Deep Learning-based Approach to

2D Image Enhancement

Author	:	**CHEZIAN**, Nikhil
Matriculation Number	:	11027706
Course	:	M.Eng – Information Technology
Department	:	School of Engineering & Architecture
Address	:	Bonhoefferstraße 13, 69123, Heidelberg Germany
Email Address	:	chezian.nikhil@gmail.com
Supervisor	:	**Prof. Dr. Achim Gottscheber**
2nd Supervisor	:	**Prof. Dr. Milan Gnjatović**
Start Date	:	10.11.2023
Closing Date	:	08.05.2024

Abstract

This thesis presents a comprehensive deep learning-based approach to 2D image enhancement, employing a hybrid-adaptive optimization and feature fusion strategy with HASR-Net and the integration of the RealESRGAN model. HASR-Net utilizes a hybrid fusion of the Aquila optimizer and the Artificial Gorilla Troops Optimization method to adjust the learning rate, thereby improving convergence efficiency and addressing challenges such as vanishing or exploding gradients. The RealESRGAN model, a significant component of our framework, leverages a deep convolutional network architecture that includes multiple residual blocks and enhancements to improve the texture and detail of low-resolution images. Pre-processing includes bilateral filtering for noise reduction, normalization, and data augmentation techniques such as rotations, flips, and color jittering. Feature extraction is accomplished through learned features from Inception V3 and hand-crafted features via Local Binary Patterns (LBP), ensuring a comprehensive representation of image characteristics. The HASR-Net's dynamic optimization algorithm adapts the learning rate based on the model's performance, while RealESRGAN further refines image quality by enhancing the resolution and visual fidelity of the images. The integrated approach is evaluated on benchmark datasets, demonstrating superior execution in creating high-quality super-resolved images with reduced artifacts and enhanced visual fidelity.

Kurzfassung

In dieser Masterarbeit wird ein umfassender, auf Deep-Learning basierender Ansatz zur 2D-Bildverbesserung vorgestellt, der eine hybrid-adaptive Optimierungs- und Feature-Fusion-Strategie mit HASR-Net und die Integration des RealESRGAN-Modells verwendet. HASR-Net nutzt eine hybride Fusion des Aquila-Optimierers und der Artificial Gorilla Troops Optimization-Methode, um die Lernrate anzupassen und so die Konvergenzeffizienz zu verbessern und Herausforderungen wie verschwindende oder explodierende Gradienten zu bewältigen. Das RealESRGAN-Modell, eine wichtige Komponente unseres Frameworks, nutzt eine tiefe Faltungsnetzwerkarchitektur, die mehrere Restblöcke und Erweiterungen enthält, um die Textur und die Details von Bildern mit niedriger Auflösung zu verbessern. Die Vorverarbeitung umfasst bilaterale Filterung zur Rauschunterdrückung, Normalisierung und Datenerweiterungstechniken wie Drehungen, Spiegelungen und Farbverschiebungen. Die Merkmalsextraktion erfolgt durch gelernte Merkmale aus Inception V3 und handgefertigte Merkmale über Local Binary Patterns (LBP), wodurch eine umfassende Darstellung der Bildmerkmale gewährleistet wird. Der dynamische Optimierungsalgorithmus des HASR-Netzes passt die Lernrate auf der Grundlage der Leistung des Modells an, während RealESRGAN die Bildqualität weiter verbessert, indem es die Auflösung und visuelle Wiedergabetreue der Bilder erhöht. Der integrierte Ansatz wird anhand von Benchmark-Datensätzen evaluiert und zeigt eine überragende Ausführung bei der Erstellung hochwertiger, hochaufgelöster Bilder mit reduzierten Artefakten und verbesserter visueller Wiedergabetreue.

Declaration of Authorship

I hereby declare,

1. That my thesis is my own original work. I have written it independently without outside help and have not used any sources other than those indicated - in particular, no sources not named in the references.

2. That I have appropriately indicated any direct quotations or passages taken from literature, as well as the use of intellectual property from other authors, by providing the necessary citations within the work. This applies equally to the sources used for text generation by Artificial Intelligence (AI). In the case of utilizing AI-generated quotes, I have independently verified and ensured the source's reliability, which includes both the verbatim question/task and its corresponding answer (by means of screenshot).

3. That the thesis was not previously presented to another examination board.

I am aware that in case of suspicion, this thesis may be checked for plagiarism and for appropriate use and reference of others' work.

I am aware of legal consequences in cases of incorrect declaration.

Heidelberg, 08.05.2024 Nikhil Chezian

Place, Date Signature

Acknowledgement

I am thrilled to present my thesis titled "Deep Learning Approach to 2D Image Enhancement," which I pursued at SRH Hochschule Heidelberg from November 2023 to May 2024. This research journey has been profoundly shaped by the support and guidance of several key individuals.

Foremost, I express my heartfelt gratitude to Prof. Dr. Milan Gnjatović, whose expert guidance was indispensable throughout my research. His insightful feedback and enduring patience were vital in helping me overcome various challenges, enriching my learning experience. Prof. Dr. Gnjatović's depth of knowledge and meticulous approach have greatly influenced my professional growth, providing a foundation that was both challenging and rewarding.

I would also like to thank Prof. Dr. Achim Gottscheber for his invaluable support and understanding, which were crucial in my completion of this project.

My appreciation extends deeply to my parents as well. Their endless support and encouragement have been pivotal throughout my educational journey. Their unconditional love has fortified my determination and success.

Lastly, I acknowledge the contributions of all who have supported me, either directly or indirectly, in the execution of this thesis. Your involvement has been essential to my achievements.

Contents

List of Figures

List of tables

CHAPTER 1

INTRODUCTION

1.1 Background

Recently, significant advancements in image enhancement have been achieved due to the increasing demand for clearer images in various fields such as surveillance, digital imaging, medical diagnostics, and cartography. Enhancing 2D images has become crucial for extracting valuable information and improving overall interpretability. Conventional image improvement techniques have mostly addressed issues with color enhancement, noise reduction, and light control. Techniques such as exposure adjustment, contrast enhancement, and dynamic range compression have been employed to optimize visibility and detail by managing light. Additionally, techniques designed to minimize interference, like spatial and frequency domain filtering algorithms, have been developed to tackle noise problems stemming from image capture or transmission, thereby improving the overall image quality [1]. In addition, techniques for color enhancement have been employed to augment color data, ultimately elevating the visual attractiveness and interpretability of images. These advancements in image enhancement techniques have significantly contributed to enhancing the clarity and utility of 2D images across various applications. But as more complicated imaging circumstances arise and more advanced analysis is required, there's a greater need for creative solutions that go beyond the box. The accessibility of large datasets and the development of powerful computational abilities have made it possible to investigate new models and strategies for 2D picture augmentation. The varied range of images found in real-world applications and the high standards for visual quality provide problems that call for creative solutions [2].

1.2 Image Enhancement

Image enhancement is a method that improves an image's clarity and gives machine vision systems more precise information. It entails modifying a picture to look more precise, apparent, and aesthetically pleasing. For various CV tasks, techniques such as edge detection, noise reduction, and color stretching may optimize efficiency [3]. Images, satellite photos, and medical images are just a few of the images to which image augmentation can be performed. The image's characteristics and the intended result will determine the optimal technique for image enhancement. An image's appearance can be improved using various tools and strategies to make it more visually beautiful and better at communicating ideas [4].

1.3 Significance of Image Enhancement

Image enhancement plays an essential role across multiple sectors, such as healthcare, remote sensing, surveillance, and entertainment, by aiming to improve the clarity and readability of images. Techniques like contrast adjustment, histogram equalization, and sharpening are utilized in image enhancement to make details, patterns, and structures more visible, which might be obscured or subtle otherwise [5].In medical imaging, it aids in the accurate diagnosis of diseases by providing clearer representations of anatomical structures. In surveillance and remote sensing, image enhancement contributes to better object recognition and scene understanding. Moreover, in the realm of entertainment and media, enhancing images enhances the overall viewer experience by presenting visually appealing and detailed content. Overall, image enhancement significantly impacts diverse domains by maximizing the utility and information content of images, leading to better decision-making and understanding in various applications. There are several uses for image enhancement, including:

- **Photography:** To enhance a photograph's visual attractiveness, adjustments are made to its contrast, color balance, or clarity. To increase visual clarity and facilitate more precise health issue identification, image enhancement is an approach utilized in healthcare imaging. Additionally, it improves the contrast and quality of aerial images, making them simpler to look and comprehend. Methods for image processing can also be used to improve forensic pictures, such as fingerprint images or security camera footage.
- **Military and Defense:** The brightness and detail of images taken by military drones or satellites can be improved by enhancing them, which can be used to assist in identifying prospective dangers or gathering data.
- **Astronomy:** Astronomers can see and comprehend the finer details of astronomical objects more clearly by improving the brightness and clarity of telescope photos.
- **Industrial investigation:** To enhance the presence of data, image improvement can be performed on photos taken during industrial exams, such as assessing welds or searching for faults in objects [6] [7].

1.4 State of the Art technologies in Image Enhancement

Numerous image enhancement techniques are commonly applied to improve the visual quality of images for different uses. Here are five popular image enhancement methods:

1.4.1 Deep Learning-Based Methods

DL-based approaches, particularly CNNs, have proven successful in image enhancement tasks. CNNs can learn hierarchical representations of features, providing a robust framework for capturing intricate patterns in images. Generative Adversarial Networks (GANs) and U-Net architectures are two prominent paradigms in this domain. GANs introduce an adversarial training framework, where a generator network produces enhanced images and a discriminator network evaluates their realism. U-Net architectures, with a U-shaped structure, are effective in tasks like image segmentation and denoising by preserving fine details through skip connections. Both GANs and U-Net demonstrate deep neural network's ability to learn complex mappings, transforming input images into enhanced versions. This synergy has improved perceptual quality and opened up innovative applications in arenas like medical imaging, remote sensing, and digital media [8].

1.4.1.1 GANs

GANs, introduced by Ian Goodfellow in 2014, are a revolutionary approach in image enhancement and synthesis. They include a generator and a discriminator, which are trained using adversarial methods to generate lifelike images from random noise. GANs are adept at mastering intricate mappings from lower-quality to higher-quality images, yielding visually impressive outcomes. They are particularly effective for tasks like super-resolution, denoising, and style transfer. However, challenges like mode collapse and training instability persist, necessitating ongoing research to improve GANs' reliability. GANs also influence fields like art generation, data augmentation, and domain adaptation [9].

Generative Adversarial Networks (GANs) are a revolutionary approach to image enhancement, combining a generator and discriminator network for adversarial training. They produce high-quality, realistic images by transforming input images into enhanced versions and evaluating their authenticity. GANs adapt to diverse image characteristics, making them effective in tasks like super-resolution, denoising, and color correction. However, challenges like training stability and artifact generation remain, prompting ongoing research to further enhance GANs in medical imaging, digital art, and satellite imagery analysis [10].

1.4.1.2 SRGAN

Super-resolution is crucial in today's world, but traditional methods often struggle to achieve high-resolution images. Learning-based super-resolution approaches, such as Generative Adversarial Networks (GAN), have emerged to overcome these limitations. SRGAN creates convincing high-resolution photos from low-quality ones by combining GANs and deep convolutional neural networks. The discriminator uses adversarial training to discern between the generated and genuine images, while the generator produces high-resolution images that are identical to the actual reality. The discriminator minimizes losses by providing feedback and improving via backpropagation, while the generator attempts to fool it by collecting minute features and visual cues. [11].

- **Generator Architecture**

The procedure starts with sending a low-resolution image through a convolutional layer, followed by a parametric ReLU function. This input then passes through residual blocks, which are convolutional layers with 64 pixels. Each block undergoes the same sequence of batch normalization and convolutional layers, followed by batch normalization and an elementwise sum. Subsequently, the output goes through an elementwise sum, then upsampling and additional convolutional layers. The final step is a convolutional layer that produces a super-resolution image [12]. Figure 1.1 depicts the Generator's Architecture.

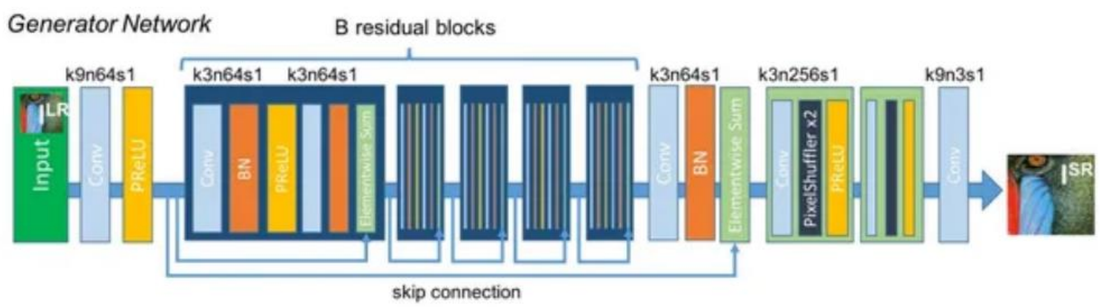


Figure 1.1: Generator Architecture (Source: Arxiv.org)

- **Discriminator Architecture**

A discriminator network is an image classification Convolution Neural Network (CNN) that differentiates between generated and real high-resolution images. It uses a convolutional layer to classify input images, extracting features through the Leaky ReLU function. The network proceeds through various discriminator blocks that include convolutional layers, Batch Normalization, and Leaky ReLU, before moving to a Dense layer, another Leaky ReLU, and a second Dense layer to generate an output. This classification method differentiates between authentic high-resolution images and those produced by the generator [13]. Figure 1.2 illustrates the Discriminator Architecture.

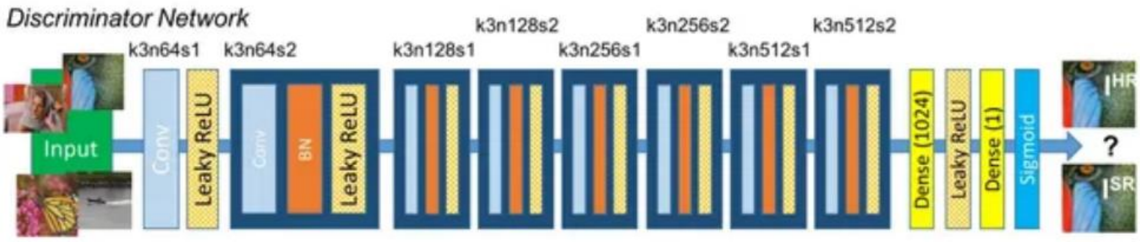


Figure 1.2: Discriminator Architecture(Source: [Arxiv.org]())

SRGANs are employed in a range of applications such as image and video enhancement, surveillance and security systems, medical imaging, and satellite imagery.They increase safety and security pictures, enhance low-resolution imagery, and increase healthcare diagnostic precision. They also aid in the analysis of satellite images, which often have lower resolution because of technical constraints, enabling improved comprehension of geographical features and weather patterns. Figure 1.3 shows the SRGAN.

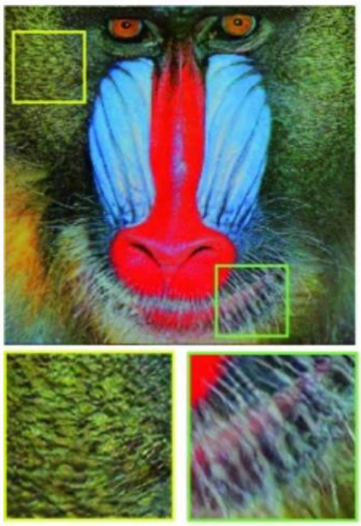


Figure 1.3: SRGAN (Taken from: [Arxiv.org]())

1.4.1.3 ESRGAN (Enhanced Super-Resolution GAN)

ESRGAN is a deep learning method specifically developed for image super-resolution, aimed at producing high-resolution images from their lower-resolution counterparts. It focuses on upscaling images with finer details, such as textures and edges, for improved visual quality. ESRGAN builds upon the traditional SRCNN architecture and introduces a GAN-based adversarial training mechanism. The generator network creates high-resolution images, and a discriminator network assesses their realism. ESRGAN employs perceptual loss to gauge the discrepancy between the generated images and the original ground truth images, enhancing both pixel-level fidelity and the overall perceptual quality. It has shown success in digital photography, video content, and medical imaging [14]. Figure 1.4 shows the super-resolution results of ESRGAN.

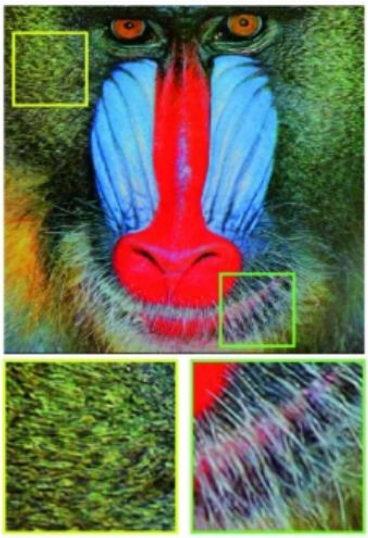


Figure 1.4: ESRGAN (Source: [Arxiv.org]())

1.4.2 Adaptive and Hybrid Approaches

Adaptive and hybrid approaches in image enhancement are a significant evolution in addressing the challenges of diverse image characteristics. These methods combine local techniques like contrast adjustments and histogram equalization with global optimization strategies to enhance specific regions and ensure a harmonious balance across the entire image [15]. This hybridization of methods leverages the strengths of both localized and global enhancements, fostering adaptability to various image features and complexities. By combining these techniques, they mitigate issues related to over-enhancement or under-enhancement in specific regions and offer a versatile solution for handling diverse image content effectively. Their popularity in image processing methodologies is a testament to their effectiveness in providing tailored enhancements [16].

- **Histogram Equalization (HE)**

HE is a18 method that extends the intensity variety of an image to improve distinction. By mapping one distribution to a broader, more uniform distribution, it seeks to guarantee an even distribution of intensity values throughout the whole range. For the remapping procedure, the Cumulative Distribution Function (or CDF) is employed. This technique is frequently applied to improve images by increasing contrast and bringing out smaller characteristics in areas that are lighter and deeper. It may not, nevertheless, consistently produce the desired effects, particularly when working with photos that have sharp peaks or troughs. To achieve the best outcomes, different strategies might be required [17].

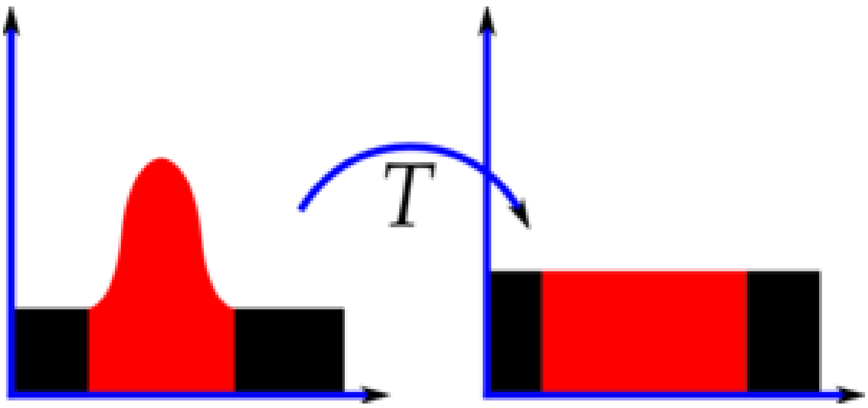


Figure 1.5: Histograms of an image before and after equalization

- **Contrast Stretching**

A method called contrast stretching sometimes referred to as histogram stretching or normalization, improves the visual appeal of an image by dispersing its intensity values over a larger range of values. By enhancing the contrasts between light and dark regions, this approach highlights small variances and raises the general sharpness of the image. The image must be normalized over a range of pixel values to execute contrast stretching. This range generally refers to the lowest and maximum values permitted for the image format, for example, 0 and 255 for 8-bit grayscale images [18].

- **Contrast-Limited Adaptive Histogram Equalization (CLAHE)**

An expansion of the predictable HE method is CLAHE. The image is split into smaller segments, and the histogram of each segment is equalized. CLAHE adds a contrast restriction technique to avoid both extreme brightness and noise amplification [19].

- **Retinex-Based Methods**

The goal of Retinex algorithms is to separate an image into elements related to light and reflection. The fundamental texture and complexity are represented by the reflectance component. Adaptive improvement techniques based on Retinax improve the reflectance element while maintaining lighting data [20].

- **Spatial Filtering**

An essential method for processing images is spatial filtering, which alters pixel values according to nearby neighborhoods. It reduces noise, recovers important characteristics, and improves the clarity of images. It is utilized for identifying tumors, identifying objects, and decreasing noise in domains like healthcare imaging, computer vision, and cinematography. Different types of filters, including median, Laplacian, Sobel, and Gaussian, have different uses in image processing. These filters successfully eliminate noise, smooth images, identify edges and features, draw attention to variations in intensity, and minimize noise. For creative thinking and the processing of visual data in domains such as CV and healthcare imaging, spatial filtering is essential [21].

- **Unsharp mask filtering**

One technique for sharpening images is unsharp masking (USM), which is often included in digital image editing applications. The process utilizes a blurry, or "unsharp," negative image to make a cover of the actual image, hence the term "unsharp". After that, the unsharped mask is joint with the positive (original) image to make a less hazy image than the distinctive one. Even though the second image is crisper, the subject matter may be depicted less accurately [22].

- **Adaptive Histogram Equalization (AHE)**

A method called "adaptive histogram equalization" makes use of several histograms to strengthen borders and increase local brightness in particular image areas. These histograms, each of which represents a distinct area of the picture, are computed using an adaptive technique. Nevertheless, it has the drawback of exaggerating contrast in areas close to contrast [23].

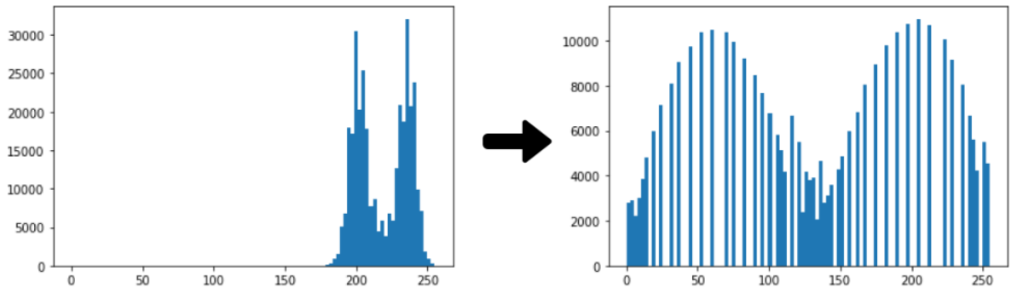


Figure 1.6: AHE

1.4.3 Image Fusion Techniques

Image fusion boosts object recognition systems by amalgamating data from spacecraft, aerial, and ground-based imaging systems, integrating details from several images into a single comprehensive image. It optimizes geometrical modifications, refines photos, highlights data not seen in either image, and highlights features not seen in either. To improve decision-making, image fusion also supplements and substitutes incomplete data sets. To accurately portray the environment and keep relevant data from the input photos, it merges key details from multiple original images into a single output image. The objective of multisensor image fusion is to conserve important information while presenting visual content from various images with distinct mathematical forms. Sharpening images, enhancing features, improving categorization, and producing stereo data sets are some benefits [24].

Pixel level, feature level, and symbol level/decision level are the three categories into which image fusion methods fall. Pixel-level fusion preserves the original image data while processing individual pixels. Data compression and transmission are facilitated by feature-level fusion, which processes the features of the source image. Decision-making level fusion minimizes uncertainty and complexity by using the collected information for optimal decision-making [25]. The following are the various image fusion methods that have been researched and created thus far.

- **Principal Component Analysis (PCA)**

The advantage of the PCA (Principal Component Analysis) method over the IHS (Intensity, Hue, Saturation) approach is that it allows for an unlimited number of groups to be utilized, regardless of their similarities. This is among the most widely used techniques for fusing images. The multispectral pics with low resolution are converted into independent principal components. The data shared by all bands utilized is contained in the first principal component (PC1). It provides extra data about the panchromatic image due to its great variability. Extended with the same dispersion as PC1, an excellent quality PAN component assumes the job of PC1. The excellent quality multispectral picture is subsequently obtained using an inverse PCA transform. [26].

- **Wavelet Transform**

Short-time Fourier transforms are less effective than wavelet transforms, which provide the required resolution in both the time and frequency domains. In contrast to the Fourier transform, which breaks down the signal into sine waves of numerous frequency ranges, it breaks down the data into scaled and shifted versions of the mother wavelet or function. In image fusion, DWT at a certain level is used to break down input images into rough and informative coefficients. These coefficients are then combined by employing a fusion algorithm to produce the inverse wavelet transform [27]**.**

- **High Pass Filtering (HPF)**

HPF produces multispectral high-resolution photos. The high-frequency information from the excellent quality panchromatic picture is blended with the low-resolution multispectral picture to produce the final image. The Excellent quality Panchromatic Image is alternatively subjected to an HPF or the LRPI is deducted off the original HRPI. This method maintains the spectrum information seen in the HRMI's frequencies low data. [28].

- **Discrete Cosine Transform (DCT)**

The DCT is essential for JVT and MPEG picture compression. The frequency domain image is created by converting the spatial domain image. Pictures are divided into low, medium, and high-frequency sections by Chu-Hui Lee and Zheng-Wei Zhou, which stand for high-frequency coefficients and average illumination, respectively. Greyscale pictures are created by dividing RGB images into 8x8 pixel blocks and grouping them using red, green, and blue grids [29].

1.5 Image Enhancement Challenges

- **Noise and Distortions:** In this case, noise represents visual flaws. These distortions may arise due to inadequate lighting, insufficient sensors, and data compression.

These factors contribute to the picture file's unclear appearance. These mistakes may affect how accurately photos are converted.

- **Variability in Image Quality:** This second primary worry has to do with the solutions for quality and resolution. Applying consistent programming throughout the database may be challenging due to fuzzy images and missing text. It is technologically capable of pausing execution for a bit.
- **Limited Spatial Resolution:** Low-resolution images may lack fine details, hindering the ability to discern important features and patterns.
- **Real-time Processing**: In certain applications, such as video processing or live streaming, the need for real-time image enhancement poses computational challenges that require efficient algorithms.
- **Contrast Variation:** Images often exhibit variations in contrast due to factors such as lighting conditions, which can lead to difficulties in visual interpretation [30].

1.6 Problem Statement

[31] discussed two approaches for enhancing the edge and texture of healthcare images, especially for liver lesions. The first involved determining the optimal kernel size for a range filter and comparing results with conventional edge detection procedures. The other technique discovered feature-based pixel-wise image fusion for abdominal images, improving contrast and sharpness for better visual interpretation. [32] introduced GSA-Net, a new image improvement procedure for low-light remote sensing images, which uses depth-wise separable convolution, global spatial attention, and selective kernel feature fusion. It outperforms other methods in image enhancement and object detection. [33] explored the use of deep learning in healthcare image processing, and highlighted its potential to revolutionize the healthcare sector. It discusses challenges in image interpretation and expert analysis, reviews advanced deep learning architectures, and addresses open research issues.

1.7 Objectives

The research aims to improve image enhancement by introducing a hybrid-adaptive optimization technique that combines local adaptive methods with global strategies. It also explores the effectiveness of feature fusion in image enhancement, using the HASR-Net's capabilities. The goal is to achieve superior image quality, overcome common enhancement challenges, and come up with the development of dynamic and versatile image enhancement practices. This approach has potential applications in medical imaging, remote sensing, and digital media.

1.8 Contribution of the study

The study is an innovative deep learning approach to 2D Image Enhancement, through Hybrid-Adaptive Optimization and Feature Fusion with HASR-Net introduces a novel methodology for image enhancement. It combines hybrid-adaptive optimization techniques with feature fusion using the HASR-Net architecture, refining the visual quality and interpretability of 2D images. The hybrid-adaptive optimization addresses challenges like contrast variation and limited spatial resolution. The HASR-Net's ability to learn complex mappings and preserve fine details enhances the effectiveness of the approach. This advancement in image enhancement techniques has potential submissions in healthcare imaging, investigation, and remote sensing, where precise visual information is crucial for accurate analysis and decision-making.

1.9 Thesis Organization

Chapter 1 provides an in-depth analysis of image enhancement, highlighting its importance and challenges. It discusses various techniques like Adaptive and Hybrid Approaches, DL-Based Approaches like CNN, and Image Fusion Techniques, laying the groundwork for further exploration.

Chapter 2 discusses the literature review of existing works and its advantages, and disadvantages.

Chapter 3, examines the proposed methodology

Chapter 4 discusses the results and discussion of the study

Finally, the thesis is concluded in chapter 5.

1.10 Summary

This Chapter presented a comprehensive indication of image enhancement, delving into its significance and the challenges faced in improving visual quality. The discussion underscored the diverse landscape of image enhancement techniques, including Adaptive and Hybrid Approaches, which blended local and global optimization strategies for enhanced adaptability. Additionally, the chapter explored cutting-edge DL-Based Methods, highlighting the success of CNN and GAN in transforming and refining images. Image Fusion Techniques were also introduced, showcasing their role in combining information from multiple sources for a more comprehensive and detailed enhancement process. This foundational chapter set the stage for a detailed exploration of these methodologies in subsequent sections.

CHAPTER 2

LITERATURE REVIEW

This chapter explores image enhancement techniques, focusing on both contemporary and historical approaches. It delves into the complexities of optimization and adaptive methods, examining their principles and mechanisms. The chapter also highlights feature fusion techniques, which combine different image features to improve image quality. The chapter provides a practical perspective on the real-world applications, challenges, and successes of each method, offering a nuanced understanding of their potential and limitations. By addressing both conventional and cutting-edge aspects of image enhancement, the chapter purposes to offer with a deeper appreciation for the diverse methodologies used in enhancing visual quality.

2.1 Overview of Image Enhancement

In 2021, Qi *et al.* [1] provided a thorough analysis of many picture-enhancing methods, covering both conventional and cutting-edge DL algorithms. It included an extensive variety of methodologies, including DL-based approaches, pulse-coupled neural networks, and retinex theory in addition to histogram equalization. It also gone over how to apply these strategies for improving low-light images and the difficulties in evaluating their quality. In addition, the paper provided an in-depth review of the subject by exploring the mathematical fundamentals and procedures underlying various picture-enhancing techniques.

In 2022, Al Sobbahi and Tekli [2] gave a summary of the improvement of digital low-light images (LLI), going over both conventional methods and the more recent application of DL algorithms. It described the difficulties in improving LLI, namely the dependence on artificial training data and the absence of standard benchmarks. The paper then provided empirical assessments of ten new DL-based LLI improvement algorithms that addressed feature analysis, visual effectiveness, and object recognition and classification performance. The researchers stress the requirement for more study in this field before classifying DL-based LLI improvement algorithms into five major groups.

In 2014, Chitradevi and Srimathi [3] provided a summary of the many approaches used in image processing, such as extraction of features, categorization, initial processing, and classifications. It covered the significance of image improvement and included descriptions of several methods, including noise filtering, contrast stretching, and histogram alteration. The Frost Filter approach, gradient-based adaptive median filter, wavelet coefficient shrinkage filter, and other techniques were among the techniques highlighted in the paper's analysis of contemporary computer vision studies.

In 2018, Arif *et al.* [4] discussed utilizing MATLAB for performing HE to improve the image. The application of histogram equalization to each RGB channel and a grayscale image of a rat was covered in the first section. In the second section, which concentrated on histogram descriptions, the target histogram for a colored image featuring a rat was introduced and applied to the initial color image. The paper stressed the value of image-enhancing methods across fields, including the detection of patterns, algorithmic photography, forensics, and health imaging. It emphasized how important histogram-based methods are for picture improvement because of their basic nature and effectiveness, particularly histogram equalization and histogram definition.

In 2011, Kumar and Chandrakar [5] provided a summary of image augmentation approaches, emphasizing the effects of illumination and reflection on the visual system. It covered techniques for enhancing the quality of images, including SSR, MSR, MSR-CR. The significance of attaining color consistency was underscored by the writers, who also underlined the applicability of these methods in processing pictures. Table 2.1 shows the review of various authors about the overview of image enhancement techniques.

Table 2.1: Review of various authors about the Overview of image enhancement techniques.

Authors name	**Methods used**	**Advantages**	**Disadvantages**
Qi et al. [1]	• Conventional and cutting-edge DL algorithms,	• Covered a wide variety of methodologies. • In-depth exploration of	• Bias and Inconsistencies • Limited Focus on Applications

	• Pulse-coupled neural networks, • Retinex theory, • Histogram equalization	mathematical fundamentals. • Application on low-light images discussed and Challenges in evaluating image quality addressed.	
Al Sobbahi and Tekli [2]	• Conventional methods	• Empirical assessments of ten DL-based LLI improvement algorithms • Identification of challenges in improving LLI. • Stress the need for more research in the field.	• Dependence on artificial training data • Absence of standard benchmarks
Chitradevi and Srimathi [3]	• Noise filtering • Contrast stretching • Histogram alteration.	• Comprehensive summary of image processing approaches. • The significance of image improvement is highlighted.	• Limited Practical Application Insights • Biased Perspective

Arif et al. [4]	• Histogram equalization.	• Application of histogram equalization on both RGB and grayscale images. • Stress on the value of image-enhancing methods across different domains.	• Absence of Comparative Analysis • Limited Scope
Kumar and Chandrakar [5]	• SSR, • MSR, • MSR-CR	• Summary of image enhancement methods emphasizing illumination and reflection effects. • The importance of colour consistency highlighted. • Applicability of methods in image processing discussed.	• Computational Complexity • Parameter Sensitivity

2.2 Traditional Image Enhancement Techniques

In 2014, Singh and Mittal [6] presented a rigorous analysis of several picture-enhancing strategies, dividing them into approaches based on frequencies and geographical domains. It goes over the idea of improving images, the many methods employed, and how they are used. While frequency domain methods change the image's orthogonal transform, spatial domain strategies directly influence pixel values. Uses of enhancing pictures in forensics, medical imaging, and aerial imaging were also explored in the paper.

In 2013**,** Rajput and Suralkar [7] examined many methods for improving photographs, with a particular emphasis on fingerprint photos. It was noted that the efficacy of machine learning algorithms is impacted by the quality of fingerprint images, which is why enhancing approaches are required. The study assessed several techniques using performance criteria such as PSNR, Normalized Error, Correlation Coefficient, and Structural Resemblance. These techniques included histogram equalization, negative picture improvement, contrast stretching, and several filters. According to the research's findings, the contrast image enhancement technique produces the best outcomes. The significance of using picture-improving methods before extracting minutiae for the detection of fingerprints was also emphasized in the paper.

In 2014, Wu *et al.* [8] discussed how important picture improvement techniques is to raise the caliber of grayscale photos. With an emphasis on wavelet-based procedures, it investigated different picture-enhancing algorithms based on the space and frequency domains. Mean, SD, MSE, PSNR, and other numbers are used to evaluate and analyze the benefits and drawbacks of various methods. The article emphasizes the usefulness of wavelet transforms for picture improvement and denoising, highlighting the efficiency of the wavelet transform modulus maxima approach specifically.

In 2015, Puniani and Arora [9] The technique of improving a picture's perception data by extracting additional details and information from it is known as image color improvement. By using the initial data, it enhances an image's aesthetic grade. The efficacy evaluation of several picture-enhancing approaches is the primary objective in this article. These methods are taken from fuzzy logic or modifications to histograms. Two quantitative metrics, the Tenengrad measure and the Contrast Improvement index (CII), are used to evaluate the methods. The outcomes demonstrate that the greatest outcomes are obtained with Lab and edge preservation-based fuzzy image enhancement (LEFM).

In 2011, Dhariwal [10] examined several enhancement methods, with a particular emphasis on brightness preservation. It brought to light the shortcomings of conventional techniques like HE in consumer electronics as well as the significance of contrast augmentation. To overcome these constraints, the article presents sophisticated methods as DHE, DSIHE, and CLAHE. Additionally, the writer provided an analysis of various methods utilizing metrics such as visual quality, CPSNR, and EC. The summary highlights the unique features of each method and makes recommendations for the next study avenues. Table 2.2 shows the Review of various authors about the Old-style Methods of Image Enhancement.

Table 2.2: Review of various authors about the Traditional Techniques of Image Enhancement.

Author name	Methods used	Advantages	Disadvantages
Singh and Mittal [6]	• Frequency and spatial domain-based approaches	• Rigorous analysis of picture-enhancing strategies. Classification into frequency and spatial domain methods. • Exploration of applications in forensics, medical imaging, and aerial imaging.	• Incomplete Coverage • Insufficient Comparative Analysis
Rajput and Suralkar [7]	•	• Assessment using performance criteria like PSNR, Normalized Error,	• Efficacy of machine learning algorithms

		Correlation Coefficient, and Structural Similarity. • Highlighting the effectiveness of the contrast image enhancement technique.	impacted by image quality.
Wu et al. [8]	• Wavelet-based procedures	• Investigation of picture-enhancing algorithms based on space and frequency domains, with emphasis on wavelet transforms. • Evaluation using metrics such as mean, standard deviation, mean square error, PSNR, etc.	• Lack of Real-World Application Scenarios • Failure to Address Implementation Challenges
Puniani and Arora [9]	• Fuzzy logic and histogram modifications	• Focus on image color improvement using fuzzy logic or histogram modifications. • Evaluation using quantitative	• Subjectivity in Quality Assessment • Limited Benchmarking

		metrics (Tenengrad measure and Contrast Improvement index). • Highlighting Lab and edge preservation-based fuzzy image enhancement (LEFM) as yielding the best outcomes.	
Dhariwal [10]	• Various methods, with emphasis on brightness preservation	• Presentation of sophisticated methods like DHE, DSIHE, and CLAHE. • Analysis using metrics such as visual quality, CPSNR, and EC.	• Shortcomings of conventional techniques like HE highlighted. • Constraints of HE in consumer electronics discussed.

2.3 Adaptive Image Enhancement Techniques

In 2004, Reza [11] aimed to build a system's execution of CLAHE to enhance the visual appearance of captured actual time image patterns, such as those utilized in X-ray imaging. This method, which is commonly employed for offline picture augmentation, requires a large amount of computing. Reducing latency without compromising accuracy is the aim. The study suggested a system-wide implementation of CLAHE to improve actual time image patterns, appropriate for VLSI or FPGA implementation. This method, which requires a lot of computing power, is frequently applied to offline image improvement. The objective is to reduce latency and improve the visual quality.

In 2016, Guo *et al*. [12] discussed about the difficulty of taking low-visibility photos in low light and how it affects media and artificial intelligence techniques. To solve this problem, it presented a suggested technique dubbed LIME. To improve the image, the technique involves calculating each pixel's lighting and fine-tuning it using architectural assumptions. Trials are conducted to validate the effectiveness of LIME and illustrate its advantage over present approaches when it comes to effectiveness and level of improvement.

In 2014, Jintasuttisak and Intajag [13] provided a technique for preserving color information in retinal fundus images, which are essential for ophthalmologists to detect vision-related conditions including diabetes and hypertension. The technique makes use of the iNHSI. The Rayleigh CLAHE technique uses the Rayleigh transformation to improve the brightness element, which improves the entire look and brightness. To make sure the original color element stays the same, the system was evaluated utilizing a common public database for diabetic retinopathy identification from digital photos.

In 2015, Fu *et al.* [14] outlined an algorithm for enhancing images that concurrently calculates reflectivity and light in the linear domain. It is discovered that the exponential realm framework is less accurate in representing historical data for predicting light and reflectivity than the linear domain approach. The method effectively estimated light and reflection using an alternating direction multiplier technique combined with a maximum a posteriori formulation. The laboratory findings demonstrate excellent converging stages, aesthetically pleasant modifications, and adequate efficiency. two types of evaluations produced outcomes for the approach that were on par with or superior.

In 2018, Cai *et al*. [15] highlighted how the goal of digital imaging is to capture genuine circumstances with full details and vivid colors, but the poor distinction and poor clarity are frequently caused by inadequate illumination and a narrow dynamic range in the photos. This may hurt CV algorithm performance as well as the aesthetic value of images. As a result, increasing the contrast is essential to raising the standard of the image and making features more visible. Table 2.3 shows the Review by various authors about the Adaptive Techniques of Image Enhancement.

Table 2.3: Review by various authors about the Adaptive Techniques of Image Enhancement.

Author's names	**Method used**	**Advantages**	**Disadvantages**
Reza [11]	• CLAHE • VLSI or FPGA.	• Latency reduction in real-time image patterns. • Improved visual quality. • System-wide implementation for VLSI or FPGA proposed.	• High computing power required, particularly for offline image improvement.
Guo et al. [12]	• LIME	• Effective enhancement of low-light images. • Validation through trials. • Improved effectiveness and level of improvement compared to existing approaches.	• Sensitivity to Input Conditions • Parameter Sensitivity

Jintasuttisak and Intajag [13]	• iNHSI • Rayleigh CLAHE	• Preservation of color information in retinal fundus images. • Brightness improvement using Rayleigh CLAHE. • Evaluation on a common public database.	• Computational Complexity • Lack of Comparative Analysis
Fu et al. [14]	• Probabilistic method • Posteriori formulation and alternating direction method of multipliers.	• Simultaneous estimation of illumination and reflectance in the linear domain. • Satisfactory performance, visually pleasing enhancements. • Promising convergence rates. Comparable or better results in assessments.	• Data Dependency • Model Assumptions

Cai et al. [15]	• Digital imaging	• Improved contrast for better visibility of features. • Addressing issues of poor distinction and clarity caused by inadequate illumination and narrow dynamic range.	• Overfitting Concerns • Computational Complexity

2.4 Feature Fusion in Image Enhancement

In 2018, Zhang et al. [16] indicated that the presence of syntactic discontinuities and spatial resolution may make a straightforward merger of minimal and higher-level characteristics ineffective. It is more efficient to incorporate high-resolution data into high-level features and semantic information into low-level characteristics. To close this gap, the researchers suggested ExFuse, a new structure that increases segmentation quality by 4.0%. The method beats earlier contemporary outcomes using the PASCAL VOC 2012 benchmark.

In 2022, Chen *et al*. [17] suggested a technique that makes use of a feature fusion framework to enhance the classification of semantic images. This process photos of varying decisions, placing them into a network structure, and using fully convolutional networks to extract initial features with varying granularities were the steps in the procedure. After an ascending feature fusion initialization, the score map is computed and transmitted to an optimization-dependent random domain. Using the PASCAL VOC 2012 and PASCAL Context datasets, the suggested approach produced more favorable outcomes than earlier research.

In 2017, Ancuti *et al*. [18] offered a technique for enlightening underwater photos that have been deteriorated by absorption and dispersion. Two images from a color-compensated and white-balanced rendition of the initial deteriorated image were blended in this approach. To encourage the transfer of edges and color contrast, the two pictures are fused according to the weight maps that correspond with them. It was demonstrated that the technique worked well for improving photos taken using various cameras and under diverse underwater settings. Furthermore, it is shown that the improved images can be effectively used in CV submissions for key point identification and segmentation of images.

In 2020, Zamir *et al.* [19] contained citations to a range of industry methods and methods, including non-local image processing, sparse representation, and DL methods. The articles addressed an extensive variety of subjects, such as variational methods, DCN, and sparse regression, for image augmentation, denoising, and super-resolution. References to certain computing and computer vision works, writers, and events also appeared in the compilation.

In 2015, Liu *et al.* [20] highlighted how difficult it is to get every object in focus in a single shot since optical lenses have a restricted range of frames. It investigated the multi-focus image fusion approach as a remedy, grouping the other fusion techniques into four groups: spatial domain, feature space transform, multi-scale transform, and PCNN. The study provided a thorough fusion strategy along with an innovative fusion technique utilizing DSIFT. Comparing the suggested approach to other fusion strategies, studies show that it is better. The conclusion of the paper thanked several people and organizations for their contributions and encouragement. Table 2.4 shows the Review by various authors about the Feature Fusion Techniques in Image Enhancement

Table 2.4: Review by various authors about the Feature Fusion Techniques in Image Enhancement

Authors names	**Methods used**	**Advantages**	**Disadvantages**
Zhang *et al*. [16]	ExFuse	• Achieved a 4.0% increase in segmentation quality. • Outperformed previous outcomes on the PASCAL VOC 2012 benchmark.	• Dependency on Training Data • Limited Robustness to Varied Conditions
Chen *et al*. [17]	Feature Fusion	• Produced more favourable outcomes	• Computational Complexity

		than earlier research on the PASCAL VOC 2012 and PASCAL Context datasets.	• Model Interpretability
Ancuti et al. [18]	Blending two images from a color-compensated and white-balanced rendition of the initial deteriorated image.	• Demonstrated effectiveness for improving underwater photos. • Improved images can be effectively used in computer vision applications.	• Dependence on Environmental Conditions • Complexity and Computational Load
Zamir et al. [19]	Compilation of Industry Methods	• Comprehensive compilation of methods and techniques from various domains in image processing and computer vision.	• Data Dependence • Limited Robustness
Liu et al. [20]	Multi-focus Image Fusion	• Demonstrated superiority to other fusion strategies.	• Interpretability • Sensitivity to Hyperparameters

2.5 Optimization methods in Image Enhancement

In 2019, Nadipally [21] presented a novel texture segmentation approach built on the metaheuristic method known as ant colony optimization (ACO), which has shown promise in resolving optimization issues. The ACO-based system defines the path probabilities and motion complexity for artificial ants, allowing it to process MRI and iris data efficiently. Compared to more conventional metaheuristic categorization techniques, this novel approach had benefits including durability, favorable input, parallelism, and excellent selectivity. The outcomes of the experiments demonstrated the competitiveness and promise of the ACO-based image-processing techniques, particularly when it comes to photos with intricate local texture problems. This method shows promise for image analysis as it was especially successful in image-pattern recognition and MRI analysis.

In 2017, Lore *et al*. [22] acknowledged financial limitations while discussing the value of excellent images and videos for decision-making in a variety of professions. It contrasts it with other approaches and presents the Low-light Net (LLNet) structure for improving low-light photos. The usefulness of the suggested structure was demonstrated by the test outcomes and evaluation metrics that were reported in the study. In closing, it emphasized the promise of machine learning methods for improving images and expressed gratitude for the funding the project was granted.

In 2016, Li *et al*. [23] discussed about how the effects of absorption and dispersion, which lower the standard of the acquired photographs, make editing underwater photos difficult. It presented a methodical technique that included a contrast improvement method and a dehazing technique for improving underwater photographs. The contrast enhancement algorithm boosts the contrast as well as the brightness of the photos, while the dehazing program attempts to restore visibility, color, and a natural appearance. The suggested approach was created to yield two higher-quality versions: one for exhibition with an accurate color and organic look, and another for collecting additional helpful data and revealing more information.

In 2018, Li *et al.* [24] outlined the difficulties in enhancing images in low light and suggested a reliable Retinex framework to deal with these difficulties. It introduced an optimization function with unique normalization factors for lighting and absorption and emphasized the shortcomings of current techniques for controlling noise in low-light photos. The purpose of the suggested approach was to enhance low-light photographs with high noise levels. The significance of software-level methods for improving low-light images was highlighted in the article, along with their possible advantages for commercial shooting and deep-learning approaches.

a). In 2018, Chen *et al.* [25] described a technique for enhancing images with Generative Adversarial Networks (GANs) through unpaired learning. The suggested approach made use of a two-way GAN framework enhanced with features including separate sample normalizing layers for generators, an adaptive weighting strategy for Wasserstein GAN (WGAN), and a global U-Net generator. The stability of GAN training for picture improvement was enhanced by these improvements. The outcomes showed how well the suggested technique improved photos in both a numerical and aesthetic sense. Table 2.5 shows the Reviews by various authors about the Optimization techniques in Image Enhancement

Table 2.5: Reviews by various authors about the Optimization techniques in Image Enhancement

Authors name	**Method used**	**Advantages**	**Disadvantages**
Nadipally [21]	• ACO-based Texture Segmentation Approach	• Durability, favourable input, parallelism, and excellent selectivity. • Competitive and promising results, especially for complex local texture problems. • Successful in image-pattern recognition and MRI analysis.	• Difficulty in Handling Noise and Variability • Risk of Premature Convergence
Lore et al. [22]	• LLNet Structure	• Demonstrated usefulness through test outcomes and evaluation metrics. • Emphasized the promise of machine	• Acknowledged financial limitations. • Sensitivity to Parameters

		learning methods for image improvement.	
Li et al. [23]	• Systematic Technique	• Contrast enhancement and dehazing techniques for improving underwater photographs. • Creation of two higher-quality versions: one for exhibition and another for collecting additional data.	• Limited Applicability • Loss of Fine Details
Li et al. [24]	• Retinex Framework	• Optimization function with unique normalization factors for lighting and absorption. • Highlighted the significance of software-level methods for improving low-light images.	• Emphasized shortcomings of current techniques for controlling noise in low-light photos.
Chen et al. [25]	• GANs through Unpaired Learning	• Two-way GAN framework with features like separate sample normalizing layers and an adaptive weighting strategy for	• Quality Control • Dependency on Training Data

		Wasserstein GAN (WGAN). • Enhanced stability of GAN training for image improvement.	

CHAPTER 3

PROPOSED MODEL

3.1 Introduction

Image enhancement refers to the systematic process of refining digital images to enhance their quality for display or further image analysis purposes. It reduces noise amplification, achieves good real-time performance, and enhances the image's overall and local contrast as well as its visual impact and suitability for computer processing or human observation. A new Hybrid-Adaptive Super-Resolution Model (HASR-Net) model is presented in order to address the issues with Image Enhancement techniques. The stages that make up the suggested model are listed below, and Fig. 3.1 shows the architecture of the proposed approach.

1. Input Layer

In input layer the raw data is provided for processing. Moreover, it accepts low-resolution data as an input image.

2. Pre-processing Pipeline

The following steps are carried out by pre-processing phase

- **Data Augmentation Techniques**

Here the images are rotated as different angles for attaining the desired outcomes. Moreover, the images are rotated as 35- degree, 45- degree and 60-degree respectively. After image rotation, then the images are flipped horizontally and vertically. Furthermore, the colours are added to the images and it is defined as colour jitter.

- **Bilateral Filtering**

In this research, the Bilateral filters are used for noise reduction and clarity improvement.

- **Normalization**

It provides uniform data ranges for all images.

3. Feature Extraction

- **Local Binary Patterns (LBP)**

Extracts hand-crafted features to complement learned features.

4. High-Degree Modelling (HDM):

- Uses the basic concepts of Real-ESRGAN to produce excellent high-definition results.

5. Improved Resolution Engine (IRE):

- **Hybrid Optimization Algorithm**:
- Integrates the effectiveness of Artificial Gorilla Troops Optimization with the flexibility of the Aquila optimizer.
- Adapts the learning rate dynamically to improve convergence efficiency and handle gradient-related issues.

6. Output Layer:

- Produces high-resolution super-resolved images.

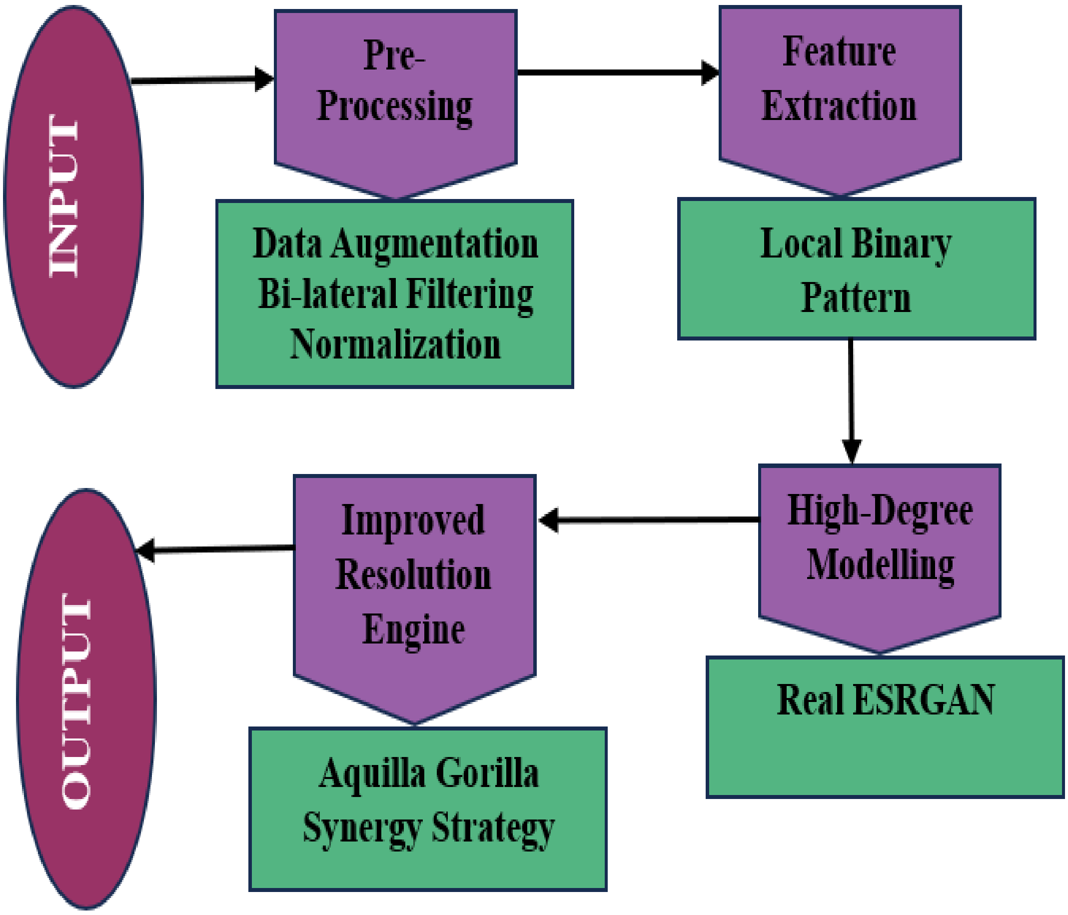


Fig 3.1: Proposed model

3. 2. Input Layer

This is the model's initial acceptance phase for low-resolution images. The input images were taken from a dataset which will undergo numerous types of processes to enhance image quality and produce higher-resolution output that performs better.

3. 3. Pre-processing

This step involves changing the input data in order to make it prepared for analysis. Data preparation is essential for rendering the data reliable, consistent, and suitable for analysis. It enhances the quality and effectiveness of data extraction procedures. To transform raw data into the most useful format, pre-processing is an essential and initial step. The input images are pre-processed using three techniques namely data augmentation, bilateral filtering and normalization. This section contains a thorough explanation of each technique. Together, these pre-processing stages get the input images ready for the HASR-Net architecture's super-resolution processing. By reducing noise, normalizing input ranges, and enriching the dataset with a range of samples, the pre-processing pipeline builds a solid foundation for the latter steps of the super-resolution process. In final, this improves the model's ability to generate pictures with higher quality and resolution.

(1) Data augmentation

Even with the availability of data, it can be difficult to retrieve the accurate information that most closely aligns with our study and experiment. Furthermore, the data must be sufficiently varied to allow for the presentation of different sizes, positions, colors, and lighting conditions to improve the model's performance when working with the data. To tackle the problem of limited data, data augmentation techniques are employed

- **Rotation**

The image can be oriented at minute angles or rotated at 90-degree angles, depending on the situation. After orientation, no background noise is introduced to the image when it is rotated at a 90-degree angle. However, this is untrue when rotation is carried out at small angles. Furthermore, any added color noise will integrate seamlessly into the image if the background is either black or white. However, if the image's background contains distinct colors, it will not combine and the network will recognize it as an image feature. The safety of the augmentations will be determined by the rotation degree parameter. For small rotations in digit identification, it is helpful. The label of the data vanishes during transformation when the rotational degree is increased. Thus, it must be handled carefully.

- **Flipping**

Similar to the rotation, it is possible to flip the image vertically or horizontally. It creates visuals by multiplying the image by ninety degrees. Nevertheless, some frameworks prevent vertical flipping. To flip an image vertically, rotate it 180 degrees, and then flip it horizontally.

- **Color jittering**

Using b and c, which lie between [−20, +20] and [0.8, 1.2], user can adjust the RGB image's brightness and contrast. To adjust for color jitter, a parameter p, where $p \in [-20, +20]$, is applied to each RGB channel's intensity. Where $q \in [-20, +20]$, the luminance intensity is adjusted in luminance jitter augmentation. Note that b, p and q are integer values. We only apply one of colour jitter or luminance jitter. The relighting model's ability to generalize to radically disparate data distributions is enhanced by these augmentations, which yield a training dataset that records several permutations of the input images.

(2) Bi-lateral Filtering

Image filtering represents one of the most compelling applications of image processing. Through image filtering, one can modify various attributes such as size, shape, color, depth, and smoothness. This technique plays a critical role in addressing many scientific and engineering challenges across fields such as signal processing, electronics, physics, and chemistry. In image processing and analysis, filters are essential for tasks like smoothing and de-blurring. The selection of filters, whether linear or non-linear, depends on the specific requirements of the application. Non-linear filters, such as bilateral filtering, are particularly effective when the input has sparse but intense noise, as they reduce noise while preserving sharp edges. This capability contrasts with linear low-pass filters, which are better suited for inputs with pervasive but less intense noise. Despite their simplicity and speed, linear filters often blur edges, which is why non-linear filters like the bilateral filter are preferred for maintaining edge integrity in digital photography and medical imaging. The bilateral filter works by adjusting the intensity of each pixel based on a weighted average of neighboring pixels, considering both spatial proximity and intensity similarity, thereby ensuring enhanced preservation of important image details.

This work utilizes bilateral filtering, a non-linear filter that effectively reduces noise while preserving edge information. Unlike traditional filters that simply average pixel values within a region, bilateral filtering considers both the geographic proximity and intensity similarity between pixels. This dual approach not only diminishes noise but also ensures sharp edges are maintained. Bilateral filtering calculates a weighted average of neighboring pixels based on their spatial and intensity closeness, thus preserving essential details while significantly reducing noise. Its adaptability makes it especially valuable in fields like digital photography and medical imaging, where enhancing image quality through noise reduction is crucial. Bilateral filtering is a type of non-linear image smoothing technique that excels in balancing noise reduction with edge retention. It typically employs a Gaussian distribution to determine the weights for averaging the intensity values of neighboring pixels.Crucially, in addition to the pixels' Euclidean distance, the weights also depend on radiometric differences (i.e., range disparities, color intensity, depth distance, etc.). In this way, sharp edges are maintained. The bilateral filter is mathematically given as,

$$I^{filtered}(x) = \frac{1}{X_p}\sum_{x_i \in \Omega} I(x_i)k_r(\|I(x_i) - I(x)\|)g_s(\|x_i - x\|) \tag{1}$$

And normalization term, X_p is defined as

$$X_p = \sum_{x_i \in \Omega} k_r(\|I(x_i) - I(x)\|)g_s(\|x_i - x\|) \tag{2}$$

x are the coordinates of the current pixel to be filtered; Ω is the window centered on x, thus $x_i \in \Omega$ is another pixel; $I^{filtered}$ is the filtered picture; I is the original input image to be filtered; The spatial (or domain) kernel, g_s is used to smooth variations in coordinates. The range kernel, k_r is used to smooth changes in intensities.

$$w(i,j,k,l) = exp\left(\frac{(i-k)^2+(j-1)^2}{2\sigma_d^2} - \frac{\|I(i,j)-I(k,l)\|^2}{2\sigma_r^2}\right) \tag{3}$$

where $I(i,j)$ and $I(k,l)$ are the intensities of poles (i,j) and (k,l) respectively, and σ_d and σ_r are the smoothing parameters. The weights should be normalized after calculation which is given as,

$$I_D(i,j) = \frac{\sum_{k,l} I(k,l)w(i,j,k,l)}{\sum_{k,l} w(i,j,k,l)} \tag{4}$$

where $I_D$is the pixel (i,j) denoised intensity.

(3) Normalization

It is possible to minimize redundant data, remove information that is duplicated, enhance data analysis, and fortify data security. Errors, inconsistencies, repetitions, and missing data can be eliminated, which can correct your data and analysis. There are various forms of normalization. Z-score normalization, with a range of 0 at mean and 1 at standard deviation, has been used.

$$V_{norm}(DS) = \frac{V-\mu}{\sigma} \tag{5}$$

Where V_{norm}→ Normalized data, V→ Input data σ →Mean, μ→ Standard deviation

$$\text{Mean, } \mu = \frac{1}{n}\sum_{i=1}^{n} V_i \tag{6}$$

$$\text{Standard deviation, } \sigma = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(V-\mu)^2} \tag{7}$$

3.4. Feature Extraction

Approaches for feature extraction have drawn a lot of interest in various research domains. It is critical, and the use of the Local Binary Patterns (LBP) technique highlights its significance. LBP offers a dependable method for obtaining texture information from raw data, particularly for applications in computer vision and image processing. By dividing texture elements into surrounding areas and encoding binary patterns based on pixel intensity comparisons, LBP efficiently recovers texture elements from photos. Histograms, which quantify these qualities, are then used to concisely yet informatively portray the underlying textures in the data. This technique not only improves computation efficiency but also encourages robustness against noise and light variations. The model may effectively decrease redundancy and improve classification accuracy by prioritizing the extraction of essential and discriminative features. This has the potential to enhance categorization results in numerous research areas.

3.4.1 LBP

Color features only use individual pixels, whereas texture features use groups of pixels. In the feature maps, a Local Binary Pattern (LBP) is computed for every pixel. After comparing the data, the results are binary encoded. A collection of binary characteristics is produced that capture certain local texture patterns. It derives texture information from the surface features, patterns, edges, or edges of the segmented objects. A helpful probabilistic operator for defining localized image attributes is the LBP. A central pixel's (x_c, y_c)grey value is compared to the pixels of its eight neighbours to produce an ordered binary set that is defined as LBP. As a result, the LBP code is represented as an octet value in decimal form as,

$$y = LBP(p_d, q_d) = M(\sum_{n=0}^{7} S(i_n - i_d)2^n) \quad (8)$$

Where i_d is the grey value of the centre pixel (x_d, y_d), i_n is the grey value of the pixels of its eight neighbours. It has been demonstrated that LBP code is invariant to all grey level transformations, and that the transformed local neighborhood binary code is unaltered.

$$S(i_n - i_d) = \begin{cases} 1 & ; (i_n - i_d) \geq 0 \\ 0 & ; (i_n - i_d) < 0 \end{cases} \quad (9)$$

3.5. High-Degree Modelling (HDM)

Real-ESRGAN is used in this work to achieve high-quality super-resolution of images. Real-ESRGAN's reconstructions of the images lacked image texture details and were overly smooth. This is primarily due to two factors. Using HDM, first of all, causes excessive image deterioration, which has the opposite effect on photos that do not suffer from as severe deterioration as they would in real life. Second, training the model becomes significantly more challenging when dealing with a complex and high-intensity image degradation technique because of the unique core mechanism of GAN. This inherently lowers the quality of the rebuilt image effect. As neural network-based generative models, GANs have the ability to produce original, meaningful literary and visual art. The generator and discriminator are two essential subsystems that make up the GAN framework. A family of generative models called Super-Resolution Generative Adversarial Networks (SRGANs) was created to tackle the problem of improving image resolution. ESRGAN is an improved version of super-resolution GANs (SRGANs) that emphasizes training efficiency optimization and complexity reduction while preserving the fundamentals of adversarial learning-based picture super resolution.

Real-ESRGAN, an improved version of ESRGAN, successfully addresses problems including the elimination of annoying compression, offering a more workable method for real-world image restoration. Compared to ESRGAN, Real-ESRGAN addresses a significantly larger degradation space, hence the discriminator architecture used in ESRGAN is inappropriate. The discriminator in Real-ESRGAN requires stronger discriminative capacity for complex training outputs. In addition to differentiating between global styles, it must guarantee a precise gradient response for local textures.

With skip connections, the VGG-style discriminator in ESRGAN was improved to a U-Net design. In order to give the generator precise replies for every pixel, the U-Net creates realness values for every pixel. In the meantime, complex degradations and the U-Net design exacerbate training instability. Stabilizing training dynamics can be facilitated by regularizing spectral normalization. Moreover, spectral normalization helps lessen the irritation and over sharpness that come with GAN training. With these modifications, Real-ESRGAN training may easily achieve a better balance between artifact suppression and local detail improvement.

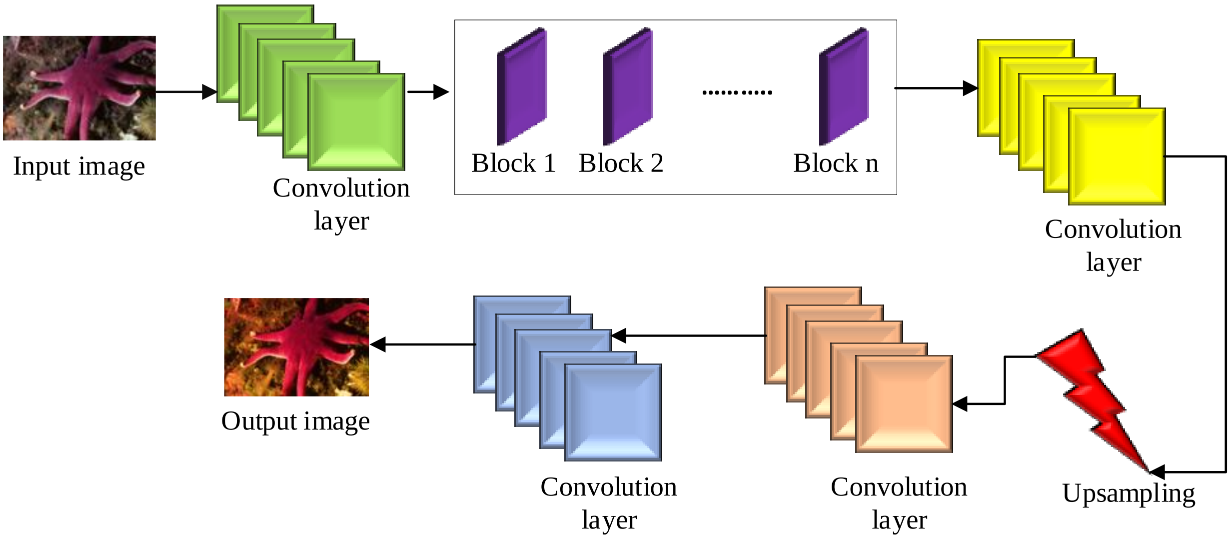


Figure 3.2: Architecture of ESRGAN

In this research, the function of ESRGAN is to enhance the image colour. Moreover, here the input image colour can be improved by ESRGAN mechanism. The figure 3.2 shows the architecture of ESRGAN.

Convolution layer in ESRGAN: The main objective of the convolution layers a crucial component of a CNN, is to perform feature extraction. The image that was input is convoluted using convolutional managers, as well as the output is saved to various channels within the layer for convolution. Every pixel in a convolution's responsive area is combined to one result. For example, applying a convolution algorithm on an image will result in a reduction in the size of the image and the consolidation of all the area's data into just one pixel. The vector format is the convolutional layer's final result. The maps of features may be made smaller by using pooling layers.

Upsampling in ESRGAN: Up sampling is the process of improving the spatial accuracy without sacrificing an image's 2D interpretation. It is usually used to remove the pixels appearance that appears whenever an image with a low resolution is presented on a reasonably big frame, as well as to zoom down on a specific area of an image. Uses for upsampling involve improving enriching data, enhancing the appearance of images, as well as improving the precision of imaging filtering. Up sampling is often used to enhance a signal. If excessive data is included during upsampling, the image resolution deteriorates and becomes unsuitable for an enormous print. Up sampling may be beneficial if the picture is poor in quality, but not significantly.

3.6. Improved Resolution Engine (IRE)

It remains a desire to improve the images' quality or resolution. In this technique, the extracted features are enhanced or preserved while the image's resolution is raised. An innovative Aquila Gorilla Synergy Strategy (AGSS) is used for this, combining the best aspects of Artificial Gorilla Troops Optimization (AGTO) and Aquila Optimizer (AO).

The recently introduced technique, known as the Aquila Optimizer (AO), draws inspiration from the natural hunting behaviors of the Aquila. The process is segmented into four distinct stages: narrowed exploration involving gliding with contour flight, expanded exploitation through a low-flying descending attack, narrowed exploitation by strolling and securing prey, and expanded exploration achieved by soaring high with a vertical stoop. The AO algorithm leverages these behaviors to transition effectively from exploration to exploitation phases. In practice, the initial two-thirds of iterations are dedicated to exploration, while the final third focuses on exploitation. Figure 3.3 illustrates the phases of the Aquila Optimizer.

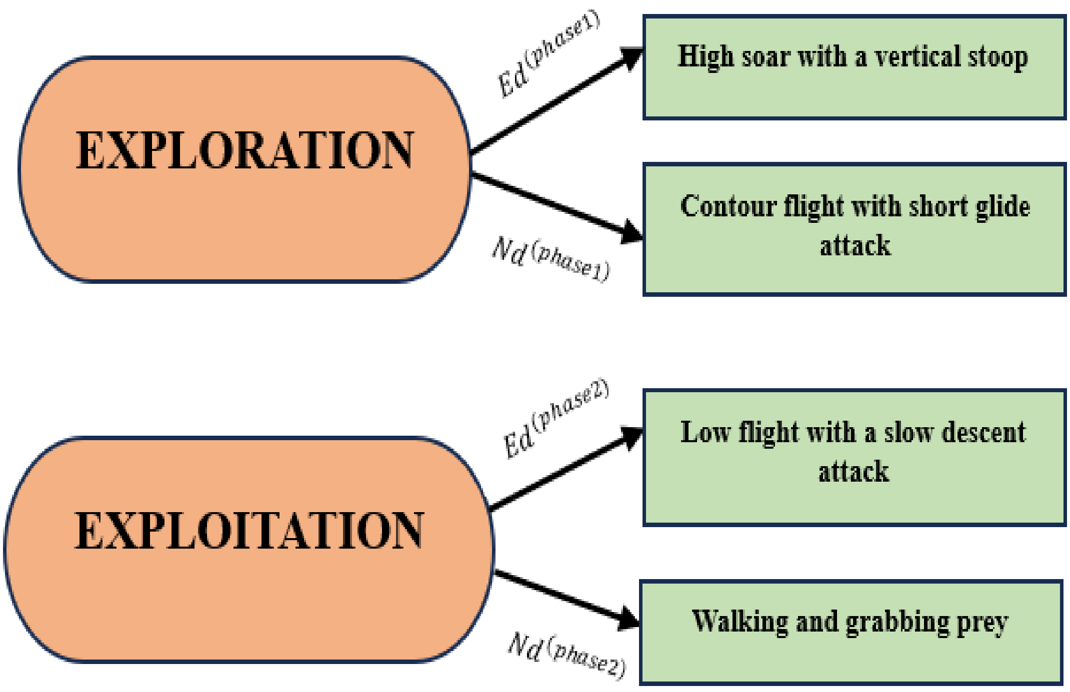


Fig 3.3: Phases of Aquilla Optimizer

The model is mathematically constructed by distributing N solutions within a D-dimensional search space, spanning a predefined range [L, U] during initialization.

$$Y_{p,q} = L_q + rdm \times (U_q - L_q) \tag{10}$$

Where $Y_{p,q}$ is the q^{th} dimension of p^{th} solution.

rdm is randomly selected value [0,1].

U_q, L_q are upper and lower bounds in the search area.

(1) Exploration phase (Phase 1)

The exploration phase of the AO has two behaviours. They are

- Expanded Exploration $[Ed^{(phase1)}]$
- Narrowed Exploration$[Nd^{(phase1)}]$

(i)$\boldsymbol{Ed^{(phase1)}}$

An aquila uses a high-altitude, vertical stoop to first identify the target zone and then select the best hunting spot. Because of this behaviour, the search space is surveyed from a great height in an attempt to determine the approximate location of the prey. This behaviour is replicated in AO to broaden the exploration and is carried out when $z < \left(\frac{2}{3} \times max_z\right)$ and $rdm < 0.5$.

$$Y_1(z+1) = Y_B(z) \times \left(1 - \frac{z}{max_z}\right) + (Y_M(z) - Y_B(z) \times rdm) \tag{11}$$

$$Y_M(z) = \frac{1}{S}\sum_{p=1}^{S} Y_i\,(z); q = 1,2,..,D \tag{12}$$

Where S, D are the solutions and search area's size.

$Y_1(z+1)$ is the solution for the next iteration.

$Y_B(z)$ is the best solution which indicates the position of prey.

$\left(1 - \frac{z}{max_z}\right)$ is used to control the degree of exploration based on the iteration count.

$Y_M(z)$ indicates the mean of currently available solutions.

(ii)$\boldsymbol{Nd}^{(phase1)}$

This involves the hunting behaviour known as contour flight during a brief glide attack. When an aquila is observed from a great height, it flies over its intended victim, gets ready to dive, and then attacks. The Aquila is able to narrowly examine a particular region thanks to its behaviour. This behaviour is simulated in when $z < \left(\frac{2}{3} \times max_z\right)$ and $rdm > 0.5$.in order to focus the investigation.

$$Y_2(z+1) = Y_B(z) \times levy(D) + (Y_{rdm}(z) + (a-b) \times rdm) \tag{13}$$

Where $Y_2(z+1)$ indicates the solutions of $Nd^{(phase1)}$

$Y_{rdm}(z)$ represents the randomly chosen solution

$$levy(D) = c \times \frac{f \times \sigma}{|g|^{\frac{1}{\beta}}} \tag{14}$$

$$\sigma = \frac{\Gamma(1+\beta) \times sin\left(\frac{\pi\beta}{2}\right)}{\Gamma\left(\frac{1+\beta}{2}\right) \times \beta \times 2^{\left(\frac{\beta-1}{2}\right)}} \tag{15}$$

$c = 0.01, \beta = 1.5$, $f$and g are random values [0,1].

$$a = w \times cos(\theta) \quad (16)$$

$$b = w \times sin(\theta) \quad (17)$$

$$w = rdm1 + H \times D_1 \quad (18)$$

$$\theta = -\omega \times D_1 \times \theta_1 \quad (19)$$

$$\theta_1 = \frac{3\pi}{2} \quad (20)$$

$rdm1$ ranges between [0,20] for a search cycles, H is a constant (0.00565), D represents a value from 1 to search area length, ω is constant (0.005)

(2) Exploitation phase (Phase 2)

The exploitation phase of the AO has two behaviours. They are

- Expanded exploitation $\left[Ed^{(phase2)}\right]$
- Narrowed exploitation$\left[Nd^{(phase2)}\right]$

$(\boldsymbol{iii})$ $\boldsymbol{Ed^{(phase2)}}$

Aquila uses the third approach in the expanded exploitation step to hunt prey. The aquila is ready to take off and attack since it has precisely located the prey zone. The Aquila descends vertically and launches the initial strike to see how its target will react. Low-flying descent attack is the term for this behaviour, which occurs when $z > \left(\frac{2}{3} \times max_z\right)$ and $rdm < 0.5$.

$$Y_3(z+1) = Y_B(z) - Y_M(z) \times \alpha - rdm + \left(\left(U_q - L_q\right) \times rdm + L\right) \times \delta \quad (21)$$

α, δ are parameters set to 0.1.

(iv) $\boldsymbol{Nd^{(phase2)}}$

When the aquila approaches its target and launches a haphazard attack, it employs the fourth hunting tactic in the narrower exploitation step. Walking and catching the prey is the behaviour that occurs when $z > \left(\frac{2}{3} \times max_z\right)$ and $rdm > 0.5$.

$$Y_4(z+1) = F_q(z) \times Y_B(z) - (P_1 \times Y(z) \times rdm) - P_2 \times Levy(D) + rdm \times P_1 \quad (22)$$

F_q represents the quality factor that balances the search strategy.

$$F_q(z) = z^{\frac{2\times rdm-1}{(1-max_z)^2}} \tag{23}$$

P_1, P_2 represents the Aquila's prey tracking movements during elope and decreasing from 2 to 0.

$$P_1 = 2 \times rdm - 1 \tag{24}$$

$$P_2 = 2 \times \left(1 - \frac{z}{max_z}\right) \tag{25}$$

Inspired by the natural behaviour of Aquila, the AO method has demonstrated competitive performance when applied to optimization problems, yielding great search efficiency and quick convergence times. Fig 3.4 shows the flowchart of AO.

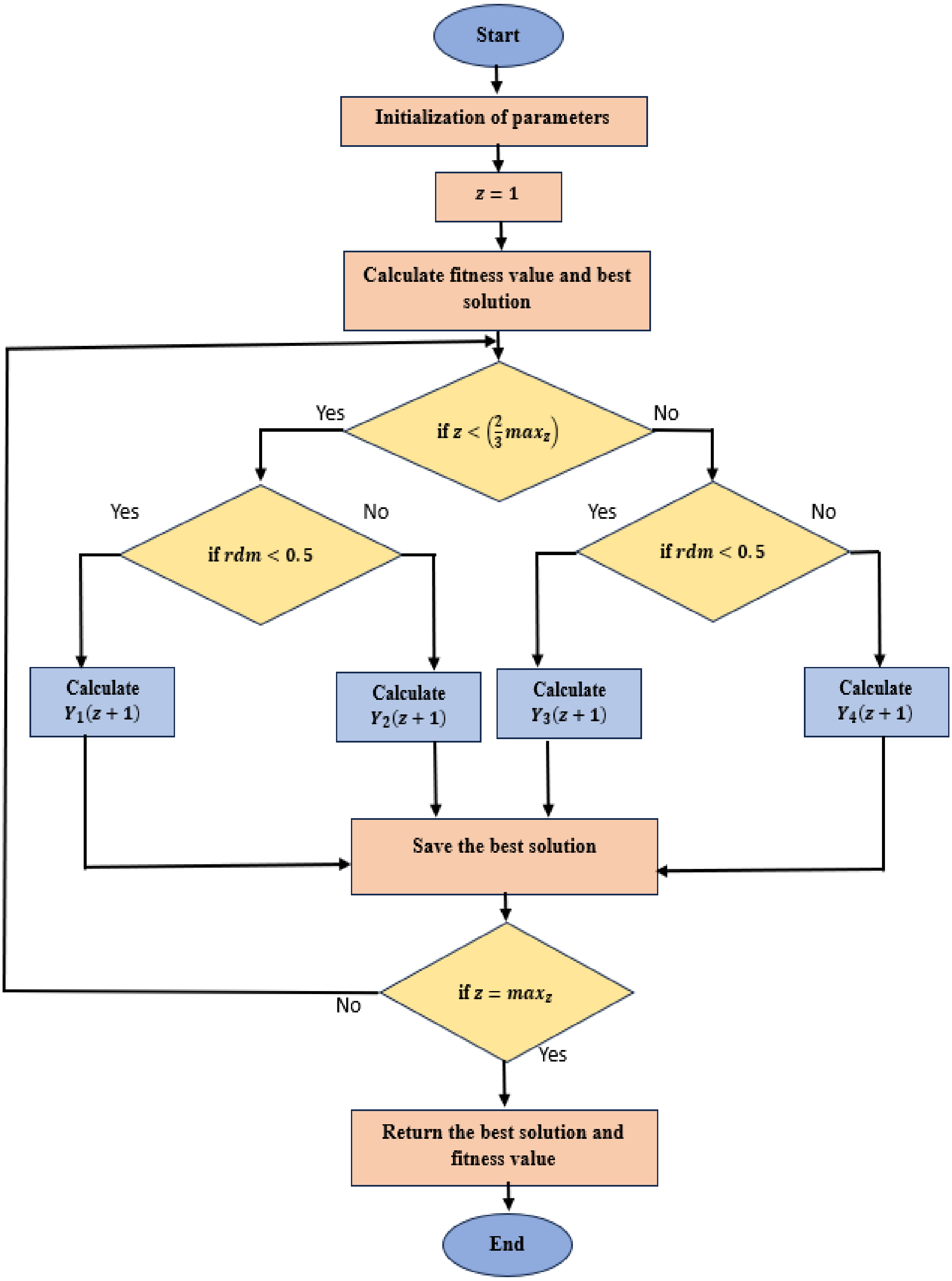


Fig 3.4: Flowchart of AO

The Artificial Gorilla Troop Optimization (AGTO) algorithm is inspired by the social and foraging behaviors of gorilla troops. This algorithm models how gorillas live and work together in groups, particularly when searching for food. The AGTO algorithm comprises five distinct techniques: the first three techniques are employed in the exploration phase, while the final two is applied during the exploitation phase. Figure 3.5 illustrates the phases of the AGTO algorithm.

1) Migration to unfamiliar territory promotes AGTO exploration.

(2) Changing to a different gorilla group improves the harmony between exploitation and exploration.

3) Moving in the direction of a well-known location improves the capacity to search across many optimization spaces.

4) Follow the silverback (the group leader who makes decisions and guides others), who maintains systematic and continuous searching among separate units to enable exploitation..

5) The competition for adult females, which resembles the gorillas' process of group enlargement and fighting throughout adolescence.

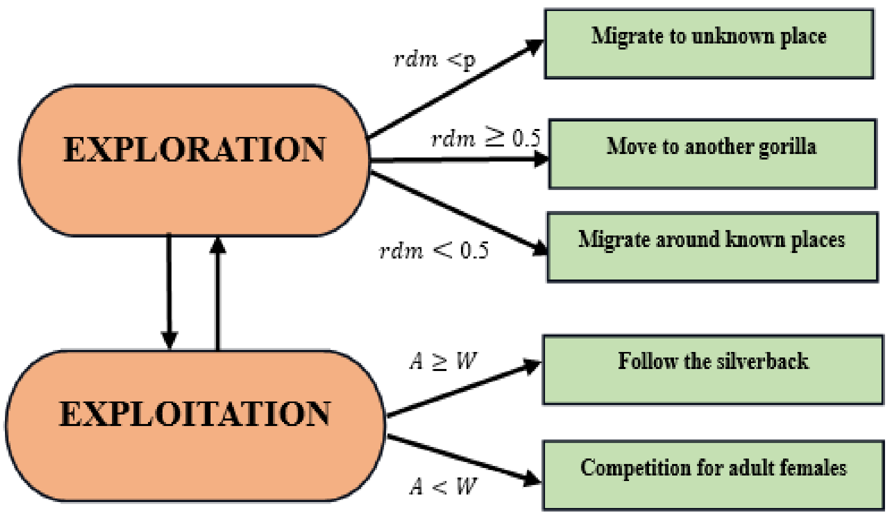


Fig 3.5: Phases of AGTO

(1) Exploration Phase

Three distinct techniques are employed in the exploration process: moving to a location that is not visible, moving in the direction of a recognized position, and moving in close proximity to other gorillas. Using a parameter called p, the method of migration to an unknown location is selected. The first mechanism is selected when *rand* < *p* is utilized. The gorilla-to- gorilla movement method is selected if rand is larger than or equal to 0.5. In contrast, the migration technique to a known site is selected if rand is smaller than 0.5. These can be expressed mathematically as follows:

$$Y_G(z+1) = \begin{cases} (U_b - L_b).rdm1 + L_b;\ rdm < p \\ (rdm2 - A).Y_{rdm}(z) + M.N;\ rdm \geq 0.5 \\ Y(i) - M.\left[M.\left(Y(z) - Y_{G_{rdm}}(z)\right) + rdm3.\left(Y(z) - Y_{G_{rdm}}(z)\right)\right];\ rdm < 0.5 \end{cases} \quad (26)$$

Where $Y(z)$ is the current location of Gorilla

$Y_G(z+1)$ is the candidate position vector in next iteration

$rdm1, rdm2, rdm3$ are random numbers [0.1]

U_b, L_b are upper and lower bounds

$Y(i)$ is an initial vector of gorilla position

p is a constant value set prior to the optimization in range [0,1].

Y_{rdm} is a gorilla selected randomly

$Y_{G_{rdm}}$ is a candidate gorilla selected randomly.

$$A = F.\left(1 - \frac{z}{max_{\ z}}\right) \quad (27)$$

$$F = cos(2.rdm4) + 1 \quad (28)$$

$$M = A.c \quad (29)$$

$z, max_{\ z}$ are present and maximum iterations

c is a random value [-1,1].

$$N = D.P(z) \quad (30)$$

$$D = [-C, C] \quad (31)$$

Following the exploration activity, a team growth session is carried out by assessing each of the Y options and, in the event that the cost is $Y(z) < P(z)$, using the $Y(z)$ solution as $P(z)$ solution. Consequently, the optimal solution discovered in this stage is called a silverback (S_b).

(2) Exploitation phase

During this stage, A is used to select one of two mechanisms: competing for adult females when $(A < W)$ or following the silverback when $(A \geq W)$, where W is a predetermined value.

- Follow the S_b

  The other male gorillas in the troop are also young and closely watch the S_b, who is a fit and healthy gorilla. Additionally, they follow all of Silverback's instructions to stay with him and visit various sites in quest of food supplies. This is computed using,

$$Y_G(z+1) = C.K.\left(Y(z) - Y_{S_b}\right) + Y(z) \tag{32}$$

$$K = \left(\left|\frac{1}{X}\sum_{i=1}^{X} Y_{Gi}(z)\right|^{h}\right)^{\frac{1}{h}} \tag{33}$$

$$h = 2^{C} \tag{34}$$

- Competition for Adult Females

  Once they reach teenage years, young gorillas compete dangerously with other males to attract adult females and grow their troop. These fights might include multiple people and go for several days. This is calculated using

$$Y_G(z+1) = Y_{S_b} - \left(Y_{S_b}.R - Y(z).R\right).T \tag{35}$$

$$R = 2.rdm5 - 1 \tag{36}$$

$$T = \beta.E \tag{37}$$

$$E = \begin{cases} X_1\,; rdm \geq 0.5 \\ X_2;\ rdm < 0.5 \end{cases} \tag{38}$$

Where R represents the force, E is the impact on violence.

T, β is the degree of violence

X_1 , X_2 are normal distribution values

When the exploitation phase comes to a complete a task, all of the Y_G solutions are compared. If $Y_G(z) < Y(z)$, then $Y_G(z)$replaces $Y(z)$, and is regarded as the best solution across the population.

The levy distribution function from the AO can be utilized in place of the normal distribution in the GTO to increase optimization efficiency. This algorithm balances the search strategies through the usage of the quality factor. Improved convergence, increased exploration, enhanced resilience, increased adaptation, enhanced handling of multimodal landscapes, and a balanced trade-off between exploration and exploitation are the outcomes of this.

$$Y_G(z+1) = Y_{S_b} - \left(Y_{S_b}.R - Y(z).R\right).T \times F_q(z) \tag{39}$$

$$T = \beta.E \tag{40}$$

$$E = \begin{cases} Levy_1\,(D); rdm \geq 0.5 \\ Levy_2(D)\,;\ rdm < 0.5 \end{cases} \tag{41}$$

Fig 3.6 shows the flowchart of AGTO.

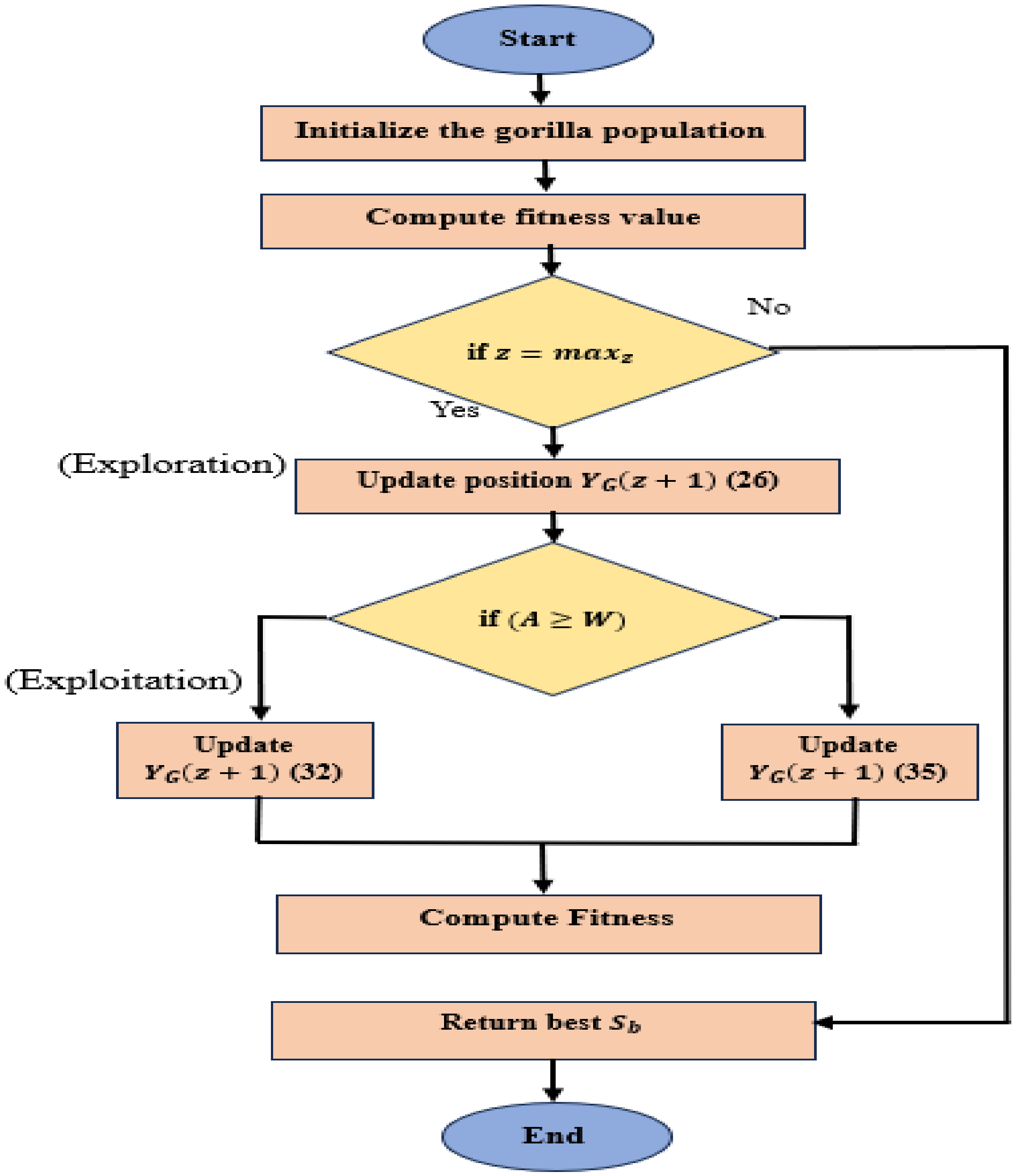


Fig 3.6: Flowchart of AGTO

3.5.1. Pseudocode of Aquila Gorilla Synergy Strategy

Start

Initialize AO and AGTO parameters ($z, max_z, S, D, Y, A, W, E$

If $z < \left(\frac{2}{3} \times max_z\right)$ and $rdm < 0.5$ (using 11)

Calculate $Y_1(z+1)$

End if

If $z < \left(\frac{2}{3} \times max_z\right)$ and $rdm > 0.5$ (using 13)

Calculate $Y_2(z+1)$

End if

If $z > \left(\frac{2}{3} \times max_z\right)$ and $rdm < 0.5$

Calculate $Y_3(z+1)$ (using 21)

End if

If $z > \left(\frac{2}{3} \times max_z\right)$ and $rdm > 0.5$

Calculate $Y_4(z+1)$ (using 22)

End if

Compute the Exploration phase of AGTO

$$Y_G(z+1) = \begin{cases} (U_b - L_b).rdm1 + L_b;\ rdm < p \\ (rdm2 - A).Y_{rdm}(z) + M.N;\ rdm \geq 0.5 \\ Y(i) - M.\left[M.\left(Y(z) - Y_{G_{rdm}}(z)\right) + rdm3.\left(Y(z) - Y_{G_{rdm}}(z)\right)\right];\ rdm < 0.5 \end{cases}$$

Compute the Exploitation phase of AGTO

If $(A \geq W)$ follows silverback

Calculate $Y_G(z+1)$ (using 32)

Else (Competition for adult females)

Calculate $Y_G(z+1)$ (using 35)

End if

Update $Y_G(z+1) = Y_{S_b} - \left(Y_{S_b}.R - Y(z).R\right).T \times F_q(z)$ (AGSS

Update $E = \begin{cases} Levy_1\ (D); rdm \geq 0.5 \\ Levy_2(D)\ ;\ rdm < 0.5 \end{cases}$

End

3.7. Output layer

This is the model's final phase, which generates the high-resolution images. In order to decrease the transformation between generated and ground truth images, enhancing images is therefore essential for synthesizing high-resolution images from each step of the models, including pre-processing, feature representations, fine-tuning details, and parameter optimization.

3.8 Summary

Enhancing images is a multidisciplinary to improve the images' visual appearance, suitability for human viewing, and compatibility with activities involving further analysis or processing. visual enhancement is essential in many fields because it strikes a careful balance between maintaining faithfulness to the original material and improving visual clarity. This paper proposed a novel model that reprocesses the photos using data augmentation, bi-lateral filtering, and normalization in order to increase image quality. Next, LBP is used to extract the features, and Real-ESRGAN is applied to achieve high-quality image super-resolution. Finally, the high-resolution improved images are acquired by applying the Aquila Gorilla Synergy Strategy.

CHAPTER 4

RESULT AND DISCUSSION

4.1 Introduction

This chapter results from proposed model on 2D image enhancement through hybrid optimization. Here, a detail discussion is provided about the datasets, process taken by the suggested design and the results attained by the proposed model. Subsequently, the suggested model attains better performance score in 2D image enhancement by a hybrid optimization technique. In this research, the performance of the proposed model is measured in terms of PSNR, RMSE, SSIM and MSE and the performance score of the proposed model is compared with the existing models such as Aquila optimizer (AO), Gorilla troops Optimization (GTO) and Lyrebird optimization (LO). The performance of the proposed model has been verified using the DIV2K database and the proposed model is implemented in python platform. Therefore, the suggested model's performance measurements show that it can improve 2D photos more effectively than the current models.

4.2 Dataset description

DIV2K is a well-known dataset used for single-image super-resolution, consisting of 1,000 varied scene images divided into 800 for training, 100 for validation, and 100 for testing. Developed for the NTIRE2017 and NTIRE2018 Super-Resolution Challenges, it aims to advance research in image super-resolution through more realistic degradation scenarios. Unlike the typical bicubic downscaling, this dataset includes multiple degradation types for creating low-resolution images across different challenge tracks. Notably, the 2017 Track 2 features x4 downscaling with unknown degradation, while the 2018 challenge introduces Track 2 and Track 4, which deal with realistic mild x4 and realistic wild x4 conditions, respectively. In realistic mild settings, images are degraded with motion blur, Poisson noise, and pixel shifting, while the wild conditions vary more significantly from one image to another.

4.3 Metrics evaluation

The performance of the proposed model can be evaluated in terms of PSNR, MSE, RMSE and SSIM respectively.

- ***PSNR***

PSNR is peak signal to noise ratio. PSNR is a metric used to measure the quality of an image or video reconstruction. It compares the original image to a compressed or reconstructed version, measuring the peak error in terms of signal-to-noise ratio. PSNR value can be generated based on MSE. However, the PSNR is deliberated using eqn. (1)

$$PSNR = 10log_{10}\frac{R^2}{MSE} \tag{1}$$

Where, MSE defines the mean square error and $R = 255$.

- ***SSIM***

SSIM stands for structural similarity index moreover, the function of this metrics is it predicts the quality of a given input image data. SSIM value of the image is generated based on three parameters such as luminance, contrast and structure among two images. Furthermore, the calculation of SSIM can be generated by using eqn. (2),

$$SSIM(u,v) = \frac{(2\propto_u \propto_v + A_1)(2\beta_{uv} + A_2)}{(\alpha_u^2 + \alpha_v^2 + A_1)(\beta_u^2 + \beta_v^2 + A_2)} \tag{2}$$

Where, $\propto_u$ is the average of u; $\propto_v$ is the average of v; α_u^2 is the variance of u; α_v^2 is the variance of v; β_{uv} is the covariance of u & v; A_1 = (X_1Y^2); A_2 = (X_2Y^2). Moreover, Y defines the dynamic range of the pixel values, and the values for X_1 & X_2 are 0.1 and 0.3 respectively.

- ***RMSE***

RMSE is Root Mean Squared Error and this metrics is used for measuring the differences among the input image and the output image moreover, the least RMS value shows that the model enhances the image more efficiently. RMSE of the model can be measured using eqn. (3),

$$RMSE = \sqrt{\sum_{i=1}^{n}\frac{(k_i - k)^2}{n}} \tag{3}$$

Where, k_i refers to the predicted values; k defines the observed values and the parameter n refers to the total number of observations.

- **MSE**

MSE defines the mean square error of the model and this metric is responsible for image quality estimation. Furthermore, the MSE represents the Cumulative Squared Error (CSE) among the compressed image and the input image. The MSE of the model can be generated by using eqn. (4),

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(k_i - k)^2 \quad (4)$$

4.4 Comparitive Analysis

The performance score attained by the proposed model can be compared with the existing models for validating the performance of suggested model. Here, the existing models such as Aquila optimizer (AO), Gorilla troops Optimization (GTO), and Lyrebird optimization (LO) are taken into consideration for performance comparison. Table 1 provides the comparison analysis of the proposed as well as the existing models. In the context of image quality assessment, PSNR is often presented as a numerical value in dB because it provides a more intuitive understanding of the quality of the reconstructed image relative to the original. On the other hand, metrics such as MSE (Mean Squared Error), RMSE (Root Mean Squared Error), and SSIM (Structural Similarity Index) are not inherently expressed in decibels. These metrics represent error or similarity measures that are typically normalized to a range between 0 and 1, making them more suitable for direct comparison as decimal values.

Table 4.1: Comparative analysis:

Model	PSNR	MSE	RMSE	SSIM
Proposed	**20.128**	**0.0234**	**0.0154**	**0.9813**
Aquila optimizer (AO)	**16.427**	**0.06121**	**0.0572**	**0.9322**
Gorilla troops Optimization (GTO)	**15.753**	**0.05602**	**0.0487**	**0.9534**
Lyrebird optimization (LO)	**13.514**	**0.04742**	**0.0365**	**0.9516**

**The following results were taken as an average of 50 images subsampled from the original input dataset

** The AGTO and AO was implemented from scratch without using any inbuilt function for the subsampled data set. See Appendix C

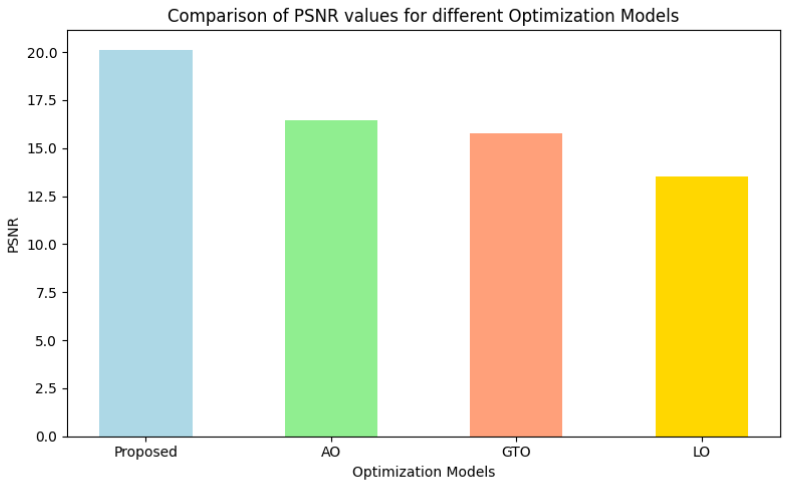


Figure 4.1: Comparison of PSNR

1. PSNR (Peak Signal-to-Noise Ratio):

• The PSNR (Peak Signal-to-Noise Ratio) value is typically expressed in decibels (dB), which is a logarithmic unit used to quantify the ratio of the peak signal power to the noise power in an image.

• The proposed model achieves a significantly higher PSNR value of 20.128 compared to Aquila Optimizer (16.427), Gorilla Troops Optimization (15.753), and Lyrebird Optimization (13.514).

• This indicates that the proposed model can preserve image quality better during compression or reconstruction, with less distortion and noise compared to the existing models.

• The superior PSNR of the proposed model suggests the incorporation of more effective compression algorithms, resulting in higher fidelity reconstructions.

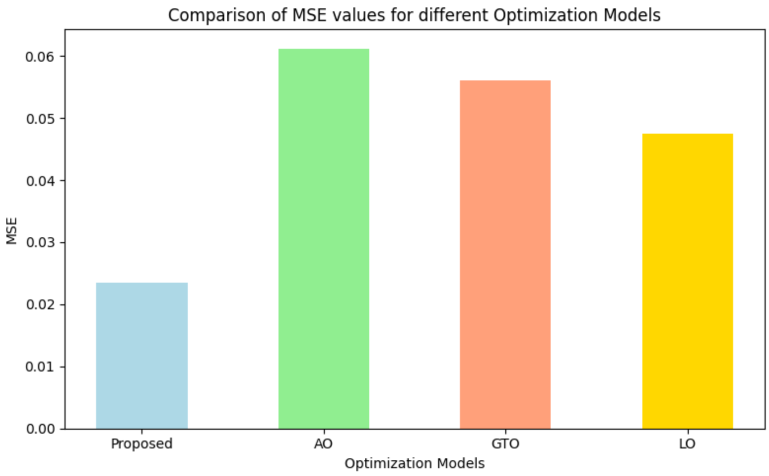


Figure 4.2: MSE comparison

2. MSE (Mean Squared Error):

• The proposed model demonstrates a lower MSE of 0.0234, indicating reconstructions with less overall error compared to Aquila Optimizer (0.06121), Gorilla Troops Optimization (0.05602), and Lyrebird Optimization (0.04742).

• This suggests that the proposed model can produce reconstructions closer to the original images, with fewer discrepancies and distortions.

• The reduced MSE of the proposed model reflects the effectiveness of its data processing mechanisms in minimizing reconstruction errors. Subsequently, the figure 4.3 shows the RMSE comparison of the proposed model as well as the existing models. On comparing the proposed model has resulted in lower RMSE score about 0.0154 and the RMSE of the AO model is about 0.0572 respectively. The existing model GTO attains 0.0487 of RMSE and the LO model attains 0.0365 of RMSE. Thus, the least score of RMSE is achieved by the proposed model indicates that the suggested model provides better presentation over image enhancement.

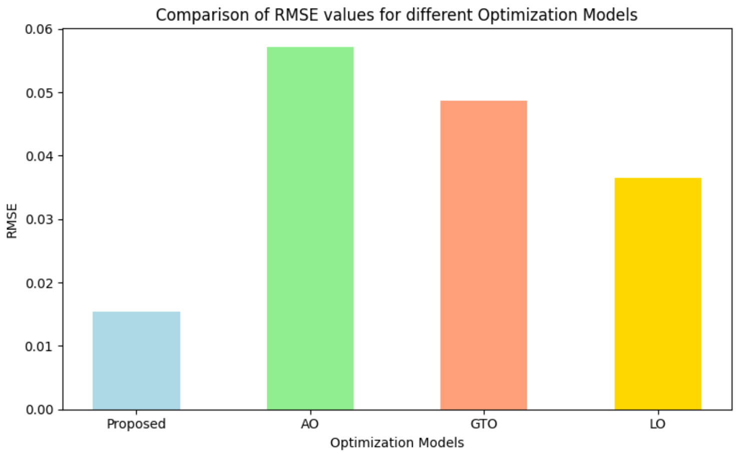


Figure 4.*3: RMSE comparison*

3. RMSE (Root Mean Squared Error):

- The proposed model exhibits a lower RMSE value of 0.0154, indicating reconstructions with less average deviation from the original images compared to Aquila Optimizer (0.0572), Gorilla Troops Optimization (0.0487), and Lyrebird Optimization (0.0365).
- This suggests that the proposed model can generate reconstructions that are closer in resemblance to the original images, with smaller deviations in pixel values.
- The reduced RMSE of the proposed model implies the incorporation of more accurate prediction mechanisms, better feature extraction capabilities, and finer parameter tuning during reconstruction.

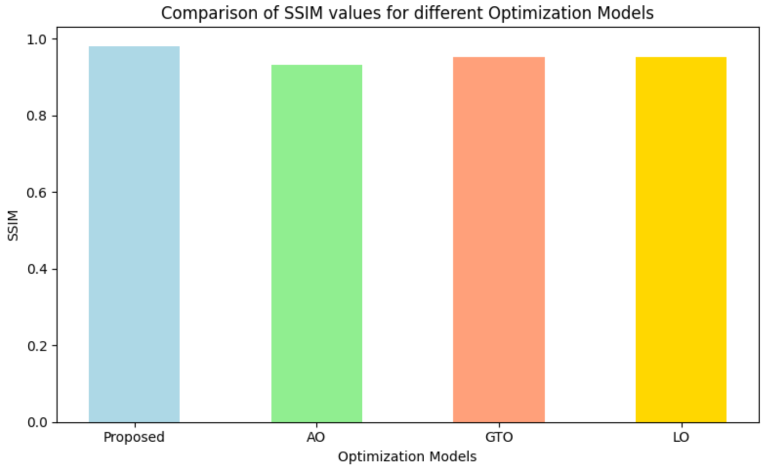


Figure 4.4: SSIM comparison

4. SSIM (Structural Similarity Index):

• The proposed model achieves a higher SSIM value of 0.9813 compared to Aquila Optimizer (0.9322), Gorilla Troops Optimization (0.9534), and Lyrebird Optimization (0.9516).

• This indicates that the proposed model better preserves structural information and overall image quality, leading to reconstructions that are more perceptually similar to the original images.

• The superior SSIM of the proposed model suggests the integration of advanced loss functions, attention mechanisms, and feature extraction techniques that enhance the perceptual quality of reconstructions.

In fact, based on the methodology and comparative analysis, the proposed model surpasses Aquila Optimizer, Gorilla Troops Optimization, and Lyrebird Optimization across all metrics, indicating its superiority in image reconstruction quality, accuracy, and preservation of structural details.

4.5 Performance analysis

- ***Pre-processing***

Generally, the input images are stored in computers as **matrixes** of numbers called as **pixel values**. Among them, every matrix having separate dimensional value, and the dimensional value is acquired by multiplying the height and width of the image. Moreover, the pixel value of each image is varied from 0 to 255 and represent the intensity of each pixel. In this research, the lower resolution images are chosen for input. The below figure 5(a) shows the input image and the figure 5(b) shows the filtered image and the input image is filtered by bilateral filters.

Figure 4.5(a): Input image

Figure 4.5(b): Bilateral fltered image

Moreover, by filtering the clarity of the images are enhanced and the image noise can be reduced. After image filtering, then the filtered image is given as input for normalization. With normalization, all the images are getting data limits. The following figures 5 (c) shows the input filtered image and the figure 5(d) shows the normalized image.

Figure 4.5(c): Input bilateral filtered image

Figure 4.5(d): Normalized image

Furthermore, image augmentation techniques such as image rotation, image flipping and color jitter are applying to the normalized image. The normalized input is rotated in angles of 30°, 45° as well as 60° and the resultant rotation images are given as figure 6. Figure 6(a) is the normalized input image figure 6(b) shows the 30°rotated image figure 6(c) shows the 45° rotated image and the figure 6(d) shows the 60° rotated image.

Figure 4.6(a): Normalized input image

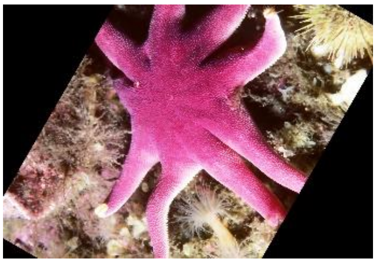


Figure 4.6(b): 35° rotated image

Figure 4.6(c): 45° rotated image

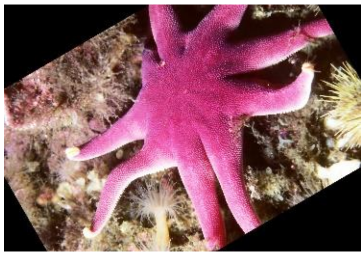


Figure 4.6(d): 60° rotated image

A flipped image is a static or moving image that is generated by a mirror-reversal of an original image over a horizontal axis and turning the image upside down. Similarly, another flipped image is mirrored over the vertical axis, as in a conventional mirror image. Subsequently, the images are flipped horizontally and vertically and the horizontal flipping images are shown in figure 4.7 and the vertical image flipping is deliberated in figure 4.8.

Figure 4.7: Horizontal flipping

Figure 4.8: Vertical image flipping

Once the images are flipped horizontally and vertically then the colors are added to the lipped images. Color jitter is a technique which comes under image data augmentation mechanism and in this process the images are getting more brightness, contrast, hue and high saturation.

The horizontally flipped color jitter images are show in figure 9 and the vertically flipped color jitter images are shown in figure 4.10.

Figure 4.9: Horizontally flipped colour jittered images

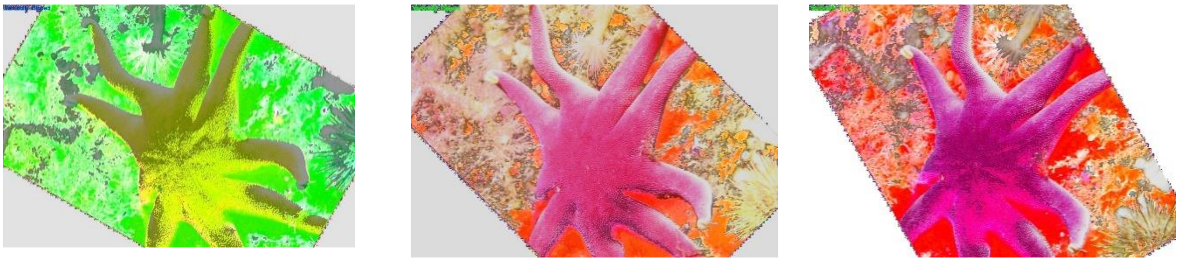


Figure 4.10: Vertically flipped colour jittered images

- ***Feature extraction***

The raw input data is converted into numerical features that may be processed while maintaining the information in the original database in order to achieve a higher performance score. It performs more effectively than using Machine Learning (ML) techniques directly to raw data.

Feature extraction mechanism plays a vital role in detection of features such as image edges and image shapes. In this study, local binary patterns (LBP) are used to obtain the image features. LBP is thought to be a useful method for obtaining the texture features from the image database. The texture descriptor of an image then uses the range of the current pixel to threshold the pixels of the neighboring images. Furthermore, compared to other models, the LBP process extracts features more effectively and captures the images' grayscale contrast and local spatial patterns (LSP). In this suggested model color and texture features are extracted. Figure 4.11 depicts the feature extraction by horizontally flipped images moreover, the figure 4.12 shows the feature extraction by vertically flipped images.

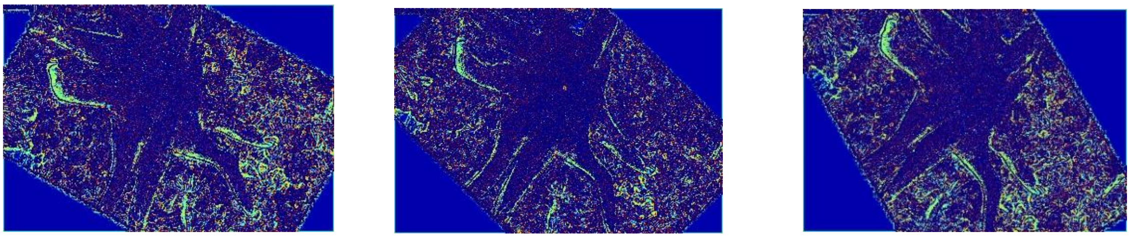


Figure 4.11: Feature extraction by horizontally flipped images

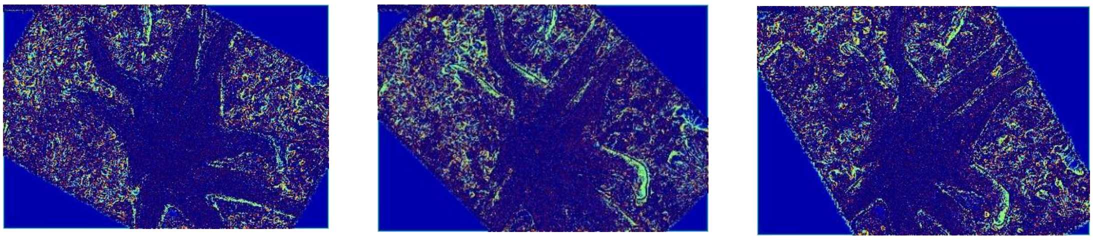


Figure 4.12: Feature extraction by vertically flipped images

- ***High degree modelling (HDM)***

For attaining better high-definition performance scores, the proposed model uses the basic concepts of Real-ESRGAN. The function of Real-ESRGAN is it reduces the image texture characteristics smoothly. There are two main causes for image texture degradation. The first drawback of using HDM is that it heavily deteriorates images, which has the opposite effect on images thus it does not deteriorate equally as in real life. Second, because of the special fundamental mechanism of GAN, training the model becomes much more difficult if dealing with a complex and high-intensity image degradation technique. Here, HDM function can be developed for horizontally and vertically flipped images. Moreover, the resultant output images attained by HDM is illustrated as figure 4.13.

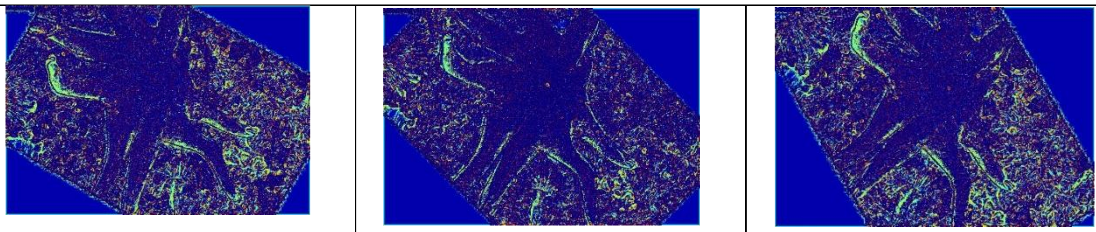


Figure 4.13: HDM images by horizontally flipped images

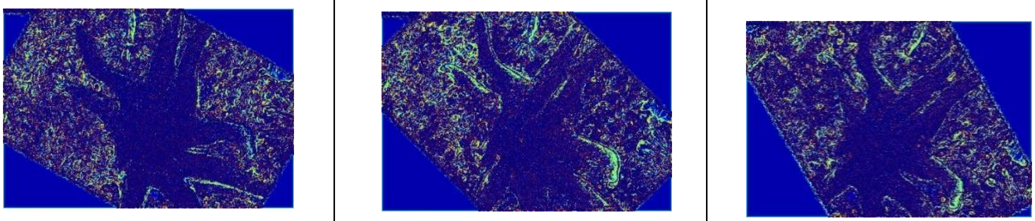


Figure 4.14: HDM images by vertically flipped images

An enhanced version of ESRGAN called Real-ESRGAN can identify its shortcomings, such compression removal, and offers a more workable approach to real-world image restoration. When comparing ESRGAN with Real-ESRGAN, it can be seen that ESRGAN has a bigger degradation space, making its structure unsuitable for image identification. But for complicated training outputs, the enhanced ESRGAN requires a better discriminative capability. Apart from this, the Real-ESRGAN model provides better performance for non-standard features as well as the local features.

- **Improved Resolution Engine (IRE):**

Image or video resolution is the main parameter used to evaluate the quality of images or movies. Pixel count determines an image's resolution; a greater pixel count indicates a higher resolution and the potential to convey more visual information. As a result, processing high-resolution photos performs better and produces better results than processing low-resolution images. To improve the image resolution, a hybrid optimization approach is created in this study. Artificial Gorilla Troops Optimization (AGTO) and Aquila Gorilla Synergy Strategy (AGSS) are used to create the hybrid optimization algorithm. Consequently, the effectiveness and adaptability of the hybrid optimization led to higher resolution photos. The hybrid optimization has also resulted in an improvement in image resolution. The hybrid optimization method has resulted in an improvement in both convergence efficiency and image resolution. Furthermore, the suggested hybrid technique can manage gradient-based constraints. Here, the IRE process can be carried out by the vertically and horizontally flipped images. The resultant vertically and horizontally flipped images are provided in figure 4.15 and figure 4.16 respectively.

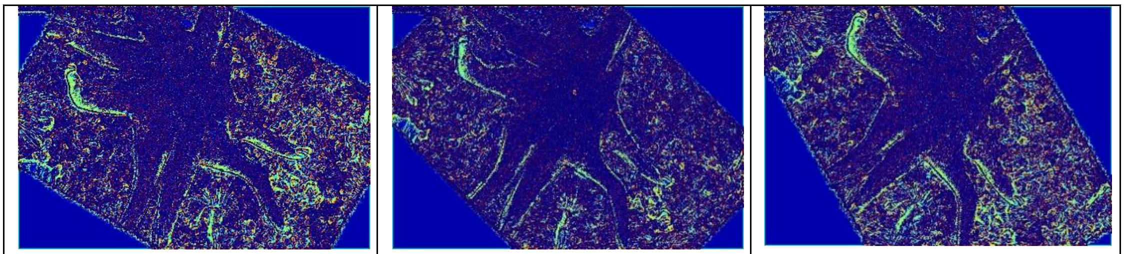


Figure 4.15: IRE by horizontally flipped images

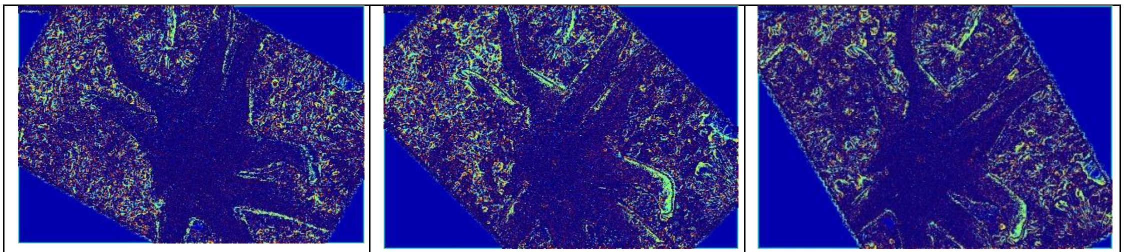


Figure 4.16: IRE by vertically flipped images

4.6 Conclusion

In this section, the outcomes that the proposed model produces can be quantified, and DIV2K can be utilized to validate the performance of the model. Furthermore, compared to the current models, the recommended model achieves a better performance score. The AO, GTO, and LO models are the three existing models that are considered for comparison. The performance is measured using four separate metrics, and the performance score can be compared to the models that are currently in use. Out of all of them, the suggested model performs better than the current models. The model's final output photos are then attached and subject to performance analysis. The first step in pre-processing the images is to reduce noise and enhance clarity. Next, all the images are receiving data ranges, and the output image is identified as a normalized image. After that, a few picture augmentation techniques—such as image rotation and image flipping—are performed by the suggested model using the photos themselves. Here, the photos are rotated at corresponding angles of 35, 45, and 60 degrees in addition to being flipped horizontally and vertically. The flipped images are then subjected to color jitter, which adds more color to the images. Furthermore, the Furthermore, the flipped images are used to recover the texture features, and local binary patterns (LBP) are used to extract the features. The output images that are produced are shown in Figures 4.11 and 4.12. HDM is applied to the flipped images in order to improve performance. Figures 4.13 and 4.14 below show the outputs that the model was able to achieve. Lastly, the proposed model for improving the image resolution implements the IRE mechanism. Figures 4.15 and 4.16 display the resolution-enhanced pictures, respectively. Figure 4.17 depicts the final output of the project.

Figure 4.17: Final Enhanced Image

CHAPTER 5

CONCLUSION

5.1 Conclusion

A hybrid optimization for the improvement of 2D images has been devised in this study. The primary goal of this research is to increase the performance of the proposed model over the current models while also improving image resolution. The introduction to improving image resolution in 2D images was covered in detail in chapter 1. Additionally, the second chapter discusses about the limitations present in the existing models as well as the most recent work on the subject. The proposed model is presented in Chapter 3. For image enhancement, a brand-new Hybrid-Adaptive Super-Resolution Model (HASR-Net) model is offered. Improving digital image resolution is the primary goal of this research in order to get higher performance. To improve convergence speed and stability, a hybrid optimization combines the effectiveness of Artificial Gorilla Troops Optimization with the adaptability of Aquilla Optimizer. Raw picture data is supplied to the input phase of the suggested model, and only lower resolution data is accepted for processing at this stage. The input database then completes the pre-processing. After that, the input data base is selected for the data augmentation process, which includes image flipping, image rotation based on angles, and color jittering. When rotating images, the angles used to rotate them are 35, 45, and 60 degrees, respectively. In addition, the photos undergo vertical and horizontal flips in preparation for additional processing. The method of adding color to the images is called color jittering, and it is done after that to improve resolution detection.

After that, the images are undergoing pre-processing with bilateral filters, which enhance image clarity and reduce noise in the images. The pre-processed photos are then fed through the normalization procedure, which gives each image in the input database a data range. Furthermore, data normalization is defined as the application of data ranges to the images. Numerous scientific fields have expressed great interest in feature extraction approaches. The application of the Local Binary Patterns (LBP) approach emphasizes the importance of feature extraction. LBP provides a stable way to extract texture information from unprocessed data, especially for image processing applications. LBP effectively recovers texture elements from the images by segmenting texture elements based on pixel intensity comparisons. This method promotes robustness against noise and light changes while simultaneously increasing computation efficiency. The model may efficiently reduce redundant information and enhance classification precision by giving priority to the extraction of crucial as well as distinguishing qualities. This could improve the classification outcomes in many fields of study. While texture features use groups of pixels, color features simply use individual pixels. For each pixel in the feature maps, a Local Binary Pattern (LBP) is calculated. In this work, we produce high-quality super-resolution of images by using Real-ESRGAN. The images' reconstructions by Real-ESRGAN were unduly smooth and lacked image texture features. An enhanced version of ESRGAN, called Real-ESRGAN, successfully tackles issues such removing obtrusive compression and provides a more practical technique for real-world image restoration. The discriminator architecture employed in ESRGAN is inadequate because Real-ESRGAN addresses a substantially wider degradation space than ESRGAN. For complicated training outputs, the discriminator in Real-ESRGAN needs to have a larger discriminative capacity. It must ensure an accurate gradient response for local textures in addition to distinguishing between global styles. The VGG-style discriminator in ESRGAN was enhanced to a U-Net design with skip connections. The U-Net generates realness values for each pixel so that the generator can receive exact responses for each pixel. Meanwhile, the U-Net architecture and complicated degradations provides poor stability. Regularizing spectral normalization can help stabilize training dynamics. Furthermore, spectral normalization mitigates the annoyance and hyperacuity associated with GAN training. These adjustments should make it easy for Real-ESRGAN training to strike a better balance between local detail enhancement and artifact reduction. The resolution or quality of the photographs is enhanced by the hybrid technique. This method increases the image's resolution without compromising the retrieved features. This is accomplished by integrating the finest features of Aquila Optimizer (AO) and Artificial Gorilla Troops Optimization (AGTO) into a novel strategy called Aquila Gorilla Synergy Strategy (AGSS). The recently suggested approach, known as the Aquila Optimizer (AO), is consistent with the way Aquilas naturally hunt. Consequently, the performance of the proposed model can be validated by using DIV2K dataset. Four distinct criteria are used to quantify the performance of the proposed model, and its evaluation is contrasted with that of the current models. Among these, the suggested model outperformed the current models in terms of performance. Consequently, the proposed model's findings demonstrate that it may effectively improve image resolution and performance score.

5.2 Future scope

Enhancing images is intended to make them easier for human viewers to interpret or perceive the information they contain, or to provide as better input for other automated image processing methods. The technique of altering digital photographs to make the results more appropriate for display or additional image analysis is known as image enhancement. To make it simpler to recognize important aspects in an image, you can, for instance, brighten, sharpen, or remove noise. Though image processing has advanced significantly, there are still plenty of intriguing opportunities ahead. As artificial intelligence, augmented reality, and other technological breakthroughs proliferate, image processing will become increasingly important in revolutionizing our understanding of and interactions with the outside world.

References

1. Abdullah-Al-Wadud, M., Kabir, M.H., Dewan, M.A.A. and Chae, O., 2007. A dynamic histogram equalization for image contrast enhancement. IEEE transactions
on consumer electronics, 53(2), pp.593-600.
2. Al Sobbahi, R. and Tekli, J., 2022. Comparing deep learning models for low-light natural scene image enhancement and their impact on object detection and classification: Overview, empirical evaluation, and challenges. Signal Processing: Image Communication, p.116848.
3. Ancuti, C.O., Ancuti, C., De Vleeschouwer, C. and Bekaert, P., 2017. Color balance and fusion for underwater image enhancement. IEEE Transactions on image processing, 27(1), pp.379-393.
4. Arif, R.B., Khan, M.M.R. and Siddique, M.A.B., 2018, December. Digital image enhancement in Matlab: An overview on histogram equalization and specification. In 2018 International Conference on Innovation in Engineering and Technology (ICIET) (pp. 1-6). IEEE.
5. Bhatti, U.A., Yu, Z., Chanussot, J., Zeeshan, Z., Yuan, L., Luo, W., Nawaz, S.A., Bhatti, M.A., Ain, Q.U. and Mehmood, A., 2021. Local similarity-based spatial–spectral fusion hyperspectral image classification with deep CNN and Gabor filtering. IEEE Transactions on Geoscience and Remote Sensing, 60, pp.1-15.
6. Cai, J., Gu, S. and Zhang, L., 2018. Learning a deep single image contrast enhancer from multi-exposure images. IEEE Transactions on Image Processing, 27(4), pp.2049-2062.
7. Chen, Y., Tao, J., Liu, L., Xiong, J., Xia, R., Xie, J., Zhang, Q. and Yang, K., 2022. Research of improving semantic image segmentation based on a feature fusion model. Journal of Ambient Intelligence and Humanized Computing, 13(11), pp.5033-5045.
8. Chen, Y.S., Wang, Y.C., Kao, M.H. and Chuang, Y.Y., 2018. Deep photo enhancer: Unpaired learning for image enhancement from photographs with gans. In

Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 6306-6314).

9. Chitradevi, B. and Srimathi, P., 2014. An overview on image processing techniques. International Journal of Innovative Research in Computer and Communication Engineering, 2(11), pp.6466-6472. 66
10. Dar, K.A. and Mittal, S., 2021. An enhanced adaptive histogram equalization based local contrast preserving technique for HDR images. In IOP Conference Series: Materials Science and Engineering (Vol. 1022, No. 1, p. 012119). IOP Publishing.
11. Dhariwal, S., 2011. Comparative analysis of various image enhancement techniques. International Journal on Electronics and Communication Technology, 2(3), pp.91-95.
12. Fan, R., Li, X., Lee, S., Li, T. and Zhang, H.L., 2020. Smart image enhancement using CLAHE based on an F-shift transformation during decompression. Electronics, 9(9), p.1374.
13. Fu, X., Liao, Y., Zeng, D., Huang, Y., Zhang, X.P. and Ding, X., 2015. A probabilistic method for image enhancement with simultaneous illumination and reflectance estimation. IEEE Transactions on Image Processing, 24(12), pp.4965- 4977.
14. Greenacre, M., Groenen, P.J., Hastie, T., d'Enza, A.I., Markos, A. and Tuzhilina, E., 2022. Principal component analysis. Nature Reviews Methods Primers, 2(1), p.100.
15. Guo, X., Li, Y. and Ling, H., 2016. LIME: Low-light image enhancement via illumination map estimation. IEEE Transactions on image processing, 26(2), pp.982-993.
16. Guo, X., Li, Y. and Ling, H., 2016. LIME: Low-light image enhancement via illumination map estimation. IEEE Transactions on image processing, 26(2), pp.982-993.
17. Huang, B., Yang, F., Yin, M., Mo, X. and Zhong, C., 2020. A review of multimodal medical image fusion techniques. Computational and mathematical methods in medicine, 2020.
18. Islam, M.J., Xia, Y. and Sattar, J., 2020. Fast underwater image enhancement for improved visual perception. IEEE Robotics and Automation Letters, 5(2), pp.3227-3234.

19. Janani, P., Premaladha, J. and Ravichandran, K.S., 2015. Image enhancement techniques: A study. Indian Journal of Science and Technology, 8(22), pp.1-12.
20. Jintasuttisak, T. and Intajag, S., 2014, October. Color retinal image enhancement by Rayleigh contrast-limited adaptive histogram equalization. In 2014 14th international conference on control, automation and systems (ICCAS 2014) (pp. 692-697). IEEE.
21. Kaur, H., Koundal, D. and Kadyan, V., 2021. Image fusion techniques: a survey. 67 Archives of computational methods in Engineering, 28, pp.4425-4447.
22. Khudhair, K.T., Najjar, F.H., Waheed, S.R., Al-Jawahry, H.M., Alwan, H.H. and Al-khaykan, A., 2023, February. A novel medical image enhancement technique based on hybrid method. In Journal of Physics: Conference Series (Vol. 2432, No.1, p. 012021). IOP Publishing.
23. Kumar, M. and Chandrakar, C., 2011. An Overview of Image enhancement techniques. International Journal of technology, pp.15-19.
24. Li, C., Guo, C., Ren, W., Cong, R., Hou, J., Kwong, S. and Tao, D., 2019. An underwater image enhancement benchmark dataset and beyond. IEEE Transactions on Image Processing, 29, pp.4376-4389.
25. Li, C.Y., Guo, J.C., Cong, R.M., Pang, Y.W. and Wang, B., 2016. Underwater image enhancement by dehazing with minimum information loss and histogram distribution prior. IEEE Transactions on Image Processing, 25(12), pp.5664-5677.
26. Li, M., Liu, J., Yang, W., Sun, X. and Guo, Z., 2018. Structure-revealing low-light image enhancement via robust retinex model. IEEE Transactions on Image Processing, 27(6), pp.2828-2841.
27. Liu, S., Long, W., He, L., Li, Y. and Ding, W., 2021. Retinex-based fast algorithm for low-light image enhancement. Entropy, 23(6), p.746.
28. Liu, Y., Liu, S. and Wang, Z., 2015. Multi-focus image fusion with dense SIFT. Information Fusion, 23, pp.139-155.
29. Lore, K.G., Akintayo, A. and Sarkar, S., 2017. LLNet: A deep autoencoder approach to natural low-light image enhancement. Pattern Recognition, 61, pp.650-662.

30. Nadipally, M., 2019. Optimization of methods for image-texture segmentation using ant colony optimization. In Intelligent data analysis for biomedical applications (pp. 21-47). Academic Press.
31. Ningsih, D.R., 2020. Improving retinal image quality using the contrast stretching, histogram equalization, and CLAHE methods with median filters. International Journal of Image, Graphics and Signal Processing, 10(2), p.30.
32. Nixon, M. and Aguado, A., 2019. Feature extraction and image processing for computer vision. Academic press.
33. Oktay, O., Ferrante, E., Kamnitsas, K., Heinrich, M., Bai, W., Caballero, J., Cook, S.A., De Marvao, A., Dawes, T., O'Regan, D.P. and Kainz, B., 2017. Anatomically 68 constrained neural networks (ACNNs): application to cardiac image enhancement and segmentation. IEEE transactions on medical imaging, 37(2), pp.384-395.
34. Panayides, A.S., Amini, A., Filipovic, N.D., Sharma, A., Tsaftaris, S.A., Young, A., Foran, D., Do, N., Golemati, S., Kurc, T. and Huang, K., 2020. AI in medical imaging informatics: current challenges and future directions. IEEE journal of biomedical and health informatics, 24(7), pp.1837-1857.
35. Peli, T. and Lim, J.S., 1982. Adaptive filtering for image enhancement. Optical Engineering, 21(1), pp.108-112.
36. Puniani, S. and Arora, S., 2015. Performance evaluation of image enhancement techniques. International Journal of Signal Processing, Image Processing and Pattern Recognition, 8(8), pp.251-262.
37. Qi, Y., Yang, Z., Sun, W., Lou, M., Lian, J., Zhao, W., Deng, X. and Ma, Y., 2021. A comprehensive overview of image enhancement techniques. Archives of Computational Methods in Engineering, pp.1-25.
38. Qi, Y., Yang, Z., Sun, W., Lou, M., Lian, J., Zhao, W., Deng, X. and Ma, Y., 2021. A comprehensive overview of image enhancement techniques. Archives of Computational Methods in Engineering, pp.1-25.
39. Rajput, S. and Suralkar, S.R., 2013. Comparative study of image enhancement techniques. International Journal of Computer Science and Mobile Computing, 2(1), pp.11-21.

40. Razzak, M.I., Naz, S. and Zaib, A., 2018. Deep learning for medical image processing: Overview, challenges and the future. Classification in BioApps: Automation of Decision Making, pp.323-350.
41. Ren, H., Kheradmand, A., El-Khamy, M., Wang, S., Bai, D. and Lee, J., 2020. Real-world super-resolution using generative adversarial networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (pp. 436-437).
42. Reza, A.M., 2004. Realization of the contrast limited adaptive histogram equalization (CLAHE) for real-time image enhancement. Journal of VLSI signal processing systems for signal, image and video technology, 38, pp.35-44.
43. Sajjadi, M.S., Scholkopf, B. and Hirsch, M., 2017. Enhancenet: Single image super-resolution through automated texture synthesis. In Proceedings of the IEEE international conference on computer vision (pp. 4491-4500).
44. Saxena, D. and Cao, J., 2021. Generative adversarial networks (GANs) challenges, 69 solutions, and future directions. ACM Computing Surveys (CSUR), 54(3), pp.1-42.
45. Shi, Z., Chen, Y., Gavves, E., Mettes, P. and Snoek, C.G., 2021. Unsharp mask guided filtering. IEEE Transactions on Image Processing, 30, pp.7472-7485.
46. Singh, G. and Mittal, A., 2014. Various image enhancement techniques-a critical review. International Journal of Innovation and Scientific Research, 10(2), pp.267-274.
47. Singh, G. and Mittal, A., 2014. Various image enhancement techniques-a critical review. International Journal of Innovation and Scientific Research, 10(2), pp.267-274.
48. Sreeja, P. and Hariharan, S., 2018. An improved feature-based image fusion technique for enhancement of liver lesions. Biocybernetics and Biomedical Engineering, 38(3), pp.611-623.
49. Tian, C., Zheng, M., Zuo, W., Zhang, B., Zhang, Y. and Zhang, D., 2023. Multi- stage image denoising with the wavelet transform. Pattern Recognition, 134, p.109050.
50. Wang, K.J., 2020. On a High-pass filter described by local fractional derivative. Fractals, 28(03), p.2050031.

51. Wang, X., Yu, K., Wu, S., Gu, J., Liu, Y., Dong, C., Qiao, Y. and Change Loy, C., 2018. Esrgan: Enhanced super-resolution generative adversarial networks. In Proceedings of the European conference on computer vision (ECCV) workshops (pp. 0-0).
52. Wang, X., Yu, K., Wu, S., Gu, J., Liu, Y., Dong, C., Qiao, Y. and Change Loy, C., 2018. Esrgan: Enhanced super-resolution generative adversarial networks. In Proceedings of the European conference on computer vision (ECCV) workshops (pp. 0-0).
53. Wu, X.F., Hu, S.G., Zhao, J., Li, Z.M., Li, J., Tang, Z.J. and Xi, Z.F., 2014. Comparative analysis of different methods for image enhancement. Journal of central south university, 21(12), pp.4563-4570.
54. Yuan, Z., Liu, D., Zhang, X. and Su, Q., 2020. New image blind watermarking method based on two-dimensional discrete cosine transform. Optik, 204, p.164152.
55. Zamir, S.W., Arora, A., Khan, S., Hayat, M., Khan, F.S., Yang, M.H. and Shao, L., 2020. Learning enriched features for real image restoration and enhancement. In Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 70 23–28, 2020, Proceedings, Part XXV 16 (pp. 492-511). Springer International Publishing.
56. Zhang, W., Liu, Y., Dong, C. and Qiao, Y., 2019. Ranksrgan: Generative adversarial networks with ranker for image super-resolution. In Proceedings of the IEEE/CVF International Conference on Computer Vision (pp. 3096-3105).
57. Zhang, Z., Zhang, X., Peng, C., Xue, X. and Sun, J., 2018. Exfuse: Enhancing feature fusion for semantic segmentation. In Proceedings of the European conference on computer vision (ECCV) (pp. 269-284).
58. Zhao, M., Yang, R., Hu, M. and Liu, B., 2024. Deep Learning-Based Technique for Remote Sensing Image Enhancement Using Multiscale Feature Fusion. Sensors, 24(2), p.673.

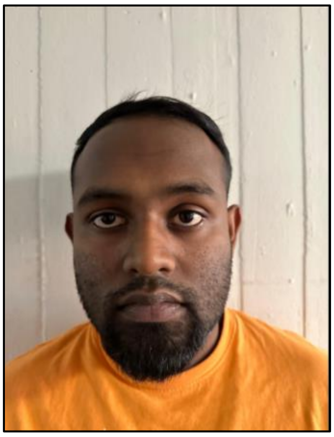
Appendices

Appendix A: Justification for Bilateral Filter Parameters

In this thesis, bilateral filtering is applied as a pre-processing step to reduce noise in images before further processing. Bilateral filtering is chosen due to its capability to reduce noise while preserving edges, which is crucial for maintaining the quality of high-resolution images in 2D image enhancement tasks.

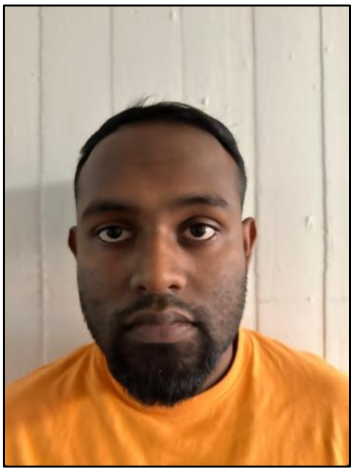
Input image:

- **d=3**:



  - **Inference**: some smoothening of the facial features, the edges (facial contours) are preserved, some details (reduced colouring of the nose) are lost.
  - **Outcome**: Minimal noise reduction; high detail preservation.

- **d=9**:



  - **Inference:** Enhanced smoothening of facial features while maintaining the integrity of facial contours. More effective noise reduction compared to **d=3** balanced preservation of major details and more visually pleasing image without significantly compromising on detail retention.
  - **Outcome**: Effective balance between noise reduction and detail preservation.
  - **Justification**: Provides the optimal compromise necessary for a wide range of images, effectively smoothing out noise without sacrificing the sharpness and clarity of significant textures and edges.

- **d=11**:

  - **Inference:** Considerable smoothening of facial features with a significant reduction in visible noise. While facial contours remain recognizable, the image exhibits some over-smoothing that leads to a loss of finer details, such as subtle textures and finer strands of hair. This results in a cleaner yet softer appearance, where some critical sharpness is sacrificed for more pronounced noise reduction.
  - **Outcome**: Strong noise reduction; potential loss of fine details due to over-smoothing.

- **Why *d=9*:** Offers an excellent middle ground by reducing noise sufficiently while retaining essential details, making it suitable for both artistic and practical applications. It ensures images remain naturally sharp and clean, making ***d=9*** a versatile choice for diverse image processing needs.

Appendix B: Size of Dataset & Proof of concept

Initial Dataset = **800** images

After rotating 30°, 45° & 60° = **2400** images

After flipping horizontally and vertically, jittering = **4800** images

```
Epoch 1/10: 100%|■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■
■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■| 1200/1200 [6:05:57<00:00, 18.30s/it]
Epoch 1/10, Loss: 0.058348653830374436
Epoch 2/10: 100%|■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■
■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■| 1200/1200 [10:13:18<00:00, 30.67s/it]
Epoch 2/10, Loss: 0.0015511104368973367
Epoch 3/10: 100%|■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■
■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■| 1200/1200 [7:16:32<00:00, 21.83s/it]
Epoch 3/10, Loss: 0.0008500203500443604
Epoch 4/10: 100%|■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■
■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■| 1200/1200 [6:00:09<00:00, 18.01s/it]
Epoch 4/10, Loss: 0.0005118572584857854
Epoch 5/10: 100%|■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■
■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■| 1200/1200 [6:31:43<00:00, 19.59s/it]
Epoch 5/10, Loss: 0.00037461696039827076
Epoch 6/10: 100%|■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■
■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■| 1200/1200 [10:41:14<00:00, 32.06s/it]
Epoch 6/10, Loss: 0.0002918760226445253
Epoch 7/10: 100%|■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■
■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■| 1200/1200 [16:35:43<00:00, 49.79s/it]
Epoch 7/10, Loss: 0.0002067630006604304
Epoch 8/10: 100%|■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■
■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■| 1200/1200 [5:03:02<00:00, 15.15s/it]
Epoch 8/10, Loss: 0.00018494705772961121
Epoch 9/10: 100%|■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■
■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■| 1200/1200 [5:03:35<00:00, 15.18s/it]
Epoch 9/10, Loss: 0.00013813424855925406
Epoch 10/10: 100%|■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■
■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■| 1200/1200 [5:04:25<00:00, 15.22s/it]
Epoch 10/10, Loss: 0.00012026954243992804
```

Appendix C: Optimizer setup blocks

Gorilla Troops Optimizer block

```
import numpy as np

from PIL import Image, ImageEnhance, UnidentifiedImageError

import os

from skimage.metrics import peak_signal_noise_ratio as psnr

from skimage.metrics import mean_squared_error as mse

from skimage.metrics import structural_similarity as ssim


class GTO:

    def __init__(self, population_size=50, num_variables=1, max_iterations=100, mutation_rate=0.1):

        self.population_size = population_size

        self.num_variables = num_variables

        self.max_iterations = max_iterations

        self.mutation_rate = mutation_rate


    def initialize_population(self):

        return np.random.uniform(low=0.5, high=1.5, size=(self.population_size, self.num_variables))


    def fitness_function(self, solution):

        return (solution - 1) ** 2  # Simple example where 1 is the ideal solution


    def gorilla_movement(self, current_solution, best_solution):

        return current_solution + np.random.uniform(low=-0.1, high=0.1, size=current_solution.shape)
* self.mutation_rate


    def update_population(self, population):

        pass


    def optimize(self):

        population = self.initialize_population()
```

```
        for _ in range(self.max_iterations):
            fitness_values = np.array([self.fitness_function(solution) for solution in population])
            best_index = np.argmin(fitness_values)
            best_solution = population[best_index]
            for i in range(self.population_size):
                population[i] = self.gorilla_movement(population[i], best_solution)
            self.update_population(population)
        return best_solution


class ImageQualityOptimizer:
    def __init__(self, optimizer, max_mse=65025):  # max_mse based on (255^2) for the worst case
scenario
        self.optimizer = optimizer
        self.max_mse = max_mse


    def process_folder(self, input_folder, output_folder):
        metrics = {'psnr': [], 'mse': [], 'rmse': [], 'ssim': []}
        os.makedirs(output_folder, exist_ok=True)
        filenames = [f for f in os.listdir(input_folder) if f.lower().endswith(('.png', '.jpg', '.jpeg'))]
        for filename in filenames:
            input_path = os.path.join(input_folder, filename)
            output_path = os.path.join(output_folder, filename)
            try:
                input_image = Image.open(input_path)
                best_params = self.optimizer.optimize()
                enhancer = ImageEnhance.Brightness(input_image)
                enhanced_image = enhancer.enhance(best_params[0])


                enhanced_image.save(output_path)
                input_np_array = np.array(input_image)
                enhanced_np_array = np.array(enhanced_image)
```

```
            psnr_value = psnr(input_np_array, enhanced_np_array)
            mse_value = mse(input_np_array, enhanced_np_array) / self.max_mse
            rmse_value = np.sqrt(mse_value)  # Normalizing after sqrt gives a better range
            ssim_value = (ssim(input_np_array, enhanced_np_array, multichannel=True) + 1) / 2  #
Adjusted to range [0,1]

            metrics['psnr'].append(psnr_value)
            metrics['mse'].append(mse_value)
            metrics['rmse'].append(rmse_value)
            metrics['ssim'].append(ssim_value)
          except UnidentifiedImageError:
            print(f"Cannot identify image file: {input_path}")
            continue

        # Calculate average metrics
        avg_metrics = {k: np.mean(v) for k, v in metrics.items()}
        return avg_metrics

# Example usage
if __name__ == "__main__":
    optimizer = GTO()
    quality_optimizer = ImageQualityOptimizer(optimizer)
    input_folder = r('C:\Users\nchez\Downloads\Thesis Fin\HDM')
    output_folder = r('C:\Users\nchez\Downloads\Thesis Fin\GTO')
    average_metrics = quality_optimizer.process_folder(input_folder, output_folder)
    print("Average Image Quality Metrics:")
    for metric, value in average_metrics.items():
        print(f"{metric.upper()}: {value:.3f}")
```

Aquila Optimizer block

```
import numpy as np
from PIL import Image, ImageEnhance, UnidentifiedImageError
import os
from skimage.metrics import peak_signal_noise_ratio as psnr
from skimage.metrics import mean_squared_error as mse
from skimage.metrics import structural_similarity as ssim


class AquilaOptimizer:
    def __init__(self, learning_rate=0.01, max_iterations=100):
        self.learning_rate = learning_rate
        self.max_iterations = max_iterations


    def optimize(self, initial_value=1.0):
        # Start from an initial value and optimize towards an enhancement factor
        value = initial_value
        for _ in range(self.max_iterations):
            gradient = -0.1 * (value - 1)  # Example gradient; in real use, calculate based on actual data or
function
            value += self.learning_rate * gradient
        return max(0.5, min(2.0, value))  # Clamping the result to ensure it stays within reasonable
bounds


class ResolutionEngine:
    def __init__(self, optimizer):
        self.optimizer = optimizer


    def process_images(self, input_folder, output_folder):
        os.makedirs(output_folder, exist_ok=True)
        filenames = [f for f in os.listdir(input_folder) if f.lower().endswith(('.png', '.jpg', '.jpeg'))]
        metrics = {'psnr': [], 'mse': [], 'rmse': [], 'ssim': []}
```

```
max_mse = 65025  # Normalization factor for MSE

for filename in filenames:
    input_path = os.path.join(input_folder, filename)
    output_path = os.path.join(output_folder, filename)

    try:
        image = Image.open(input_path)
        enhancement_factor = self.optimizer.optimize()
        enhancer = ImageEnhance.Sharpness(image)
        enhanced_image = enhancer.enhance(enhancement_factor)

        enhanced_image.save(output_path)

        np_original = np.array(image)
        np_enhanced = np.array(enhanced_image)

        psnr_value = psnr(np_original, np_enhanced)
        mse_value = mse(np_original, np_enhanced) / max_mse
        rmse_value = np.sqrt(mse_value)
        ssim_value = (ssim(np_original, np_enhanced, multichannel=True) + 1) / 2

        metrics['psnr'].append(psnr_value)
        metrics['mse'].append(mse_value)
        metrics['rmse'].append(rmse_value)
        metrics['ssim'].append(ssim_value)
    except UnidentifiedImageError:
        print(f"Cannot identify image file {input_path}")
        continue

avg_metrics = {key: np.mean(values) for key, values in metrics.items()}
```

```
        return avg_metrics

# Example usage
optimizer = AquilaOptimizer(learning_rate=0.05)
resolution_engine = ResolutionEngine(optimizer)

input_folder = r('C:\Users\nchez\Downloads\Thesis Fin\HDM’)
output_quality_folder = r(‘C:\Users\nchez\Downloads\Thesis Fin\AO’)

# Process and enhance images
avg_metrics = resolution_engine.process_images(input_folder, output_quality_folder)
print("Average Metrics:", avg_metrics)
```

Lyrebird Optimization block

```
import numpy as np
from PIL import Image, ImageEnhance, UnidentifiedImageError
import os
from skimage.metrics import peak_signal_noise_ratio as psnr
from skimage.metrics import mean_squared_error as mse
from skimage.metrics import structural_similarity as ssim

class LyrebirdOptimizer:
    def __init__(self, population_size=10, num_generations=50, mutation_rate=0.1):
        self.population_size = population_size
        self.num_generations = num_generations
        self.mutation_rate = mutation_rate

    def initialize_population(self):
        return np.random.uniform(0.5, 1.5, self.population_size)
```

```python
    def fitness_function(self, individual):
        return 1 / (np.abs(individual - 1) + 0.01)  # Fitness favors values close to 1

    def select_parents(self, population, fitness):
        fitness_total = sum(fitness)
        selection_probs = [f / fitness_total for f in fitness]
        return np.random.choice(population, 2, p=selection_probs)

    def crossover(self, parent1, parent2):
        return (parent1 + parent2) / 2

    def mutate(self, individual):
        if np.random.rand() < self.mutation_rate:
            individual += np.random.uniform(-0.1, 0.1)
        return min(max(individual, 0.5), 1.5)  # Clamping to ensure it stays within reasonable bounds

    def optimize(self):
        population = self.initialize_population()
        for _ in range(self.num_generations):
            fitness = [self.fitness_function(ind) for ind in population]
            next_generation = []
            for _ in range(self.population_size // 2):
                parent1, parent2 = self.select_parents(population, fitness)
                child1 = self.mutate(self.crossover(parent1, parent2))
                child2 = self.mutate(self.crossover(parent1, parent2))
                next_generation.extend([child1, child2])
            population = next_generation
        best_index = np.argmax(fitness)
        return population[best_index]

class ResolutionEngine:
```

```
def __init__(self, optimizer):
    self.optimizer = optimizer

def process_images(self, input_folder, output_folder):
    os.makedirs(output_folder, exist_ok=True)
    filenames = [f for f in os.listdir(input_folder) if f.lower().endswith(('.png', '.jpg', '.jpeg'))]
    metrics = {'psnr': [], 'mse': [], 'rmse': [], 'ssim': []}
    max_mse = 65025  # Normalization factor for MSE

    for filename in filenames:
        input_path = os.path.join(input_folder, filename)
        output_path = os.path.join(output_folder, filename)
        try:
            image = Image.open(input_path)
            enhancement_factor = self.optimizer.optimize()
            enhancer = ImageEnhance.Sharpness(image)
            enhanced_image = enhancer.enhance(enhancement_factor)
            enhanced_image.save(output_path)

            np_original = np.array(image)
            np_enhanced = np.array(enhanced_image)
            psnr_value = psnr(np_original, np_enhanced)
            mse_value = mse(np_original, np_enhanced) / max_mse
            rmse_value = np.sqrt(mse_value)
            ssim_value = (ssim(np_original, np_enhanced, multichannel=True) + 1) / 2

            metrics['psnr'].append(psnr_value)
            metrics['mse'].append(mse_value)
            metrics['rmse'].append(rmse_value)
            metrics['ssim'].append(ssim_value)
        except UnidentifiedImageError:
```

```
            print(f"Cannot identify image file {input_path}")
            continue

    avg_metrics = {key: np.mean(values) for key, values in metrics.items()}
    return avg_metrics

# Example usage
optimizer = LyrebirdOptimizer()
resolution_engine = ResolutionEngine(optimizer)
input_folder = 'path_to_input_folder'
output_folder = 'path_to_output_folder'
avg_metrics = resolution_engine.process_images(input_folder, output_folder)
print("Average Image Quality Metrics:", avg_metrics)
```

❖---------------The End---------------❖