Minimising Gaussian noise from real time CCTV images using GAN

[1]Adyapadi Suraj, [2]Akshay Prashant Hegde, [3]Pranava Aithal K S, [4]Manoj Kumar M V, [5]Prashanth B S
[1]*Department of Information Science & Engineering,* NITTE-Meenakshi, Bangalore 560064, Karnataka, India
adyapadisuraj@gmail.com, akshayphegde2002@gmail.com, pranava990@gmail.com, manoj.kumar@nmit.ac.in, prashanth.bs@nmit.ac.in

***Abstract*—This paper describes a system that uses generative adversarial networks (GANs) to eliminate gaussian noise from CCTV of images. Unwanted noise like gaussian noise frequently degrades the quality of images and makes them more difficult to interpret. In our approach, a discriminator network is used to direct the training of a generator network, whose job it is to produce denoised images from noisy inputs. This framework includes operations like picture enhancement, noise reduction, and evaluation with metrics like the structural similarity index (SSIM) and peak signal-to-noise ratio (PSNR). The resultant denoised images show enhanced visual quality and have potential uses in image analysis and computer vision.**

***Keywords*—Noise Removal, Generative Adversarial Networks (GANs), Visual Quality Improvement, Image Interpretation, Discriminator Network, Generator Network, PSNR (Peak Signal-to-Noise Ratio), SSIM (Structural Similarity Index), Image Processing Framework, Real-world Scene Images, Image Analysis Potential.**

I. Introduction

The ubiquity of undesired noise in real-world scene images presents an enduring obstacle in today's digital landscape. Traditional denoising methods often struggle to distinguish between noise and essential image elements, resulting in the loss of critical information or the introduction of aberrations. Generative Adversarial Networks (GANs) emerge as a promising solution to this dilemma. This initiative seeks to explore the capacity of GANs to enhance the visual quality of real-world scene images by mitigating unwanted noise.

Real-world scene images frequently suffer from the intrusion of unwanted noise, degrading visual clarity and complicating interpretation. In response to this challenge, we propose a framework harnessing Generative Adversarial Networks (GANs) to diminish undesired noise and improve the visual fidelity of real scene images.

Our framework comprises two principal components: noise reduction and image enhancement. A discriminator network distinguishes between pristine and generated images, while a generator network undergoes adversarial training to transform noisy images into clean renditions.

Ongoing research efforts endeavor to reconstruct high-resolution images from low-resolution inputs. Convolutional Neural Networks (CNNs), especially Deep Learning Architectures, have achieved remarkable progress in super-resolution (SR) techniques, building upon the pioneering work of Super Resolution Convolution Neural Network (SRCNN). However, the widespread use of bicubic downsampling kernels deviates from authentic degradation processes, hindering the practical applicability of such methods.

Blind super-resolution methods are typically classified into two main categories: explicit and implicit modeling approaches, both addressing the challenge of enhancing low-resolution images affected by complex and unknown factors. Explicit modeling relies on predefined degradation models like blur, downsampling, noise, and compression, aiming to approximate real-world degradation processes. However, these models often struggle to accurately capture the diverse and intricate nature of real-world degradation, limiting their effectiveness. In contrast, implicit modeling techniques leverage data-driven approaches such as distribution learning and Generative Adversarial Networks (GANs) to learn complex degradation patterns directly from data.

Gaussian noise, which has a constant power spectral density and is sometimes called white noise, has a few unique characteristics. In essence, it's a collection of haphazard tiny blips or specks incorporated into the original signal. It's like a series of unexpected events dispersed throughout, as each blip exists independently of the others. It is present almost everywhere, including in CCTV image and radio broadcasts. Since they are additive in nature, the noisy signal is usually produced by adding them to the original signal. It is a good model for many kinds of random disturbances found in both natural and artificial systems because of this additive feature. Because of their widespread use in practical settings and well-understood statistical characteristics, they serve as a common and helpful model in signal processing and image processing applications.

II. Literature Survey

Since the inception of SRCNN, notable progress has been achieved in the realm of image super-resolution. Generative adversarial networks (GANs) have gained traction as a favored method for loss supervision, aiming to align solutions more closely with the natural image distribution and yield visually appealing outcomes. Nonetheless, many existing techniques lean on bicubic downsampling kernels and often grapple with generating precise outputs when confronted with real-world images. Recent strides in image restoration methodologies

978-1-6654-9790-9/22/$31.00 ©2022 IEEE

have begun integrating reinforcement learning and GANs to confront these hurdles.

Blind super-resolution (SR) has attracted considerable attention in the research domain. One category of approaches centers on explicit representations of degradation, typically encompassing two primary facets: degradation prediction and conditional restoration. These methods may execute the two facets independently or iteratively, relying heavily on predetermined representations of degradation, such as degradation types and levels. Nonetheless, these approaches frequently overlook intricate real-world degradations and may introduce artifacts if degradation estimations prove inaccurate.

An alternative approach entails acquiring or generating training pairs that closely mirror real data, followed by training a unified network to tackle blind super-resolution. Obtaining such training pairs typically involves dedicated cameras and demands meticulous alignment. Alternatively, these pairs can be gleaned from unpaired data using cycle consistency loss. Another avenue involves synthesizing the pairs by estimating blur kernels and extracting noise patches. However, the data collected is constrained to degradation associated with specific cameras, limiting its applicability to other real-world images. It proves challenging to accurately capture and analyze subtle deteriorations using data not directly paired with original images, often yielding unsatisfactory results.

Image denoising techniques employing Generative Adversarial Networks (GANs) have emerged as powerful tools across diverse domains, addressing challenges posed by noise in various imaging modalities. One such application discussed in Zhong et al.'s paper [6] focuses on blind denoising of fluorescence microscopy images, crucial in life sciences but often afflicted by strong noise due to formation and acquisition constraints. Their proposed blind global noise modeling denoiser (GNMD), utilizing a GAN to simulate image noise globally, outperforms existing methods in suppressing background noise, thereby facilitating downstream image segmentation tasks. Another notable contribution by Zhiping et al. [5] introduces a novel GAN architecture for texture-preserving image denoising. Their approach involves a generator network trained using a newly devised loss function to accurately measure the disparity between the data distribution of clean and denoised images, resulting in superior denoising performance compared to other methods.

In a distinct application domain, Alsaiari et al. [1] propose a GAN-based solution for noise reduction in animation studio-rendered 3D scenes. By leveraging neural networks, particularly GANs, rendering time is drastically reduced while maintaining photorealistic image quality, thus enhancing efficiency in animation production pipelines. Meanwhile, addressing challenges in medical imaging, Yang et al. [4] tackle noise in low-dose CT images using a GAN-based approach with Wasserstein distance and perceptual loss. Their method demonstrates promising outcomes, with the Wasserstein distance enhancing GAN performance and perceptual loss preserving critical image details, thereby improving diagnostic accuracy.

Lastly, Tian et al. [2] focus on denoising disruptive noise in Magnetic Resonance Imaging (MRI) images using conditional GANs. Their method, employing a CNN to separate real and fake image pairs, coupled with an adversarial learning-based convolutional encoder-decoder generator, effectively reduces MRI image noise. Experimental results on synthetic and clinical MRI datasets illustrate the method's high structural similarity and stability, even at higher noise levels compared to conventional methods. These diverse applications underscore the versatility and effectiveness of GAN-based image denoising techniques in addressing noise-related challenges across different domains.

Additionally, Wang et al. [3] highlight the shortcomings of many blind super-resolution techniques in addressing general real-world degraded images, despite their success in restoring low-resolution images with unknown and intricate degradations. The authors propose a practical restoration approach, termed Real-ESRGAN, which leverages the potent capabilities of ESRGAN and is trained using purely synthetic data. They employ high-order degradation modeling to emulate complex real-world degradation scenarios, taking into account prevalent ringing and overshoot artifacts during synthesis. To enhance discriminator performance and stabilize training dynamics, they adopt a U-Net discriminator with spectral normalization. Real-ESRGAN surpasses previous works in visual performance on real datasets, offering efficient methods to generate training pairs as needed.

III. Proposed Methodology

A. *Traditional Degradation Model*

Blind super-resolution is the process of enhancing the resolution of an image without prior knowledge of the degradation model or the low-resolution input[3]. The conventional degradation model is commonly employed to generate the low-resolution input. Typically, the original image y is initially convolved with a blur kernel k. Next, a downsampling operation is executed using a scale factor of r. The low-resolution x is acquired through the addition of noise n. Additionally, PNG compression is implemented due to its extensive usage in real-world images.

$$x = D(y) = [(y * k)\downarrow_r + n]_{\text{PNG}} \tag{1}$$

In equation (1), D represents the degradation process. Next, we will briefly review these frequently encountered deteriorations.

Blur: The `cv2.GaussianBlur()` function in OpenCV applies Gaussian smoothing to an image using a Gaussian filter kernel. The function takes the following parameters:

- `src`: The input image.
- `ksize`: The kernel size. This parameter specifies the width and height of the kernel. It must be an odd number (e.g., 3, 5, 7, etc.).

- `sigmaX`: The standard deviation of the Gaussian kernel along the X-axis.
- `sigmaY` (optional): The standard deviation of the Gaussian kernel along the Y-axis. If not specified, it defaults to `sigmaX`.
- `borderType` (optional): Specifies how to handle border pixels. Default is `cv2.BORDER_DEFAULT`.

The Gaussian kernel $G(x, y)$ is defined as:

$$G(x, y) = \frac{1}{2\pi\sigma_X\sigma_Y} e^{-\frac{x^2+y^2}{2\sigma_X^2}} \tag{2}$$

For the above mentioned equation (2), σ_X and σ_Y are the standard deviations along the X and Y axes, respectively. The parameter k_{size} determines the size of the kernel, and it must be an odd integer.

The `GaussianBlur` function performs convolution between the input image I and the Gaussian kernel G to produce the smoothed image I_{smooth}. Mathematically, the operation can be represented as:

$$I_{\text{smooth}}(x, y) = \sum_{i=-k}^{k} \sum_{j=-k}^{k} I(x + i, y + j) \cdot G(i, j) \tag{3}$$

In the above equation (3), (x, y) represents the pixel coordinates in the image, and (i, j) represents the coordinates in the Gaussian kernel.

Degradation: Gaussian blur kernels are frequently employed to model blur degradation in images. However, it is suggested that while Gaussian blur kernels are prevalent, they may not precisely capture real camera blur. To broaden the scope of kernel shapes, we can also utilize generalized Gaussian blur kernels in conjunction with a plateau-shaped distribution.

The probability density functions (PDFs) of these kernels are delineated as follows:

- For the generalized Gaussian blur kernels: $\frac{1}{N} \exp\left(\frac{1}{2}(CT-1)^C\right)$, where C is the shape parameter.
- For the plateau-shaped distribution: $\frac{1}{N} \frac{1}{1+(CT-1)^C}$, also depending on the shape parameter.

Noise: The code implements a function add_gaussian_noise that adds Gaussian noise to an input image. Gaussian noise is a type of statistical noise characterized by its Gaussian (normal) probability distribution. It is commonly encountered in various imaging scenarios and often represents random fluctuations in pixel values.

Mathematically, Gaussian noise is typically represented as:

$$\text{Gaussian Noise} = \mu + \sigma \times \epsilon \tag{4}$$

In the equation (4) mentioned above:

- μ (mu) represents the mean of the Gaussian distribution, controlling the central tendency of the noise.
- σ (sigma) represents the standard deviation, controlling the spread or dispersion of the noise.
- ϵ (epsilon) is a random sample drawn from a standard normal distribution with a mean of 0 and a standard deviation of 1. This random sample introduces variability and randomness into the noise.

The function add_gaussian_noise accepts an input image and generates Gaussian noise with the specified mean (μ) and standard deviation (σ). It then adds this noise to the input image to create a noisy version. Finally, the noisy image is clipped to ensure pixel values remain within the valid range [0, 255] and converted to an unsigned 8-bit integer format.

This process helps simulate the effect of Gaussian noise commonly observed in real-world images, enabling researchers to evaluate the performance of denoising algorithms and other image processing techniques under realistic conditions.

Resize: Downsampling plays a crucial role in generating low-resolution images, serving multiple purposes in various image processing tasks. Firstly, it facilitates the creation of low-resolution inputs for tasks such as super-resolution. In this context, high-resolution images are converted into lower-resolution counterparts to be processed by algorithms. Additionally, downsampling can serve as a form of data augmentation, expanding training datasets and bolstering model robustness. Moreover, it enables more efficient processing by reducing computational complexity and memory requirements, which is particularly advantageous in real-time applications. Furthermore, downsampling can simulate real-world scenarios with naturally low-resolution images, aiding in algorithm development and testing under realistic conditions.
To encompass a broader spectrum of diverse and intricate resizing effects, we harness the capabilities of the cv2.resize() function from the OpenCV library. This function enables us to execute resizing operations using various interpolation methods, including area, bilinear, and bicubic interpolations. We deliberately omit nearest-neighbor interpolation due to potential misalignment issues, thereby ensuring a more robust and precise resizing process.

PNG Compression: PNG (Portable Network Graphics) is a widely used format for lossless compression of digital images. Unlike JPEG compression, which is lossy, PNG compression preserves all image data without introducing artifacts. This compression method is particularly effective for images with sharp edges and areas of uniform color.

When saving an image in PNG format, the image data is encoded using a predictive coding method that takes advantage of similarities between adjacent pixels. This allows PNG files to achieve high compression ratios without sacrificing image quality.

The quality of PNG compressed images is not controlled by a quality factor like in JPEG compression. Instead, PNG compression aims to minimize file size while preserving image fidelity as much as possible.

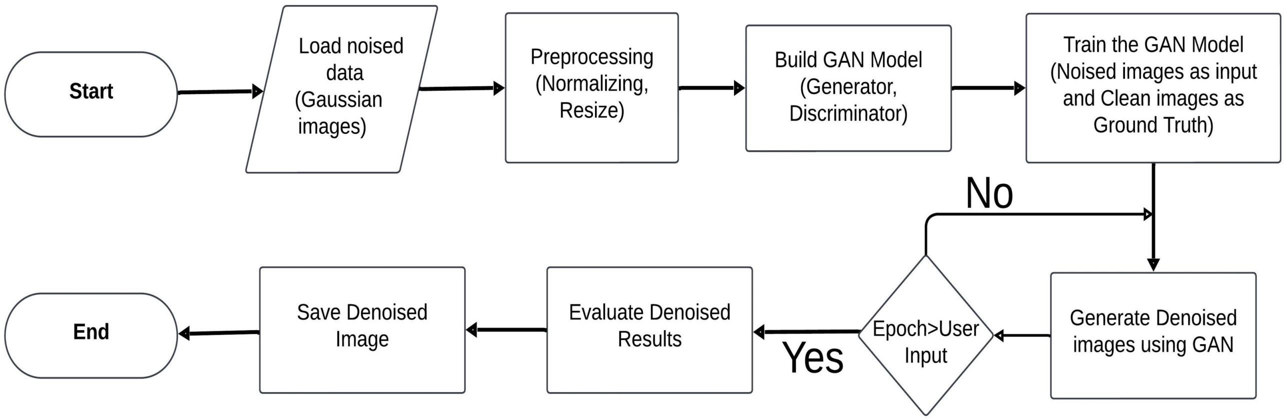

Figure 1. Architecture diagram

In our implementation, we utilize the PNG format to save the noisy image after adding Gaussian noise and resizing it. This ensures that the image retains its quality without introducing compression artifacts commonly associated with JPEG compression.

We employ the Python library `Pillow` to save images in PNG format. The code snippet below demonstrates how to save an image in PNG format using `Pillow`:

```
from PIL import Image
```

By using PNG compression, we ensure that the image maintains its quality throughout various processing steps, making it suitable for further analysis and usage in applications where preserving image fidelity is crucial.

B. Model of High-order Degradation

The classical degradation model, based on first-order modeling, includes only a limited number of fundamental degradation. However, real-world degradation processes are more varied and complex, involving multiple procedures such as camera imaging, image editing, and internet transmission. For example, an image captured with a mobile phone may already contain imperfections like blurriness, sensor noise, low detail, and compression artifacts. Further editing operations like sharpening and resizing can introduce additional artifacts. Uploading images to social media platforms and digital transmission also contribute to degradation.

To address the limitations of the first-order model, we can use a higher-order degradation model. In an n-order model, degradation processes are repeated n times, each adhering to the classical degradation model with varying hyperparameters. The ”high-order” aspect refers to the number of times a particular operation is executed. The passage clarifies that not all shuffled degradation are necessary for effective degradation modeling.

$$x = \mathcal{D}''(y) = (\mathcal{D}_n \circ ... \circ \mathcal{D}_2 \circ \mathcal{D}_1)(y) \tag{5}$$

Equation (5) is presented to illustrate the high-order degradation process, where x represents the degraded image obtained from the original image y through multiple degradation operations $\mathcal{D}_1, \mathcal{D}_2, ..., \mathcal{D}_n$.

It is acknowledged that the enhanced high-order degradation process may not encompass the entirety of real-world degradation scenarios but serves to expand the capabilities of previous blind super-resolution methods by enhancing the synthesis of training data.

C. Artifacts featuring ringing and overshoot

Two common types of artifacts often encountered in digital images are Ringing artifacts and Overshoot artifacts.

Ringing artifacts manifest as visual distortions, appearing as spurious edges or halos near sharp transitions within an image. These artifacts frequently present as bands or ”ghosts” encircling edges, diminishing the visual quality of the image. They typically arise from processes like image sharpening or other forms of image enhancement.

In contrast, Overshoot artifacts are characterized by an exaggerated jump or overshoot at the transition between different image regions. Often accompanying ringing artifacts, they exacerbate visual distortions near edges. Overshoot artifacts stem from the amplification of high-frequency components during image processing, such as sharpening operations or compression algorithms like JPEG compression.

Both types of artifacts can detrimentally affect the clarity and fidelity of an image, underscoring the importance of understanding and minimizing their occurrence, particularly in applications where image quality is paramount, such as photography, medical imaging, and computer vision.

To replicate these artifacts for training image pairs, the sinc filter is employed. The sinc filter, an idealized filter, attenuates high frequencies. The mathematical expression of the sinc filter kernel is as follows:

$$k(i,j) = \frac{\omega_c}{2\pi\sqrt{i^2+j^2}} J_1(\omega_c\sqrt{i^2+j^2}) \tag{6}$$

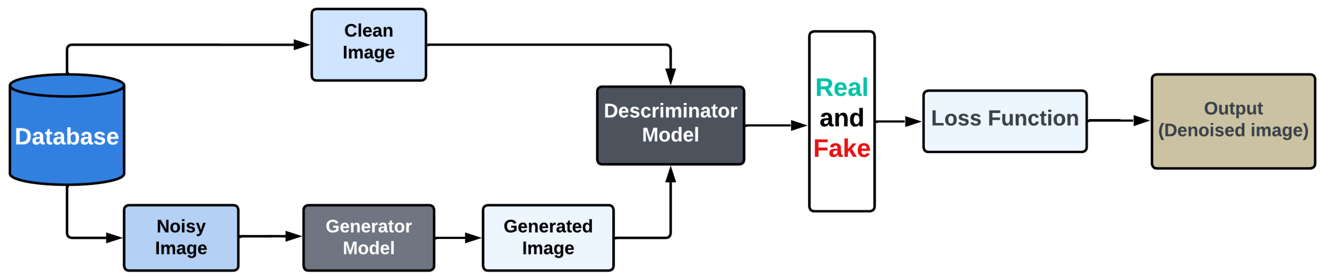

Figure 2. Architecture diagram

In this equation (6), (i, j) represents the coordinates of the kernel,ω_c denotes the cutoff frequency, and J_1 represents the first-order Bessel function of the first kind.

Sinc filters are utilized in two stages: during the blurring process and in the final synthesis step. The order of the final sinc filter and PNG compression is randomly alternated to encompass a broader degradation space. This variation is essential because some images may initially experience oversharpening (resulting in overshoot artifacts) before undergoing PNG compression, while others may undergo PNG compression first, followed by a sharpening operation.

D. Training and Networks

SRGAN generator: We employ the same generator, referred to as the SR network, utilized in SRGAN, which consists of multiple residual-in-residual dense blocks (RRDB). Additionally, we expand the original x4 SRGAN architecture to accommodate super-resolution with scale factors of x2 and x1.

To address the high computational demands of SRGAN, we initially utilize pixel-unshuffle, which is the inverse of pixel shuffle. This process reduces the spatial dimensions and increases the channel dimensions of the input data before it is fed into the main SRGAN architecture. This strategy enables most computations to be executed in a lower resolution space, leading to reduced GPU memory usage and more efficient utilization of computational resources.

The U-Net discriminator: Real-SRGAN aims to tackle a wider spectrum of image degradation compared to SRGAN, rendering the original discriminator design of SRGAN inadequate. Specifically, the Real-SRGAN discriminator requires enhanced capabilities to discern and categorize intricate training outputs. Additionally, the system must not only recognize global fashion trends but also provide precise and detailed analysis of gradient variations in particular textures.

Drawing inspiration from previous studies, we enhance the VGG-style discriminator in SRGAN by implementing a U-Net architecture with skip connections. The U-Net architecture allows for more comprehensive analysis of image degradation and restoration by incorporating skip connections. These skip connections facilitate the flow of information from the early layers to the later layers and vice versa, enabling the discriminator to capture both global and local features effectively.

Moreover, by drawing inspiration from previous studies, the Real-SRGAN discriminator aims to provide enhanced capabilities to discern and categorize intricate training outputs. This includes recognizing global fashion trends and providing precise analysis of gradient variations in particular textures.

The adoption of the U-Net architecture in the Real-SRGAN discriminator enables it to tackle a wider spectrum of image degradation compared to SRGAN. The complex deteriorations in images, such as noise, blur, and artifacts, can be effectively identified and categorized by the discriminator, leading to improved performance in image super-resolution tasks. Overall, the incorporation of the U-Net architecture enhances the discriminator's ability to provide detailed and accurate assessments of image quality, contributing to the overall success of the Real-SRGAN system.

Training process: The process is comprised of two distinct stages. First, we train a model that is optimized specifically for Peak Signal-to-Noise Ratio (PSNR) by utilizing the L1 loss function. This model is named Real-SRNet. Afterwards, we use the trained PSNR-optimized model as an initial state for the generator. Subsequently, we train the Real-SRGAN model by employing a fusion of the L1 loss, perceptual loss, and Generative Adversarial Network (GAN) loss.

IV. Dataset Description

The DIV2K dataset is widely recognized for its high-resolution 2K images and plays a crucial role in the field of image super-resolution research. As a crucial element in our project, we have incorporated the DIV2K dataset to train and assess the effectiveness of our super-resolution model.

We chose to include the DIV2K dataset due to its wide variety of images and its ability to effectively train our super-resolution model. The DIV2K dataset provides a vast collection of high-resolution images, making it an invaluable

TABLE I
DATASET DESCRIPTION TABLE

Dataset	Description
Number of Images	50
Image Type	Artistic/pointillistic portrait images
Image Contents	Faces/Full-body portraits of different people (men, women, children)
Image Style	Composed of Gaussian Noise on the original image
Intended Use	Image processing, style transfer, or generative art tasks.
Image Dimension	256 x 256
File Format	PNG

resource for the development and improvement of image enhancement algorithms.

The DIV2K dataset consists of a significant quantity of high-resolution images, typically with a resolution of 2K. The dataset is commonly partitioned into subsets, such as DIV2K train and DIV2K valid, and is available in a standardized file format, usually PNG, to ensure compatibility with different image processing frameworks and tools.

Prior to employing the DIV2K dataset in our project, we performed essential preprocessing procedures to enhance its appropriateness for our particular application. These tasks may have involved adjusting the size of images, implementing data augmentation techniques, or dividing the dataset into training and validation sets to enhance the efficiency of model training and evaluation.

During our project, we made extensive use of the DIV2K dataset to train our super-resolution model. We developed customized methodologies to utilize specific portions of the dataset for training and validation, guaranteeing the creation and assessment of a strong model. In addition, we utilized precise evaluation metrics to accurately assess the performance of our model.

The incorporation of the DIV2K dataset was pivotal in influencing the results of our project. It served as a fundamental source for the development and validation of models, ultimately leading to the improvement and enhancement of our super-resolution algorithms. We are grateful for the availability of the DIV2K dataset and acknowledge its significant contribution to the success of our project.

V. RESULT

The Graphical User Interface (GUI) developed for image denoising leverages the tkinter library, a popular Python toolkit for creating GUI applications. With its intuitive and customizable features, tkinter provides the backbone for constructing the user interface, enabling seamless integration of image processing functionalities. The GUI itself comprises several key elements, including feature boxes for selecting and processing images, as well as instructional guidance for users. Within the GUI, users can interact with the ”Select Image” feature box to choose an image from their local file system, facilitated by the tkinter filedialog module. Upon selection, the chosen image is displayed within the GUI, and the ”Process Image” button becomes active, thanks to tkinter's event-driven programming paradigm. Subsequently, users can initiate the denoising process by clicking the ”Process Image” button, which triggers the application of denoising techniques using a pre-trained model. The denoised image is then rendered in a new window using the PIL (Python Imaging Library) module for image display. Additionally, users have the option to save the denoised image locally, facilitated by the tkinter filedialog module. Overall, the GUI offers a user-friendly platform for denoising images, powered by tkinter's versatility and the seamless integration of image processing functionalities within the provided code.

We present a comprehensive analysis of the denoising process, including visual representation and quantitative evaluation. Specifically, we have displayed the original image, the corresponding noised image, and the resulting denoised image side by side for comparison. Additionally, we have calculated the Peak Signal-to-Noise Ratio (PSNR) between the original and denoised images to provide a quantitative measure of the denoising effectiveness. By presenting both visual and numerical assessments, we aim to provide a thorough understanding of the denoising process and its outcomes to the evaluator.

VI. LIMITATION

Real-SRGAN, while capable of restoring the majority of real-world images, does have certain constraints.

1) Certain restored images, particularly those depicting buildings and indoor scenes, may exhibit distorted lines as a result of aliasing problems.
2) GAN training results in the presence of undesirable artifacts in certain samples.
3) It is unable to eliminate complex deteriorations that occur outside of the expected range in real-world scenarios. Furthermore, it has the potential to magnify these artifacts.

These limitations significantly affect the practical implementation of Real-SRGAN and require immediate attention in future research.

VII. FUTURE RESEARCH PROSPECTS

Research opportunities utilising Generative Adversarial Networks (GANs) to eliminate or reduce Gaussian noise in real-time CCTV image are extremely promising. One approach might be to improve GAN architectures so they can better adjust to the intricate and dynamic structure of CCTV image. This would involve taking advantage of deep learning techniques to improve noise reduction capabilities while maintaining important details. Investigating cutting-edge training techniques like meta-learning and self-supervised learning could improve these models' resilience and applicability in a variety of surveillance scenarios and environmental settings.

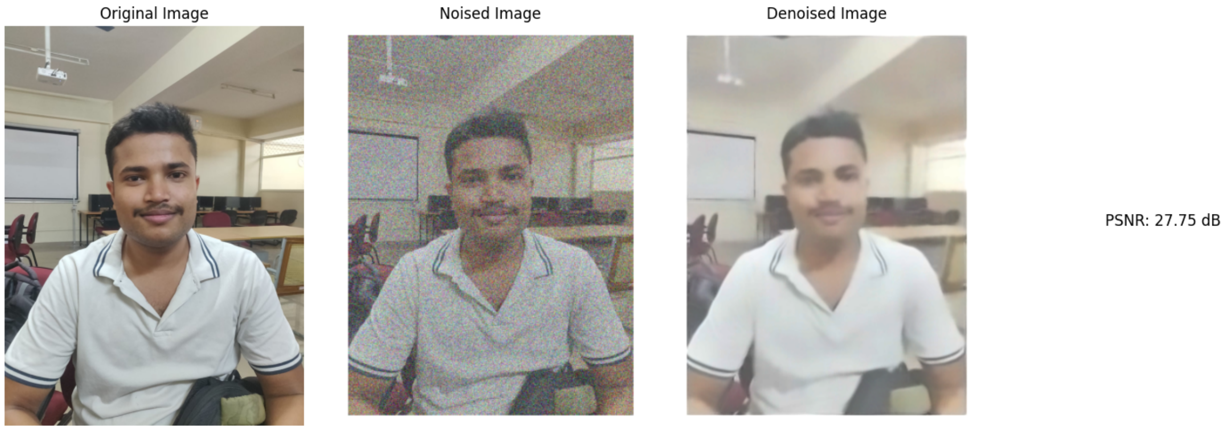

Figure 3. Final Result

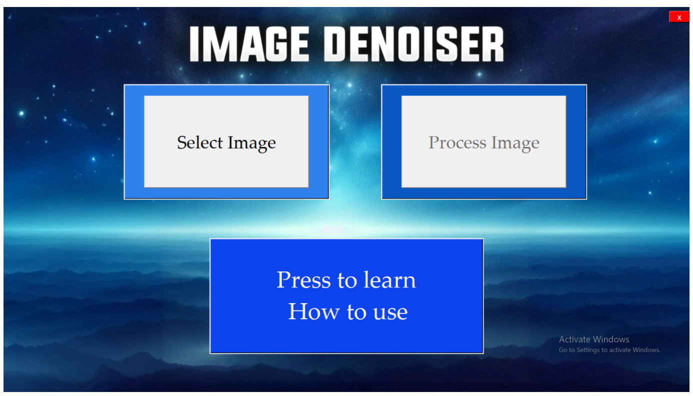

Figure 4. GUI Landing Page

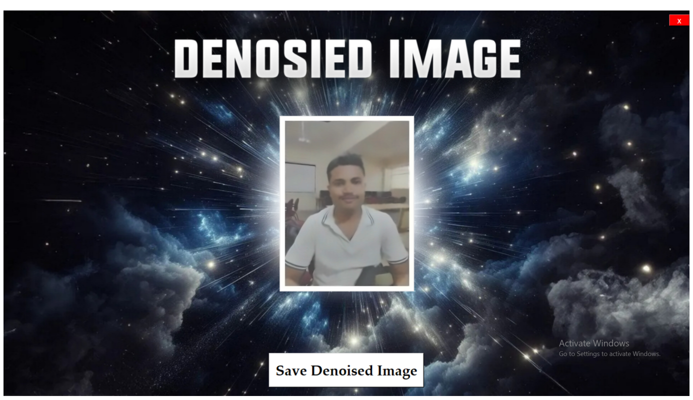

Figure 5. GUI Result Page

Furthermore, the incorporation of domain-specific knowledge, such as the comprehension of common noise patterns in security image, may facilitate the creation of more focused and effective noise reduction algorithms.

VIII. Conclusion

This paper concentrates on training Real-SRGAN for blind super-resolution in real-world scenarios exclusively using synthetic training pairs. To introduce more realistic degradation effects in images, the proposal is to employ a high-order degradation method alongside sinc filters to simulate common artifacts such as ringing and overshoot. Furthermore, a U-Net discriminator is utilized, incorporating spectral normalization regularization. This method augments the discriminator's capabilities and guarantees more stable training dynamics. When trained using synthetic data, Real-SRGAN possesses the ability to enhance the level of detail in real-world images while simultaneously eliminating bothersome artifacts.

References

[1] Abeer Alsaiari, Ridhi Rustagi, Manu Mathew Thomas, Angus G Forbes, et al. Image denoising using a generative adversarial network. In *2019 IEEE 2nd international conference on information and computer technologies (ICICT)*, pages 126–132. IEEE, 2019.

[2] Miao Tian and Kaikai Song. Boosting magnetic resonance image denoising with generative adversarial networks. *IEEE Access*, 9:62266–62275, 2021.

[3] Xintao Wang, Liangbin Xie, Chao Dong, and Ying Shan. Real-esrgan: Training real-world blind super-resolution with pure synthetic data. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1905–1914, 2021.

[4] Qingsong Yang, Pingkun Yan, Yanbo Zhang, Hengyong Yu, Yongyi Shi, Xuanqin Mou, Mannudeep K Kalra, Yi Zhang, Ling Sun, and Ge Wang. Low-dose ct image denoising using a generative adversarial network with wasserstein distance and perceptual loss. *IEEE transactions on medical imaging*, 37(6):1348–1357, 2018.

[5] Qu ZhiPing, Zhang YuanQi, Sun Yi, and Lin XiangBo. A new generative adversarial network for texture preserving image denoising. In *2018 Eighth International Conference on Image Processing Theory, Tools and Applications (IPTA)*, pages 1–5. IEEE, 2018.

[6] Liqun Zhong, Guole Liu, and Ge Yang. Blind denoising of fluorescence microscopy images using gan-based global noise modeling. In *2021 IEEE 18th International Symposium on Biomedical Imaging (ISBI)*, pages 863–867. IEEE, 2021.