Tuijin Jishu/Journal of Propulsion Technology
ISSN: 1001-4055
Vol. 45 No. 3 (2024)

Comparison of Loss Functions for Deblurring Images with Conditional GANs

Ayush Sharma, Mrs. Anubhooti Papola
Veer Madho Singh Bhandari Uttarakhand Technical University, Dehradun

Abstract: Generative Adversarial Networks (GANs) have transformed the field of image synthesis, particularly with the introduction of Conditional GANs (cGANs) which allow for a more customized approach by integrating extra information throughout the generative process. The presence of blurry images can have a detrimental impact on image quality and can impede subsequent image processing activities. To combat image blurriness, we introduce a novel single-image blur removal technique that relies on conditional generative adversarial networks (CGAN). In this method, CGAN acts as the fundamental framework, taking the blurred image as supplementary conditional data and enforcing a Lipschitz constraint. The network architecture is trained using a combination of conditional adversarial loss, content loss, and perception loss to rectify the blurred regions and reconstruct the image. Through experimental evaluations, it is evident that the proposed approach outperforms existing algorithms in terms of blur removal, effectively diminishing blurriness while maintaining image sharpness.

Keywords: Generative Adversarial Networks, Conditional GANs, Image Synthesis, Training Stability, Blurry Images, Generator Loss Function.

1. Introduction

The study evaluates the performance of the standard classifiers in classifying the real images from AI-generated images. To carry out the study, first we need to understand in depth about the problems that can be created using AI-generated images.

1.1 History of Image Synthesis

Image synthesis, also known as computer graphics or rendering, has a rich and fascinating history that spans several decades. The roots of image synthesis can be traced back to the mid-20th century when early pioneers in computer science began exploring the concept of creating images using computers. In the 1960s, Ivan Sutherland created the first ever computer-generated image, known as Sketchpad, which laid the foundation for future developments in this field. In the following years, researchers and innovators like David Evans, Ed Catmull, and Alvy Ray Smith made significant contributions to the field by developing techniques such as hidden surface removal and rendering algorithms. These advancements paved the way for the emergence of image synthesis as a field of study and opened up new possibilities for creating realistic and visually appealing computer-generated images. In the 1980s and 1990s, the advent of powerful graphics hardware and software further accelerated the progress in image synthesis, leading to the development of increasingly sophisticated algorithms and rendering techniques. Today, image synthesis is an essential component in a wide array of fields such as entertainment, virtual reality, simulation, and scientific visualization. This technology is constantly progressing and improving through the development of methods such as ray tracing, global illumination, and physically-based rendering. The history of image synthesis reflects the continuous pursuit of creating images that are indistinguishable from reality and has undoubtedly left an indelible mark on the field of computer graphics.

1.2 An Overview of Generative Adversarial Networks (GANs)

Generative Adversarial Networks (GANs) were first introduced by Ian Goodfellow and colleagues in 2014. This innovative machine learning framework comprises two neural networks - the generator and the discriminator -

that participate in a competitive game. GANs are primarily used as generative models for image synthesis, aiming to understand the underlying probability distribution of the training data and generate new samples that closely resemble real data. The generator network takes random noise as input to produce synthetic data, while the discriminator network's role is to differentiate between real and generated data. Through adversarial learning, GANs are trained by having the generator and discriminator networks engage in a two-player game, continuously enhancing their capabilities. The ultimate goal of training is for the generator to create data that is indistinguishable from real data by the discriminator, achieving a state where D(G(z)) ≈ 0.5.

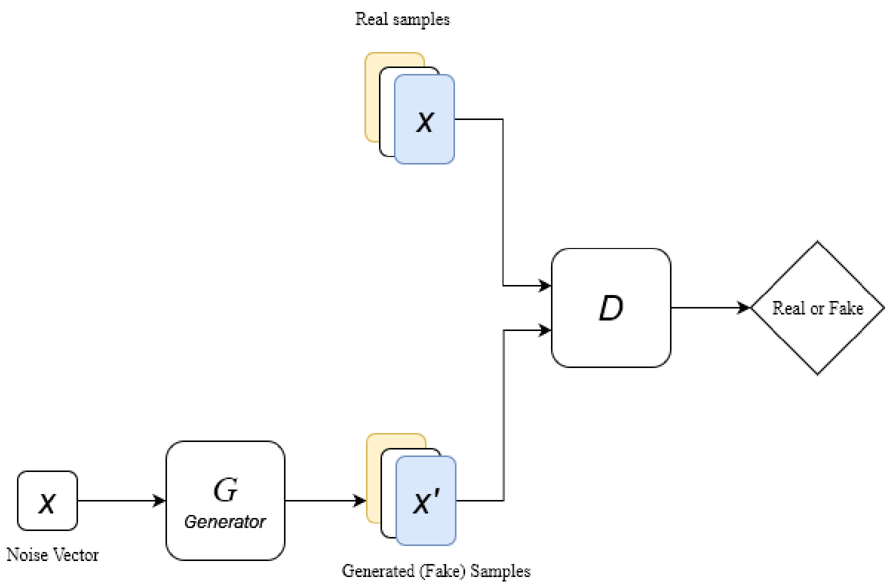

Fig. 1: Architecture of GANs

In terms of applications, Generative Adversarial Networks (GANs) are primarily utilized for tasks related to image synthesis, such as creating lifelike images from random noise, translating images from one style to another, and transferring artistic styles. They have demonstrated success in generating visually impressive and highly realistic synthetic images. Conversely, CNNs are extensively applied in a range of computer vision assignments like classifying images, detecting objects, segmenting images, and recognizing images. CNNs are praised for their ability to capture complex features and patterns within images, enabling precise predictions to be made.

It is worth noting that while GANs and CNNs have divergent architectures and purposes, there is potential for integration. CNNs can be incorporated into the architecture of GANs, combining the generative capabilities of GANs with the feature extraction abilities of CNNs. This integration holds promise for improving image synthesis results by leveraging CNNs to capture relevant features and patterns in the generated images.

1.3 Conditional Generative Adversarial Networks (cGANs)

Conditional Generative Adversarial Networks (cGANs) expand upon the GAN framework by incorporating supplementary information throughout the training phase. Conditional Generative Adversarial Networks (cGANs) involve the provision of additional conditioning information to both the generator and discriminator, typically in the form of labels or other auxiliary data. This extra information allows for the generation of targeted and specific outputs. The architecture of a cGAN involves feeding both the noise vector and the conditional information to the generator, creating a more controlled and directed generative process. The training procedure for conditional Generative Adversarial Networks (cGANs) closely resembles that of traditional GANs, but with the incorporation of conditional information. Throughout the training phase, the generator and discriminator collaborate, with the generator striving to generate authentic data based on the provided input information, while the discriminator is tasked with distinguishing between genuine and synthetic samples while considering the additional conditioning. This iterative process persists until the generator successfully produces data that is virtually indistinguishable from real data, and the discriminator is no longer able to discern between the two sources.

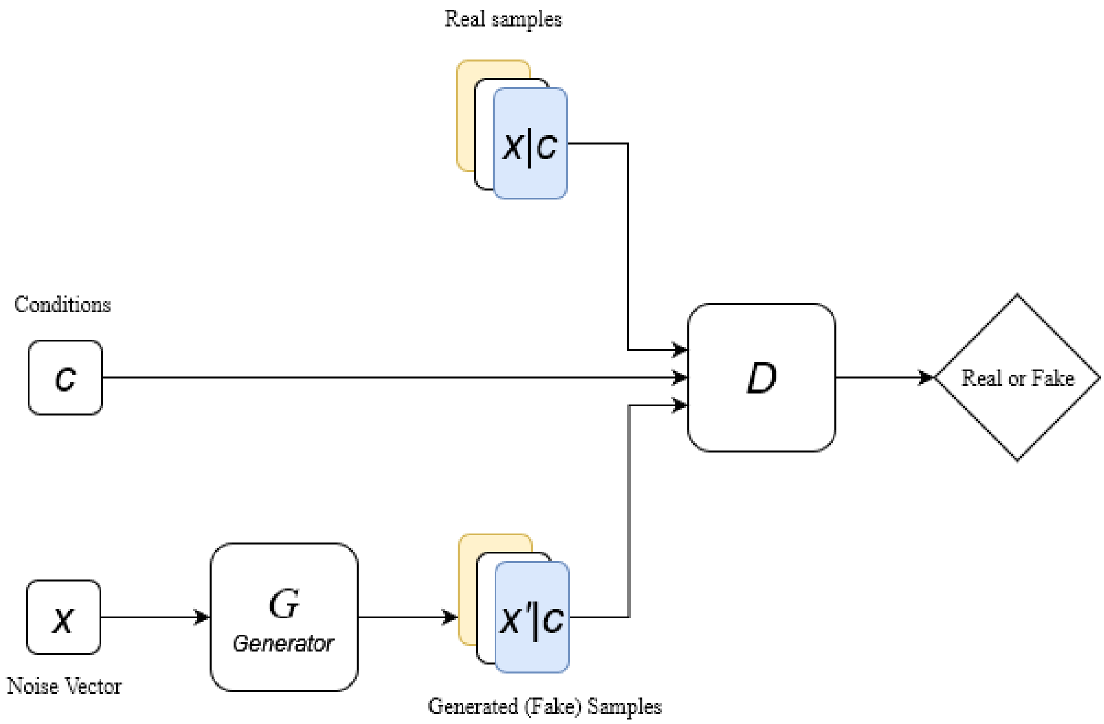

Fig. 2: Architecture of cGANs

1.4 Introduction to Generator Loss Function

1.4.1 Adversarial Loss: The adversarial loss drives the generator to generate images that are indistinguishable from real ones. The primary aim of generator G is to trick the discriminator D. The adversarial loss for the generator can be formulated as:

$$L_{adv}(G) = -E_{z\sim p_z}[\log D(G(z))]$$

where z represents the noise vector fed into the generator, while D(G(z)) denotes the discriminator's assessment of the likelihood that the produced image is authentic.

1.4.2 Content Loss: The content loss ensures that the generated image G(z) is similar to the ground truth image x. This can be measured using L1 or L2 loss. L1 loss is often preferred for its ability to produce sharper images. The content loss is given by:

$$L_{content}(G) = E_{z,x\sim p_{data}}[\|G(x) - x\|_1]$$

where x represents the ground truth, while the L1 norm is denoted by ‖·‖1.

1.4.3 Perceptual Loss: Using a pre-trained network such as VGG, the perceptual loss evaluates high-level features between the generated images and the ground truth images. This helps maintain details and textures. The perceptual loss is formulated as:

$$L_{perceptual}(G) = E_{z,x\sim p_{data}} \sum_i \|\phi_i(G(z)) - \phi_i(x)\|_2^2$$

where ϕ_i denotes the feature map obtained from the pre-trained network's i-th layer while $\|\cdot\|_2$ denotes the L2 norm.

1.5 Motivation

GANs possess a natural ability to understand intricate data patterns, leading to their utilization in a wide range of industries such as computer vision, artistic creation, and enhancing datasets. However, traditional GANs lack control over the content and style of the generated images. Conditional GANs address this limitation by introducing additional information, such as class labels or specific attributes, providing a mechanism to guide the synthesis process. The motivation behind this research lies in the need for advanced generative models that not only produce high-quality images but also allow users to influence and condition the generated content. Whether applied to style transfer, image-to-image translation, or the creation of diverse datasets, conditional image synthesis holds promise for numerous practical applications.

1.6 Objective

This study aims to explore the structure and processes of conditional Generative Adversarial Networks (cGANs) in order to synthesize images, enhance the stability and robustness of cGAN training, explore novel applications and case studies where cGANs excel in generating diverse and realistic images, and analyze the limitations to propose future improvements. Specifically, the study will examine how cGANs use conditional information to guide image generation, employ advanced techniques like spectral normalization to stabilize training, and highlight applications in fields such as medical imaging and creative industries. Additionally, it will address current limitations, such as scaling and control of complex attributes, and propose future directions, including more sophisticated conditioning mechanisms and the use of transformer architectures, to improve cGAN performance.

2. Literature Review

The realm of image deblurring has experienced notable progressions due to the emergence of deep learning methodologies, specifically through Conditional Generative Adversarial Networks (CGANs). This section reviews related works that have employed CGANs and other deep learning methods for image deblurring, providing a context for the current study.

The implementation of deep learning has greatly enhanced the ability to remove blurriness from images. Xu et al. (2014) proposed a deep convolutional network for blind deblurring, demonstrating the potential of CNNs to learn blur patterns directly from data. Similarly, Sun et al. (2015) introduced a CNN for non-uniform motion blur removal, highlighting the robustness of deep learning models.

CGANs have been specifically tailored for image deblurring by conditioning the generation process on the input blurred images. Kupyn et al. (2018) introduced DeblurGAN, a CGAN-based approach that significantly improved deblurring performance by incorporating adversarial and content losses. DeblurGAN leveraged a multi-scale architecture and perceptual loss to enhance the deblurring quality.

Building on this, Kupyn et al. (2019) proposed DeblurGAN-v2, which utilized a deeper ResNet-based generator and incorporated feature pyramid networks to capture blur at multiple scales. The method successfully attained cutting-edge outcomes across various standard datasets, showcasing the efficiency of advanced CGAN architectures for deblurring tasks.

The choice of loss functions significantly impacts the performance of CGANs in image deblurring. Adversarial loss encourages the generation of realistic images, while content loss (e.g., L1 or L2 loss) ensures fidelity to the ground truth. Perceptual loss, as used by Johnson et al. (2016), measures high-level feature differences using pre-trained networks, contributing to visually pleasing results. Evaluation metrics like Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index (SSIM) are frequently utilized to measure the effectiveness of deblurring techniques.

Qian et al. (2018) introduced a pioneering work using CGANs for raindrop removal, proposing a network that combined adversarial training with a perceptual loss. Their model, called "Attentive Generative Adversarial Network," included an attention mechanism to focus on raindrop regions, significantly enhancing the removal process. The perceptual loss, computed through a pre-trained VGG network, guaranteed that the produced images preserved high-level feature coherence with the ground truth images.

Fig 3: Zhang et al.'s images degraded with

raindrops and corresponding ground-truth images

Content loss is commonly calculated by comparing the pixel values of the generated images with those of the ground truth images, ensures that the generator accurately reconstructs image details. The L_1 or L_2 norm is commonly used:

$$\boldsymbol{L_{content}(G) = E_{z,x\sim p_{data}}[\|G(x) - x\|_1]}$$

Zhu et al. (2019) demonstrated that incorporating content loss alongside adversarial loss helps maintain fidelity to the original scene while removing raindrops.

Fig 4: Zhu et al's Loss function comparison on the grounds of performance

This study seeks to make a meaningful contribution to the continuous improvement of image deblurring methods by utilizing CGANs. It aims to provide valuable perspectives and feasible remedies for practical use in real-life scenarios through the synthesis of these progressions.

3. Metrics for Assessing Performance

The assessment of the quality and effectiveness of the outputs generated by Conditional Generative Adversarial Networks (cGANs) heavily relies on performance evaluation metrics. Here are some commonly used metrics:

3.1 Fréchet Inception Distance (FID)

The comparison between the distribution of synthetic samples and authentic samples in the feature space is quantified by the FID score. A lower FID score suggests higher quality and variety in the generated images. The data distribution of these features is represented by a multivariate Gaussian distribution with a mean of μ and a covariance of Σ. The FID between the real images *x* and generated images *g* is computed as:

$$\mathbf{FID} = \left\|\boldsymbol{\mu_r - \mu_g}\right\|^2 + \boldsymbol{T_r}\left(\sum\nolimits_r + \sum\nolimits_g - 2\left(\sum\nolimits_r \sum\nolimits_g\right)^{1/2}\right)$$

Where:

- $\left\|\boldsymbol{\mu_r - \mu_g}\right\|^2$ is the euclidean distance between the means of the real and generated feature distributions.
- $\boldsymbol{T_r}$ denotes the sum of the diagonal elements that is called the trace of a matrix.
- $(\Sigma_r \Sigma_g)^{1/2}$ is the matrix square root of the product of the covariance matrices Σ_r and Σ_g.

3.2 Precision, Recall, and F1 Score for Specific Attributes

Tuijin Jishu/Journal of Propulsion Technology
ISSN: 1001-4055
Vol. 45 No. 3 (2024)

Attribute-specific evaluation metrics for conditional GANs, relevant to tasks like image-to-image translation where certain attributes need to be preserved or modified. High precision, recall, and F1 score indicate successful attribute preservation or modification.

$$\textbf{Precision} = \frac{\textbf{True Positives}}{\textbf{True Positive} + \textbf{False Positives}}$$

$$\textbf{Recall} = \frac{\textbf{True Positives}}{\textbf{True Positive} + \textbf{False Negatives}}$$

$$\textbf{FI Score} = \textbf{2 X} \frac{\textbf{Precision x Recall}}{\textbf{Precision x Recall}}$$

3.3 Peak Signal-to-Noise Ratio (PSNR)

The quality of generated images can be assessed through the comparison of pixel values with real images. A higher PSNR value signifies superior image quality, as PSNR is determined by comparing the Mean Squared Error (MSE) of the original (reference) image with that of the generated (or reconstructed) image.

$$\textbf{MSE} = \frac{1}{mn} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} [I(i,j) - K(i,j)]^2$$

In this context, the variables $I(i,j)$ and $K(i,j)$ denote the pixel values of the initial and produced images, correspondingly, while m and n stand for the dimensions of the images.

$$\textbf{PSNR} = 10 \,.\, log_{10}\left(\frac{MAX^2}{MSE}\right)$$

The maximum pixel value of the image is denoted as MAX. In the case of an 8-bit image, MAX is set at 255.

3.4 Structural Similarity Index (SSIM)

SSIM is formulated to depict alterations in structural details present in an image. Structural details pertain to the arrangements of pixel intensities, which play a vital role in identifying the substance and texture of the image. The human visual system has evolved to effectively perceive structural details, while SSIM strives to replicate this functionality.

$$\textbf{SSIM}(x,y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)}$$

Luminance Comparison Function:

$$l(x,y) = \frac{2\mu_x\mu_y + C_1}{\mu_x^2 + \mu_y^2 + C_1 \mathrm{l}}$$

Contrast Comparison Function:

$$c(x,y) = \frac{2\sigma_x\sigma_y + C_2}{\sigma_x^2 + \sigma_y^2 + C_2}$$

Structural Comparison Function:

$$s(x,y) = \frac{\sigma_{xy} + C_3}{\sigma_x\sigma_y + C_3}$$

3.5 Kernel Inception Distance (KID)

Evaluates the distinction in feature representations of original and generated images through the use of kernelized feature spaces. Lower KID scores signify better image synthesis quality.

Given two sets of feature representations, X (real images) and Y (generated images), the KID is computed as follows:

$$\mathbf{KID} = \frac{1}{m(m-1)}\sum_{i\neq j} k(x_i,x_j) + \frac{1}{n(n-1)}\sum_{i\neq j} k(y_i,y_j) - \frac{2}{mn}\sum_{i,j} k(x_i,y_j)$$

Where:

- $x_i, x_j \in$ X and $y_i, y_j \in$ Y
- $k(a,b)$ is the polynomial kernel, typically defined as $k(a,b) = \left(\frac{a^T b}{d}+1\right)^3$, where d is the dimensionality of the feature representations.

3.6 Diversity Metrics (e.g., Multi-Modality Metrics)

Metrics that assess the diversity of generated images, ensuring that the model produces a broad range of outputs. Higher diversity metrics indicate a more varied set of generated samples.

3.7 User Studies and Human Evaluation

Involves obtaining subjective opinions from human evaluators, often through surveys or comparisons. Human evaluation provides insights into perceptual aspects of generated images that may not be captured by quantitative metrics.

3.8 Domain-Specific Metrics:

Metrics tailored to specific application domains, such as medical imaging or art generation. These metrics capture domain-specific requirements and nuances in the evaluation process.

4. Implementation

Implementation of the work involves various steps and resources which are detailed as follows.

4.1 Dataset

A crucial element in any research is the dataset utilized to conduct the study. The dataset chosen for this research is notably diverse, encompassing a substantial number of both real and AI-generated images. A review of existing literature revealed that previous studies on raindrop datasets and synthetic blur models typically employed a combination of real objects and associated prompts to generate images. In contrast, this work leverages the publicly available Blur Dataset from Kaggle. This dataset consists of 1050 images (350 triplets), where each triplet comprises three photos of the same scene: a sharp image, a defocused-blurred image, and a motion-blurred image. The primary purpose of this dataset is to validate blur detection algorithms. Although it can also be employed for testing image deblurring techniques, the triplets are not "pixel-to-pixel" aligned, thus precluding direct comparison between blurred and sharp images based on PSNR (Peak Signal-to-Noise Ratio) or SSIM (Structural Similarity Index). Nevertheless, the sharp images can still serve as a basis for visual comparison. The Blur Dataset contains thousands of images categorized by different types of blur, including Gaussian blur, motion blur, and defocus blur. Each category includes images that replicate real-world scenarios where blurring commonly occurs. The dataset is meticulously labeled, facilitating researchers in training and testing their models on specific types of blur.

(a) (b) (c)

Fig. 5: Images (a) and (b) exhibit blurriness, while image (c) is characterized by sharpness.

4.2 Methodology Used

The process of enhancing image clarity through the utilization of Conditional Generative Adversarial Networks (CGANs) encompasses a unique approach that merges the capabilities of Conditional GANs with conventional deblurring methods. Here's an outline of the process:

4.2.1 Data Preparation

Dataset Collection: Collect a dataset of blurred and corresponding sharp (ground truth) images.

Preprocessing: Normalize the images and perform any necessary augmentations to increase dataset diversity and robustness.

4.2.2 Network Architecture

Generator: The generator network commonly consists of a convolutional neural network (CNN) designed to receive a blurred image as its input and produce a deblurred image as its output. Popular architectures include U-Net, ResNet, or encoder-decoder networks.

Discriminator: The discriminator is another CNN that distinguishes between real (sharp) images and fake (deblurred) images generated by the generator. It conditions on the input blurred image to better guide the deblurring process.

4.2.3 Training Procedure

Initialization: Initialize the generator and discriminator networks with appropriate weights.

Adversarial Training: Train the generator and discriminator iteratively. For each batch of training data:

- Update the discriminator by maximizing the adversarial loss.
- Update the generator by minimizing the combined loss (adversarial + content + perceptual).

4.2.4 Post-Processing

Refinement: Apply additional techniques to enhance the quality of the deblurred images if necessary (e.g., sharpening filters, noise reduction).

4.2.5 Evaluation

- **Qualitative Evaluation**: Visually inspect the deblurred images to assess their quality.
- **Quantitative Evaluation**: Employing metrics like Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index (SSIM), or Learned Perceptual Image Patch Similarity (LPIPS) is essential in assessing the effectiveness of the model.

4.3 Workflow in Pseudocode

```
# Initialize models
gen_model = GeneratorNetwork()
disc_model = DiscriminatorNetwork()
# Define optimizers
gen_opt = Adam(gen_model.parameters(), lr=learning_rate)
disc_opt = Adam(disc_model.parameters(), lr=learning_rate)
# Training loop
for epoch in range(total_epochs):
    for low_res_image, high_res_image in data_loader:
```

```
    # Update Discriminator
    disc_opt.zero_grad()
    real_pred = disc_model(low_res_image, high_res_image)
    generated_image = gen_model(low_res_image)
    fake_pred = disc_model(low_res_image, generated_image.detach())
    discriminator_loss = -torch.mean(torch.log(real_pred) + torch.log(1. - fake_pred))
    discriminator_loss.backward()
    disc_opt.step()
    # Update Generator
    gen_opt.zero_grad()
    fake_pred = disc_model(low_res_image, generated_image)
    adv_loss = -torch.mean(torch.log(fake_pred))
    l1_content_loss = F.l1_loss(generated_image, high_res_image)
    vgg_perceptual_loss = perceptual_loss_fn(generated_image, high_res_image)
    generator_loss = adv_loss + l1_content_loss + vgg_perceptual_loss
    generator_loss.backward()
    gen_opt.step()
  print(f"Epoch [{epoch}/{total_epochs}], Generator Loss: {generator_loss.item()}, Discriminator Loss: {discriminator_loss.item()}")
```

4.4 Algorithm

A dataset comprising real and AI-generated images is employed, and it is divided into separate training and testing sets. The training set is composed of 70% blurred images, while the testing set contains the remaining 30% sharp images. Our methodology entails training various generator models using different loss functions, namely Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index (SSIM). These metrics are utilized to assess the performance of each model throughout both the training and testing stages.

Following training, a comprehensive comparative analysis is conducted to assess the efficacy of each loss function in enhancing image deblurring quality. This evaluation aims to identify which approach achieves superior results, offering valuable insights into the effectiveness of PSNR and SSIM in the context of image restoration tasks.

This research methodology not only contributes to advancing the understanding of image deblurring techniques but also provides practical guidance for selecting optimal loss functions in similar applications.

4.5 Approach

The working of the project is explained in a simplified manner using the following figure:

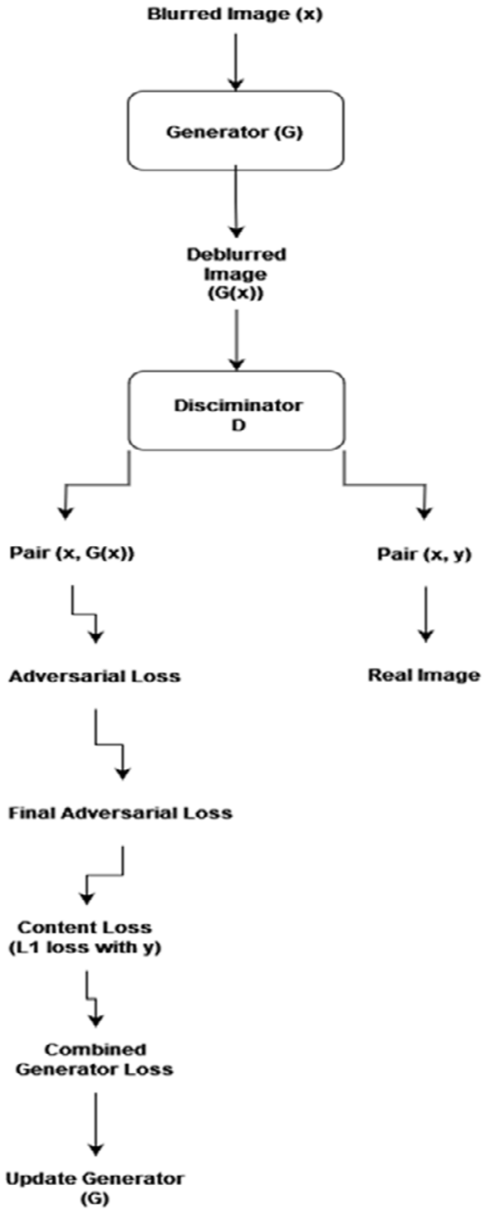

Fig. 6: Research work flow diagram

5. Analysis of experimental results

The analysis begins with an evaluation of how various loss functions influence image blur. Subsequently, the algorithm's effectiveness in removing blur is compared with that of other algorithms using both synthetic blur datasets and real-world blurred images.

5.1 Evaluating the Effectiveness of Different Loss Functions

To assess the impact of various loss functions on mitigating blur, this study evaluates different algorithms based on their respective loss functions. Table 1 presents the mean PSNR and SSIM values from 100 experimental groups subjected to iterative training. The results indicate that the LL1 loss function consistently yields higher PSNR values, indicative of its efficacy in reducing noise and distortion associated with blur. Conversely, the LP loss function demonstrates a capacity to enhance SSIM values, thereby effectively preserving the overall structural integrity of the images. These findings underscore the critical role of selecting appropriate loss functions tailored to optimize specific aspects of image restoration tasks, particularly in the context of blur rectification.

Loss Function	Indices	Test100
L_{CGAN}	**PSNR**	26.3250
	SSIM	0.6254
$L_{CGAN} + L_{L1}$	**PSNR**	30.2841
	SSIM	0.6928
$L_{CGAN} + L_{Lp}$	**PSNR**	28.2946
	SSIM	0.8577
$L_{CGAN} + L_{L1} + L_{Lp}$	**PSNR**	32.7554
	SSIM	0.8295

Table 1: Analysis of Loss Functions through PSNR and SSIM Measurements

Fig. 7 depicts the contrast in processing outcomes for a sample image from the blur dataset when employing various sets of loss functions. Specifically, when using L_{CGAN} alone for blur removal, significant distortions in image texture structure and color are observed. Introducing L1 helps mitigate some of these issues associated with L_{CGAN}, resulting in fewer artifacts in the blur removal effect. However, residual fuzziness and partial image distortion may still occur, accompanied by oversaturation in some reconstructed images. Incorporating the perceptual loss function L_P further enhances clarity, thereby generating clearer images with reduced blurring effects. Consequently, the combination of $L_{CGAN} + L_{L1} + L_P$ not only eliminates artifacts but also preserves more image details, leading to the creation of more realistic and sharper images in the context of blur removal.

Fig. 7: Performance comparison of loss function

6. Conclusion and Future Prospect

After carrying out the extensive study, the following conclusion is drawn. In addition to the conclusion, some future advancement regarding the research are also presented in detail.

6.1 Conclusion

This idea behind the project is to deblur the images using various loss functions such as CGAN. The application of Conditional Generative Adversarial Networks (CGANs) in image deblurring has made substantial advancements, revolutionizing the field with its ability to produce high-fidelity, deblurred images that surpass traditional methods. CGANs utilize adversarial training, in which the generator and discriminator networks are trained to work against each other. This approach leads to the production of high-quality and detailed images from their blurry counterparts. The competitive nature of this configuration motivates the generator to produce images that are challenging for the discriminator to differentiate from authentic, high-resolution images, ultimately improving the overall quality of the deblurred results. A key factor contributing to the success of CGANs in image deblurring is the effective design and integration of various loss functions. The adversarial loss, fundamental to the GAN framework, ensures that the generated images are realistic. Meanwhile, content loss, typically computed as the pixel-wise L_1 or L_2 norm between the generated and base supplied images, helps maintain the fidelity of the original scene by accurately reconstructing image details.

6.2 Future Work

The research on image deblurring using CGANs has made substantial strides, offering promising solutions to a long-standing problem in computer vision. By leveraging adversarial training, sophisticated loss functions, and advanced network architectures, CGAN-based approaches have set new benchmarks in image restoration quality. Future advancements will continue to refine these techniques, addressing current challenges and expanding the applicability of CGANs in real-world scenarios. The ongoing evolution of CGANs holds great promise for the future of image deblurring and broader image restoration tasks, driving forward both academic research and practical applications.

7. Conflict of Interest Statement

The publication of this manuscript is free from any conflict of interest.

References

[1] Trippe, T., Genzel, M., Macdonald, J., & März, M. (2022). Let's enhance: A deep learning approach to extreme deblurring of text images. *arXiv preprint arXiv:2211.10103*.

[2] Xiang, Y., Zhou, H., Li, C., Sun, F., Li, Z., & Xie, Y. (2024). Application of Deep Learning in Blind Motion Deblurring: Current Status and Future Prospects. *arXiv preprint arXiv:2401.05055*.

[3] Qian, R., Tan, R. T., Yang, W., Su, J., & Liu, J. (2018). Attentive generative adversarial network for raindrop removal from a single image. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 2482-2491).

[4] Zhu, M., Fang, C., Du, H., Qi, M., & Wu, Z. (2019). Application research on improved CGAN in image raindrop removal. *The Journal of Engineering*, *2019*(22), 8404-8408.

[5] Isola, P., Zhu, J. Y., Zhou, T., & Efros, A. A. (2017). Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1125-1134).

[6] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., ... & Bengio, Y. (2014). Generative adversarial nets. *Advances in neural information processing systems*, *27*.

[7] Wang, Z., Chen, J., & Hoi, S. C. (2020). Deep learning for image super-resolution: A survey. *IEEE transactions on pattern analysis and machine intelligence*, *43*(10), 3365-3387.

[8] Frants, V., Agaian, S., & Panetta, K. (2023). QCNN-H: Single-image dehazing using quaternion neural networks. *IEEE Transactions on Cybernetics*, *53*(9), 5448-5458.