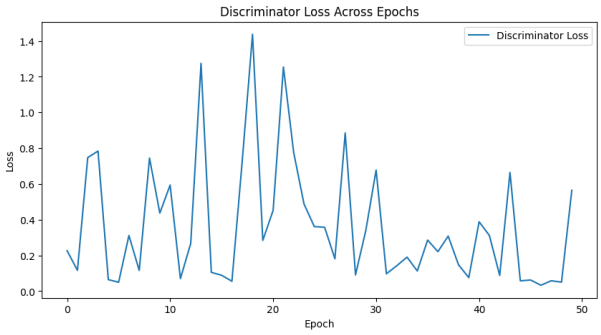

Figure 9. Discriminator loss for Pix2Pix with LS-GAN

References

[1] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein gan, 2017. 1, 2

[2] A. Buades, B. Coll, and J.-M. Morel. A non-local algorithm for image denoising. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 2, pages 60–65 vol. 2, 2005. 1

[3] Gulrajani et al. Improved training of wasserstein gans. *Advances in neural information processing systems*, 30, 2017. 1

[4] Isola et al. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1125–1134, 2017. 1, 3

[5] Lin et al. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988, 2017. 1, 3

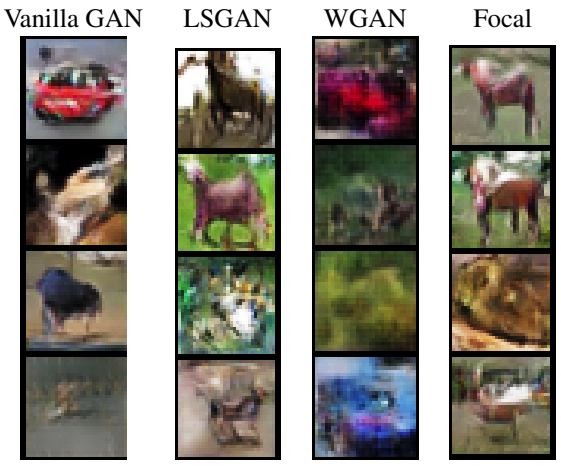

Table 1. Images generated by Unconditional GANs on CIFAR-10 dataset

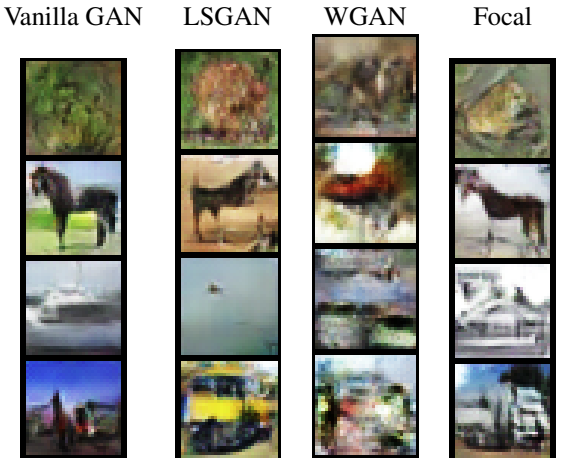

Table 2. Images generated by Conditional GANs on CIFAR-10 dataset for class labels: frog, horse, ship, and truck

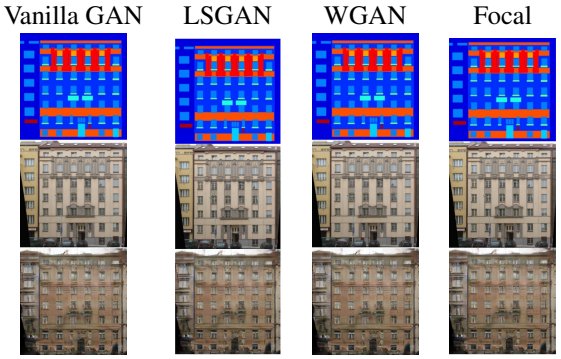

Table 3. Images generated by Pix2Pix on facades dataset.

Table 4. Pix2Pix PSNR Mean Values

Loss	PSNR Mean
Vanilla GAN	28.813
WGAN-GP	27.778
LS-GAN	28.056
Focal	28.351

[6] Xiaopeng et al. Improved training of spectral normalization generative adversarial networks. In *2020 2nd World Symposium on Artificial Intelligence (WSAI)*, pages 24–28, 2020. 2

[7] Xining Zhu et al. Gan-based image super-resolution with a novel quality loss. *Mathematical Problems in Engineering*, 2020:5217429, 2020. 1

[8] Zhang et al. A method for the estimation of finely-grained temporal spatial human population density distributions based on cell phone call detail records. *Remote Sensing*, 12(16):2572, 2020. 2

[9] Fei Gao and et al. Zhu. Incremental focal loss gans. *Information Processing & Management*, 57(3):102192, 2020. 2

[10] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020. 1, 2

[11] Youssef Kossale, Mohammed Airaj, and Aziz Darouichi. Mode collapse in generative adversarial networks: An overview. In *2022 8th International Conference on Optimization and Applications (ICOA)*, pages 1–6. IEEE, 2022. 2

[12] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. 4

[13] Xudong Mao, Qing Li, Haoran Xie, Raymond YK Lau, Zhen Wang, and Stephen Paul Smolley. Least squares generative adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2794–2802, 2017. 1, 2

[14] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015. 2

[15] Cyprien Ruffino, Romain Hérault, Eric Laloy, and Gilles Gasso. Pixel-wise conditioned generative adversarial networks for image synthesis and completion. *Neurocomputing*, 416:218–230, 2020. 1

[16] Leslie N. Smith. Cyclical focal loss, 2022. 1

[17] Akash et al. Srivastava. Veegan: Reducing mode collapse in gans using implicit variational learning. *Advances in neural information processing systems*, 30, 2017. 2

[18] Hoang Thanh-Tung and Truyen Tran. Catastrophic forgetting and mode collapse in gans. In *2020 international joint conference on neural networks (ijcnn)*, pages 1–10. IEEE, 2020. 2

[19] Zhaoyu Zhang, Mengyan Li, and Jun Yu. On the convergence and mode collapse of gan. In *SIGGRAPH Asia 2018 Technical Briefs*, pages 1–4. 2018. 2