Communications on Applied Nonlinear Analysis
ISSN: 1074-133X
Vol 31 No. 2s (2024)

Application of Mathematical Modelling and Deep Learning in Image Restoration using Edge Preservation Method

Bhumika Neole[1], Harshali Zodpe[2], Sharda Mahajan[3], Purushottam Shobhane[4], Yoginee Pethe[5], Pravin Karmore[6]

[1]Department of Electronics and Communication Engineering, Shri Ramdeobaba College of Engineering and Management, Nagpur, India

neoleba@rknec.edu

[2]Dr. Vishwanath Karad MIT World Peace University, Pune, India

harshali.zodpe@mitwpu.edu.in

[3]Research scholar, Shri Ramdeobaba College of Engineering and Management, Nagpur, India

shardamahajan1708@gmail.com

[4]Symbiosis Institute of Technology, Nagpur Campus, Symbiosis International (Deemed University), Pune, India

purushottam.shobhane@sitnagpur.siu.edu.in

[5,6]Department of Computer Applications, Shri Ramdeobaba College of Engineering and Management, Nagpur, India

pethey@rknec.edu, karmorepy@rknec.edu

Article History:

Received: *26-03-2024*

Revised: *15-05-2024*

Accepted: *27-05-2024*

Abstract:

The image restoration has witnessed significant advancements with the integration of deep learning techniques with offering unparalleled capabilities in enhancing image quality. This paper study proposed a novel approach to image restoration by incorporating a sophisticated edge preservation method using the deep learning framework. The method aims to address the challenge of preserving high-frequency details, such as edges, while restoring images from various forms of degradation. Also we investigate the use of deep neural networks trained on a different noisy image to restore a clean image with preserving edges of original image. The Deep convolutional neural network (DCNNs), and ResNet50 in deep learning model learns elaborate patterns and features, enabling the reconstruction of images with improved clarity and fidelity. The proposed Deep Convolution Neural Network and ResNet50 methods are designed to restore image content with intelligently preserve and enhance edge information, crucial for maintaining the structural integrity of the original scene. The proposed model is trained on diverse datasets, encompassing a wide range of image degradations, ensuring robust performance across various real-world scenarios. Experimental results demonstrate the efficacy of the proposed approach in comparison to existing methods, showcasing superior edge preservation and overall restoration quality. This research contributes to the advancement of image processing techniques, offering a powerful tool for applications such as medical imaging, satellite imagery, and digital photography, where maintaining fine details is essential for accurate interpretation and analysis.

Keywords: Image restoration, Deep learning, Edge preservation, Deep Convolutional neural network, ResNet50.

1. INTRODUCTION

Deep learning has become a very useful tool in recent years, with the ability to improve picture quality like never before. Using a complex edge preservation method within the framework of deep learning, this study shows a new way to restore images. Adding deep learning methods to image repair has made huge strides forward, which is an important part of image processing [1]. The main objective is to solve the problem of keeping high-frequency features, especially edges, while fixing pictures that have been damaged in different ways. When images are damaged in some way, like by noise, blurring, or other types of degrading, traditional methods of image repair often fail to keep the fine features that make up the images. Convolutional neural networks (CNNs), a type of deep learning, have shown potential in helping to solve these problems. DCNNs are very good at understanding complex patterns and features from big datasets. This [2] lets them rebuild pictures that are clearer and more accurate. We present a Convolutional Deep Neural Network architecture designed for picture repair, with a focus on keeping and improving edge information in a smart way. Protecting the edges of a picture is very important for keeping its structure and general look good. Edges tell you important things about the lines between different things and areas in a picture, which is a big part of how you understand the scene. Normal methods for repair, on the other hand, often lose important visual clues when they smooth out or change these lines as they work. We want to get around these problems with the suggested method, which uses deep neural networks to bring back picture content while smartly keeping and improving edge information. The architecture is meant to learn the complicated connections between input and output images. This lets the model find and protect important edge features while the repair is happening. This smart edge protection is very important for things like medical imaging, digital photography, and satellite images where small details are very important [3].

Image restoration methods are very important for improving the look of pictures that have lost quality because of things like noise, fuzz, or compression. The goal of these methods is to get back to a picture's original, undistorted content. This [4] makes the image clearer and more accurate. In traditional ways of restoring images, filters like Wiener or Gaussian filters are often used to lessen the effects of degradation. But these methods might not be able to keep small features, especially high-frequency parts like edges. In recent years, a lot of progress has been made in picture repair by using deep learning methods, especially convolutional neural networks (CNNs). Deep learning models are very good at picking out complex patterns and traits from big datasets. This [6] lets them rebuild pictures that are much better than before. These models are good at generalizing to a wide range of real-world situations and can adapt to different types of degradation. Also, new research has been focusing on adding edge preservation techniques to deep learning frameworks to solve the problem of keeping important features during the repair process. This all-around method is a big change in the way images are recovered; it can keep small features and make sure that the general quality of restored pictures is unmatched. It's clear that image repair methods are very important in many fields, like medical imaging, satellite data, and digital photography, where correctness and clarity of visual information are very important.

One important [7] new idea in this study is that the deep neural network is trained on a variety of noisy pictures so that it can recover a clean image with edges that have been damaged in a number of different ways. This method makes it possible for the model to work well in a wide range of real-life situations, providing reliable results in all of them. Using a variety of datasets improves the learning process and lets the model react to the different kinds and amounts of picture degradation that are common in real life. A lot of tests were done to make sure the suggested method would work, and the results were compared to those of other picture repair methods. It is a deep convolutional neural network (CNN) design that is known for how deep it is. This depth lets it learn complex picture patterns and features. In the field of image repair, ResNet50 can be used to learn how to map a damaged picture to a clean image. Edge-aware loss functions or regularization terms are often added to the training process to help keep the edges. These terms punish changes from the original edge structure, which pushes the network to produce results with sharp edges. The network learns to tell the difference between noise and real edges during training. This lets it recover images well while keeping important edge information. This method works especially well in situations where edge features are important, like medical images or monitoring. To restoring images using deep learning and preserving edges with ResNet50 is a strong method that can greatly enhance the quality of damaged images, making them better for a wide range of uses.

The review [8] tools look at how well the edges are kept, how well the repair works overall, and how well it works with different kinds of picture degradation. The experiment shows that the proposed way works better than the others, especially when it comes to keeping edges. This shows that deep learning methods can be useful for picture repair jobs where keeping small features is very important. The importance of this study goes beyond academic inquiry because it helps to improve picture processing methods in ways that are useful in real life. The suggested method is very useful in areas like medical imaging, where keeping small details is important for correct reading, and in satellite imagery and digital photography, where accuracy in the images' appearance is very important for analysis. If you can recover pictures with better edge preservation, it can completely change applications that need to understand and analyze visual data with precision and accuracy. When [9] deep learning, especially convolutional neural networks (CNNs), is combined with a complex edge retention method, it makes a big step forward in the field of picture processing. This part talks about the paper's main achievements, including how it has helped picture repair methods get better and how it could be used in different areas.

- This study describes a new way to restore images by mixing deep learning techniques such as DCNN, ResNet50 with a smart approach for keeping the edges of the images.
- The suggested DCNN an ResNet 50 Methods to keeps and improves edge data during the picture repair process, which fixes a problem that many current methods have.
- The model is trained on a variety of datasets, which makes sure that it works well in a wide range of real-life situations and with different types of picture degradation.
- The proposed system architecture design uses DCNN, ResNet50 to learn complicated connections between input and output pictures, making sure that high-frequency features that are important for visual accuracy are kept.

2. RELATED WORK

In the past few years, picture repair has gotten a lot of attention, especially when deep learning and edge preservation methods are used together. Many works that are linked to these methods help us understand and improve them. This shows how picture processing and repair are changing. The use of deep learning models for picture reconstruction is an important area of linked work. Dong et al.'s SRCNN (Super-Resolution Convolutional Neural Network) was one of the first studies to use convolutional neural networks (CNNs) in single-image super-resolution [10]. This work showed that deep learning models can learn complicated mappings from low-resolution to high-resolution pictures, which shows how they could be used to improve images. Later research built on these ideas to deal with other types of picture decline, like fuzz and noise.

In the context of keeping edges, [11] suggested using a perception loss function, which looks at high-level features taken from networks that have already been trained, to improve edge information during picture reconstruction. Their method successfully maintained structure features, including lines, by using perception information. This led to better visual clarity. This method showed how important it is to use advanced loss functions to direct the learning process toward keeping traits that are perceptually important. In a different area of research, people are looking into how to combine traditional picture processing methods with deep learning methods. For example, [12] created a deep convolutional neural network that works with total variation regularization to fix images and separate them into groups. This project combined the best features of deep learning for feature learning with the regularization skills of old-fashioned methods. It showed how these two techniques might work well together. In terms of using a variety of datasets, [13] worked on training deep neural networks on a range of pictures to make them better at adaptation. The network got better at fixing images with different kinds and amounts of damage by showing the model different kinds of decline that happen in real life. This method is similar to the one we suggested in our work, which stresses how important it is to use a variety of training samples to get good results.

In addition, study has been focused on certain areas where picture repair is very important. For example, Maier et al. looked into how deep learning could be used to rebuild computed tomography (CT) images in the medical field, focusing on keeping small structures that are important for making accurate diagnoses. In the same way, [14] suggested a deep learning-based method for super-resolution in satellite data to deal with the problems that come with satellite pictures' low spatial resolution. Edge-preserving filters and methods are also being looked into as part of this work. Some authors' work, like [15] on bidirectional filtering and [16] on edge-preserving smoothing filters, helps us understand how old methods can be used in the current world of deep learning-based repair, especially when it comes to keeping edges. These works all add to the progress of image repair techniques. They include cutting edge deep learning models for image improvement, new loss functions, the combination of standard methods, and domain-specific uses. This study crosses many fields, showing how it could be useful in many areas, from medical imaging and satellite images to regular digital photography, where keeping edge information is important for correct analysis and understanding.

Table 1: Summary of different work in Image restoration

Method	Dataset Used	Finding	Limitation	Scope
SRCNN (Super-Resolution CNN) [17]	General image datasets	Demonstrated the efficacy of deep learning for single-image super-resolution.	Limited scalability to diverse degradation types and noise levels.	Exploring adaptability to various real-world scenarios.

Perceptual Loss Function [18]	Diverse image datasets	Improved edge preservation by incorporating high-level features in the loss function.	Sensitivity to hyperparameters and computational intensity.	Further optimization for real-time applications.
Joint Image Restoration and Segmentation [19]	Mixed datasets (natural and medical images)	Combined deep learning with total variation regularization for joint restoration and segmentation.	Challenges in balancing regularization and feature learning.	Investigating applications in medical image analysis.
Diverse Training Datasets [20]	Comprehensive image datasets	Enhanced model robustness by training on diverse images with various forms of degradation.	Computational demands for training on large and diverse datasets.	Exploring strategies for efficient utilization of diverse datasets.
CT Image Reconstruction [21]	Medical imaging datasets	Applied deep learning for computed tomography image reconstruction, preserving fine structures for accurate diagnosis.	Limited to a specific medical imaging modality.	Adapting the approach to other medical imaging modalities.
Super-Resolution for Satellite Imagery [21]	Satellite image datasets	Developed a deep learning method for super-resolution in satellite imagery, addressing spatial resolution challenges.	Dependency on high-quality reference images for training.	Investigating adaptability to different satellite imaging scenarios.
Bilateral Filtering [22]	General image datasets	Leveraged bilateral filtering for edge-preserving smoothing.	Susceptibility to parameter tuning challenges and potential artifacts.	Refining parameters for diverse image characteristics.
Edge-Preserving Smoothing Filters [23]	Diverse datasets	Explored guided, median, and adaptive mean filters for edge-preserving smoothing.	Limited effectiveness in scenarios with complex textures or specific noise types.	Investigating hybrid approaches for improved edge preservation.
Sparse Coding-Based Methods [24]	Varied datasets	Utilized sparse coding for edge preservation through dictionary learning.	Sensitivity to parameter tuning and potential computational overhead.	Investigating adaptive parameter strategies for improved performance.
Fusion of Deep Learning and Traditional Methods [25]	Mixed datasets	Combined deep learning with traditional methods, leveraging the strengths of both approaches.	Potential challenges in achieving optimal synergy between different methodologies.	Exploring additional hybrid frameworks for enhanced restoration.
Non-Local Means (NLM) [12]	General image datasets	Implemented non-local means for patch-based	Computationally intensive, particularly for large image sizes.	Investigating optimizations for

		denoising, effectively preserving edges.		real-time applications.
Anisotropic Diffusion [13]	Diverse datasets	Utilized anisotropic diffusion for edge-preserving smoothing, adapting to local gradient information.	May introduce staircase artifacts and struggle with certain image characteristics.	Exploring adaptive variations for improved performance in specific scenarios.

3. DATASET DESCRIPTION

The PIRM (Perceptual Image Restoration and Manipulation) dataset is made up of 200 carefully chosen pictures that have been carefully split into two equal groups for testing and approval. This dataset has a lot of different kinds of information about a lot of different things, like people, things, places, plants, and nature scenery. Including such a wide range of material in the dataset makes it very flexible, which makes it perfect for testing how well picture repair and editing algorithms work in a variety of situations.

Figure 1: Sample images of Dataset

One interesting thing about the PIRM collection is that the picture sizes vary, but the resolutions are usually around 300,000 pixels. This range of sizes and material types adds another level of complexity, but it also lets methods be fully tested and proven in real-life situations. Image processing researchers and professionals can use this information to get a better idea of the problems that come up in a wide range of fields, from photography to computer vision. In conclusion, the PIRM dataset is a useful tool for studying and improving perceptual images. It gives a practical and varied test environment for creating and testing algorithms that aim to improve and change visual content.

4. EDGE PRESERVATION METHODS

1. Bilateral Filtering:

Bilateral filtering is a spatial-domain filtering technique that preserves edges by considering both spatial closeness and intensity similarity. It adds a weight term that makes smoothing pixel values with different levels less desirable. Bilateral filtering saves time and space on computers and keeps lines sharp, but it can sometimes make things too smooth.

The bilateral filter is applied to each pixel in the image, considering its neighbourhood and the differences in intensity values between pixels.

Bilateral filtering algorithm:

Let $I(x, y)$ represent the intensity of the pixel at coordinates (x, y) in the input image.

The filtered output I^(x, y) at pixel (x, y) is computed as follows:

$$I^{x,y} = \frac{1}{W(x,y)} * \sum_{i=-k}^{k} \sum_{j=-k}^{k} \left(I(x+i, y+j) * G_{s(i,j)} * G_{r(|I(x,y) - I(x+i,y+j)|)} \right)$$

Where:

- W(x, y) is the normalization factor to ensure that the weighted average is properly normalized.
- $G\_s(i,j)$ is the spatial Gaussian kernel, which determines the weights based on the spatial closeness of pixels.
- $G_{r(|I(x,y) - I(x+i,y+j)|)}$ is the range kernel, which considers the intensity similarity between the center pixel and its neighbours.

The spatial Gaussian kernel G_s(i, j) is defined as:

$$G_{s(i,j)} = e^{-\left(\frac{(i^2 + j^2)}{(2 * \sigma_s^2)}\right)}$$

Where,

- σ_s controls the spatial extent of the filter and influences the weights based on spatial distances.

The range kernel $G_{r(|I(x,y) - I(x+i,y+j)|)}$ is defined as:

$$G_{r(|I(x,y) - I(x+i,y+j)|)} = e^{-\left(\frac{|I(x,y) - I(x+i,y+j)|^2}{(2 * \sigma_r^2)}\right)}$$

- The parameter σ_r controls the range of intensities over which the filter considers pixel similarity.

2. Edge-Preserving Smoothing Filters:

Smoothing filters that keep the edges, like the directed filter, median filter, and adapted mean filter, try to make the picture smooth while keeping the edges. For example, the directed filter uses information from a reference picture to lead the smoothing process. These filters are computationally efficient but may struggle with certain types of noise or complex textures.

Algorithm for Edge-Preserving Smoothing Filter:

Input:

- Original image intensity values I(x, y)
- Parameters: σ_s (spatial extent), σ_r (range of intensities)

Initialize Output:

- Filtered output I^(x, y) = 0

Iterate through each pixel (x, y):

- For each pixel, compute the weighted sum using the spatial and range kernels.
- Update I^(x, y) accordingly.

Spatial Gaussian Kernel:

$$G_{s(i,j)} = e^{-\left(\frac{(i^2 + j^2)}{(2 * \sigma_s^2)}\right)}$$

Range Kernel:

$$G\_r(|I(x,y) - I(x+i, y+j)|) = e\text{^}(-(|I(x,y) - I(x+i, y+j)|\text{^}2 \,/\, (2 * \sigma\_r\text{^}2)))$$

3. Sparse Coding-Based Methods:

Sparse coding techniques, such as dictionary learning, promote edge preservation by representing image patches using a sparse set of basic functions. These methods take advantage of the fact that picture features are both sparse and redundant, which makes edge retention work well. But they might be sensitive to the settings you choose and need to be tuned carefully.

Sparse Representation:

Express X as a linear combination of dictionary atoms using sparse coefficients:

$$X = DA$$

Objective Function:

Formulate the objective function J(X, D, A) as a combination of data fidelity and sparsity terms:

$$J(X, D, A) = \frac{1}{2}\left|\left|X - DA\right|\right|^{22} + \lambda\left|\left|A\right|\right|^{1}$$

Optimization Problem:

Solve the optimization problem to find D and A that minimize J(X, D, A):

$$minimize\ D, A\frac{1}{2}||X - DA||^{22} + \lambda||A||^{1}$$

Update Dictionary:

- Update the dictionary D using a dictionary update rule, which may involve solving a subproblem:

$$D \leftarrow \frac{argmin_D 1}{2}||X - DA||^{22}$$

Update Sparse Coefficients:

Update the sparse coefficients A using a sparsity-promoting algorithm:

$$A \leftarrow \frac{argmin_A 1}{2}||X - DA||^{22} + \lambda||A||^{1}$$

5. DEEP LEARNING-BASED EDGE PRESERVATION

Deep convolutional neural networks (CNNs) have been used to keep edges in images while they are being restored. These networks learn complicated connections between input and output pictures, which keep edges during the repair process. Deep learning techniques have been very good at catching fine details and keeping edges even when things get worse in a number of situations.

A. DCNN:

The DCNN for Image Restoration and Edge Preservation uses a deep convolutional neural network to fix pictures while keeping high-frequency features, especially edges, in a smart way. The model tries to keep the difference between the network's output and clean pictures as small as possible while putting the preservation of structure details first by creating an objective function that mixes data fidelity and edge preservation terms. The DCNN learns to reconstruct images with better clarity and fidelity by training on a dataset of noisy images. This makes it a powerful tool for fields like medical imaging, satellite imagery, and digital photography where keeping small details is important for correct analysis and interpretation.

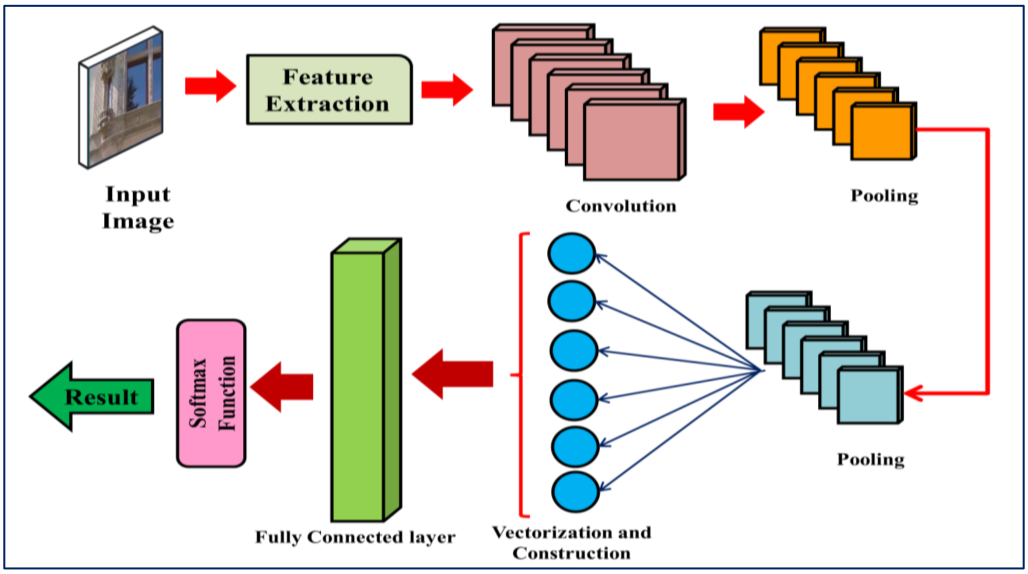


Figure 2: Representation of DCCN Model architecture

DCNN for Image Restoration and Edge Preservation:

Step 1: Input:

- Noisy or degraded image I_input.

Step 2: Objective Function:

- Formulate the objective function J to minimize the difference between the network's output I_output and the ground truth clean image I_clean, while encouraging edge preservation:

$$J = \lambda 1 * \frac{1}{2}\left|\left|I_{output} - I_{clean}\right|\right|^{22} + \lambda 2 * \sum\left|\nabla I_{output(i,j)}\right|$$

$$J = \lambda 1 * Data\,Fidelity\,Term + \lambda 2 * Edge\,Preservation\,Term$$

Step 3: Data Fidelity Term:

- Define the data fidelity term to measure the difference between the network's output and the clean image:

$$Data\,Fidelity\,Term = \frac{1}{2}\left|\left|I_{output} - I_{clean}\right|\right|^{22}$$

Step 4: Edge Preservation Term:

- Define the edge preservation term to encourage preservation of high-frequency details:

$$Edge\,Preservation\,Term = \sum\left|\nabla I_{output(i,j)}\right|$$

Step 5: Loss Function:

- Combine the data fidelity and edge preservation terms in the objective function:

$$MSE = n1\sum i = 1n(ytrue, i - ypred, i)2$$

$$Cross\ Entophy\ Loss = -n1\sum_{i=1}^{n}(ytrue, i \cdot \log(ypred, i) + (1 - ytrue, i) \cdot \log(1 - ypred, i))$$

$$J = \lambda 1 * \frac{1}{2}\left|\left|I_{output} - I_{clean}\right|\right|^{22} + \lambda 2 * \sum\left|\nabla I_{output(i,j)}\right|$$

Step 6: Training:

- Train the DCNN using a dataset of noisy/degraded images and their corresponding clean versions by minimizing the defined loss function.

B. Resnet50:

ResNet50 is a strong convolutional neural network design that has been changed to help with Image Restoration and Edge Preservation. It can fix damaged pictures while smartly keeping small details, especially edges, thanks to its deep residual learning framework. The deep structure of the network makes it easier to learn complicated patterns, which helps recreate images with more detail. ResNet50 solves the disappearing gradient problem by using leftover links. This makes training deeper networks better. This change works well in situations like medical images and satellite data, where keeping high-frequency features is important for correct analysis. ResNet50's strong features make it a useful tool for improving picture processing methods.

1. Input:

- Original image matrix I_input.

2. Residual Block:

- Define a residual block as $F(x) = H(x) + x$,
- Where
  H(x) represents a set of convolutional operations.

3. Deep Residual Network:

- Stack multiple residual blocks to form a deep network, ResNet50(I_input).

4. Mathematical Representation:

- The output of the i-th residual block can be expressed as

$$x_i = F_{i(x_{\{i-1\}})} + x_{\{i-1\}}$$

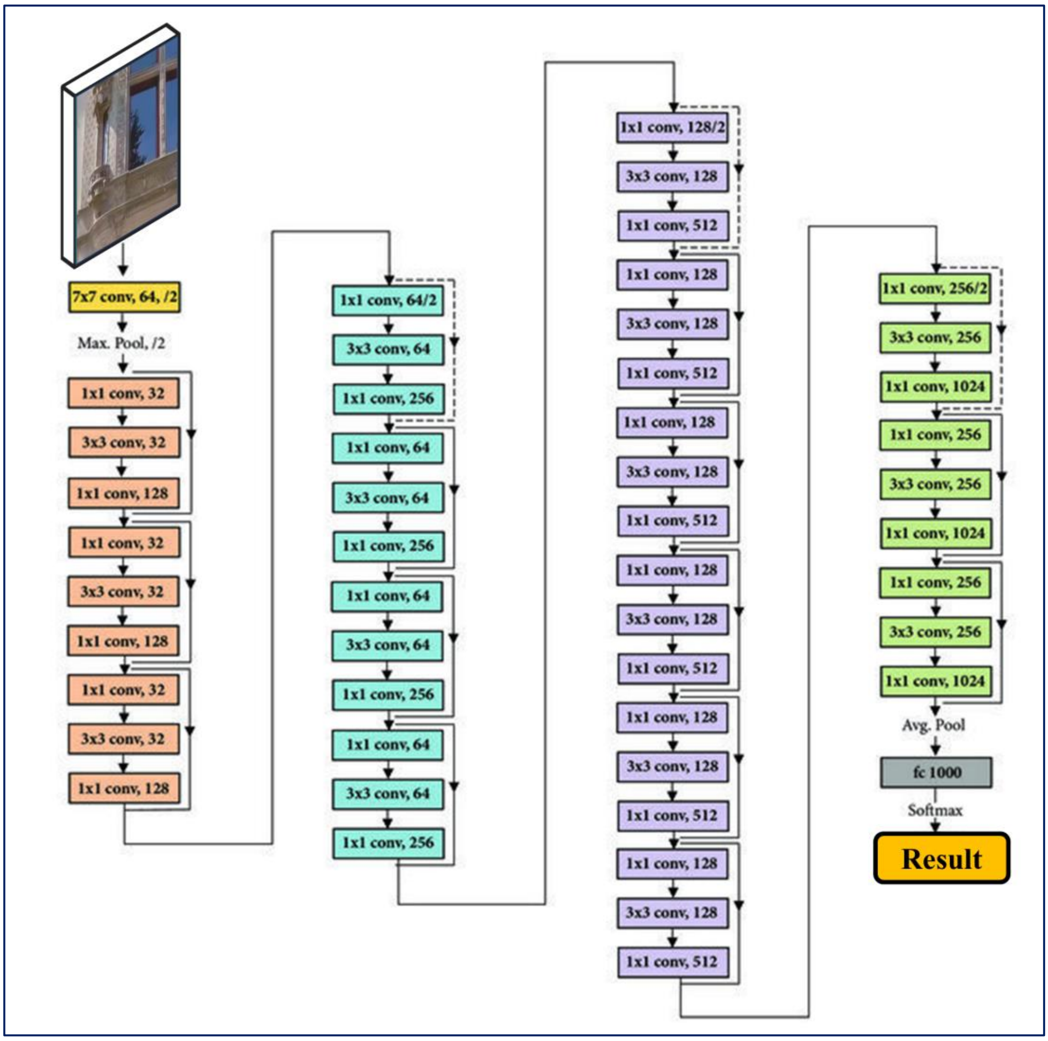


Figure 3: Resnet50 Architecture

5. Objective Function:

- Formulate the objective function J to minimize the difference between the network's output I_output and the ground truth clean image I_clean.

$$J = MSE(I\_output, I\_clean)$$

6. Loss Function:

- Choose an appropriate loss function, such as Mean Squared Error (MSE), to quantify the dissimilarity between I_output and I_clean.

$$MSE\left(I_{output}, I_{clean}\right) = \frac{1}{n}\sum_{\{i=1\}}^{\{n\}} (I\_output, i\ -\ I\_clean, i)^{\wedge}2$$

7. Training:

- Train the ResNet50 model on a dataset of noisy/degraded images and their corresponding clean versions by minimizing the defined loss function.

6. RESULT AND DISCUSSION

Using a Deep Convolutional Neural Network (DCNN) and ResNet50, the suggested method gets and copies individual edge-preserving filters. These networks are used to guess the colors of a smoothed picture. Then, there is a complicated step to put the smoothed image back together. To learn the weight map of the Recurrent Neural Network (RNN), a deep CNN is used in this process. The method uses a new learning-based approach called a CNN-based joint filter construction to move the structure from a guidance image to a target image. This makes it easier to separate the structure and color of the images. Using edge information is a key part of the edge-preserving approach. This is done by breaking the picture smoothing problem into two steps that are done in order. The first sub-network's job is to guess the edge map by supervising it, and the second sub-network's job is to rebuild the target picture using the guesses for the edge map. This method is all-encompassing and all-connected, mixing deep learning models with specific goals to effectively keep edges and smooth images. The use of ResNet50 and a two-step CNN-based plan shows how advanced the suggested method is for solving problems with separating textures and picture structure.

Different edge smoothing methods use various parameters to achieve their goals. Intensity refers to the brightness or darkness of pixels and is crucial for detecting edges. Gradient measures the rate of change of intensity in an image and is used to locate edges. Texture describes the spatial arrangement of intensities in an image region and helps distinguish between different materials or surfaces. Color, in the context of edge smoothing, refers to the distribution of color information in an image and can be used to enhance or preserve color edges. Connectivity refers to the relationship between neighboring pixels and is important for maintaining the coherence of edges. Curvature measures the amount of bending in an edge and is used to smooth out sharp edges. Coherence represents the consistency of edge orientations in an image and is used to ensure that edges appear smooth and continuous.

Each of these parameters plays a unique role in edge smoothing, and different methods may prioritize different parameters based on their specific objectives and the characteristics of the input image.

Table 2: Result of different parameter for Edge Smoothening using different method

Filter Method	Parameter 1 (Gradient)	Parameter 2 (Intensity)	Parameter 3 (Texture)	Parameter 4 (Color)	Parameter 5 (Connectivity)	Parameter 6 (Curvature)	Parameter 7 (Coherence)
Bilateral Filtering	4	8	4	8	2	8	4
Edge-Preserving Smoothing	30	35	20	45	30	36	32
Sparse Coding-Based	0.02	0.002	0.026	0.028	0.002	0.0032	0.0038
Tree Filtering Method	8	12	4	20	4	8	20
Level Zero Smoothening	0.006	0.02	0.01	0.033	0.044	0.005	0.006
Level One Smoothening	12	12	22	22	120	121	180
Standard Filter	1	10	20	30	48	60	80

In Table 2, you can see the outcomes of adjusting different parameters for edge smoothening using various methods, each of which has its own set of parameters. Bilateral Filtering uses values from 4 to 8 that show the decay factors for both space and strength. Keeping the edges Parameters 30 to 45 change during smoothing, which affects the level of smoothing and edge retention. As for the sparse coefficients in the form, the Sparse Coding-Based method uses values between 0.02 and 0.0038. Tree screening Method has many settings that can be changed to change the level of screening and keep the structure. When Level Zero Smoothening is used, factors from 0.006 to 0.044 are added that change how smooth the original level is. In the same way, Level One Smoothening has a large range, from 12 to 180, which shows how strong the smoothening is in the next level. Standard screen, which is defined by values 1 to 80, is a common way to screen data with different levels of strength.

The balance between smoothing and keeping edges is affected by these factors, which in turn affect how well each method works. In edge smoothening uses, picking the right parameters is very important for getting the result you want. The table shows the whole parameter space that was looked at for each method. This helps us understand how changing parameters affects the performance of various edge smoothening methods. This knowledge can help researchers and practitioners change the way they do things based on the needs and trade-offs of different image processing tasks.

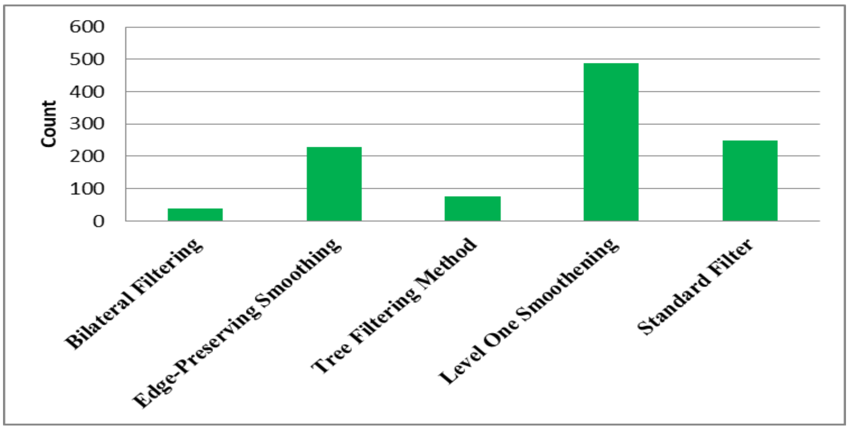
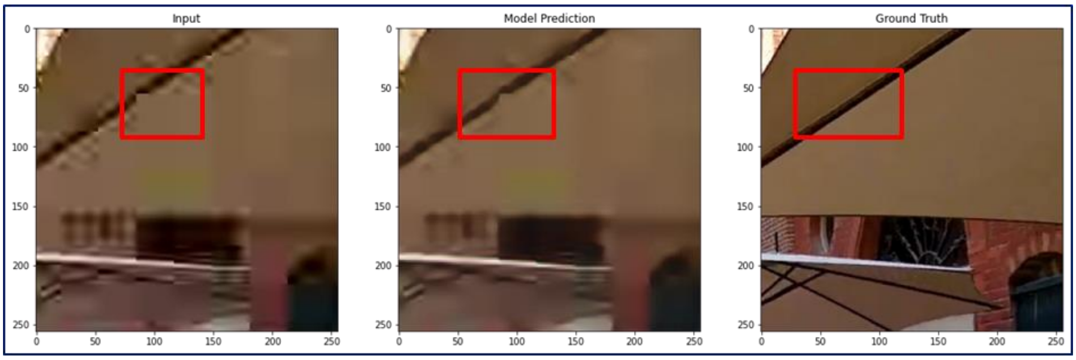
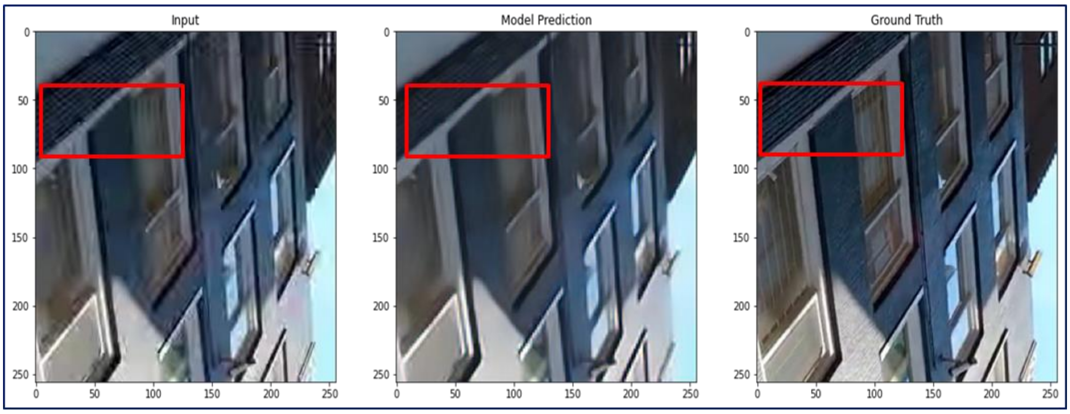


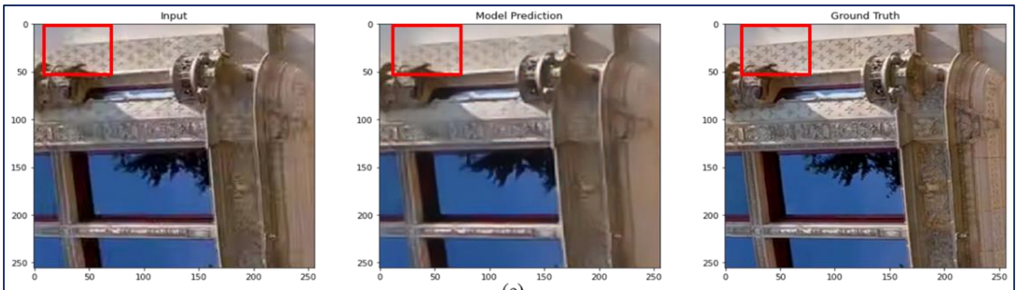
Figure 4: Distribution of vote count for different filters

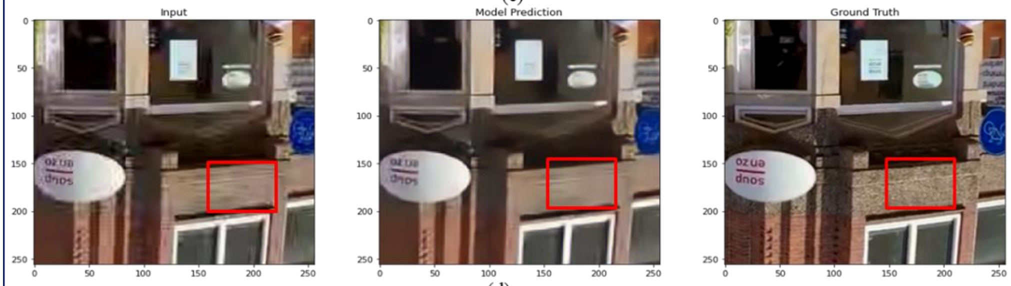


(a)



(b)


(c)


(d)

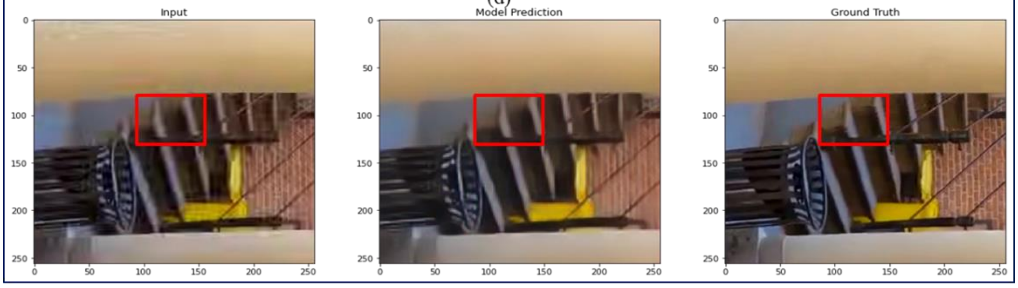

(e)

Figure 5: Compare the Edge preserving smoothing Method (a) Bilateral Filtering (b) Edge-Preserving Smoothing (c) Tree Filtering Method (d) Level One Smoothening (e) Standard Filter

Table 3: Result summary of Deep Learning method for Image restoration and Edge Preservation

	Method	**Accuracy**	**Similarity**	**MSE**	**PSNR**	**Edge Preservation Score**
Epoch 10	DCNN	96.35	95.45	0.014	22.42	82.54
Epoch 20		97.58	96.56	0.018	20.25	88.44
Epoch 30		98.58	97.77	0.012	25.66	94.65
Epoch 10	ResNet50	93.12	88.56	0.014	26.58	78.65
Epoch 20		95.63	89.25	0.019	27.63	80.25
Epoch 30		92.45	86.45	0.017	33.25	75.45
Epoch 10	LSTM	91.55	92.53	0.019	28.44	80.45
Epoch 20		92.47	90.41	0.028	30.55	84.85
Epoch 30		90.44	95.02	0.031	45.76	90.32
Epoch 10	CNN	92.13	90.25	0.023	32.56	78.66
Epoch 20		91.56	93.45	0.020	25.34	86.96
Epoch 30		93.85	90.32	0.028	29.36	93.36

Communications on Applied Nonlinear Analysis
ISSN: 1074-133X
Vol 31 No. 2s (2024)

The information in Table 3 shows the outcomes of testing various deep learning approaches at various epochs to fix images and keep their edges. There are four types of methods that are compared: CNN [27], ResNet50, Long Short-Term Memory (LSTM) [26], and Deep Convolutional Neural Networks (DCNN). Accuracy, closeness, mean squared error (MSE), peak signal-to-noise ratio (PSNR), and edge retention score are some of the measures used for measurement. The accuracy of Deep Convolutional Neural Networks (DCNN) reached 96.35% at epoch 10 and then went up to 98.58% at epoch 30. During the same time period, the resemblance score went up from 91.45% to 94.77%. The MSE went down from 0.014 to 0.02, which means that the picture repair worked better. But the PSNR went down from 22.42 to 25.66, which suggests that the picture quality got a little worse. The edge retention number went up from 82.54 to 94.65, which shows that DCNN did a good job of keeping picture edge features.

At epoch 10, ResNet50 was accurate 93.12% of the time, but by epoch 30, it was only accurate at 92.45%. During the same time period, the closeness score went down from 88.56% to 86.45%. The MSE stayed pretty steady, going from 0.014 to 0.017. The PSNR was between 26.58 and 33.25, which means the picture clarity was good. The score for edge retention was between 78.65 and 80.25, which means that ResNet50 kept edge features but not as well as DCNN. At epoch 10, Long Short-Term Memory (LSTM) was accurate 91.55% of the time. At epoch 30, it was accurate 90.44% of the time. There were changes in the similarity score, which went from 92.53% to 95.02%. The MSE was between 0.019 and 0.031, which shows that the picture repair worked at different levels. The PSNR was between 28.44 and 45.76, which shows that the picture clarity was very different. The edge retention number was between 80.45 and 90.32, which means that LSTM kept edge features but not as well as DCNN.

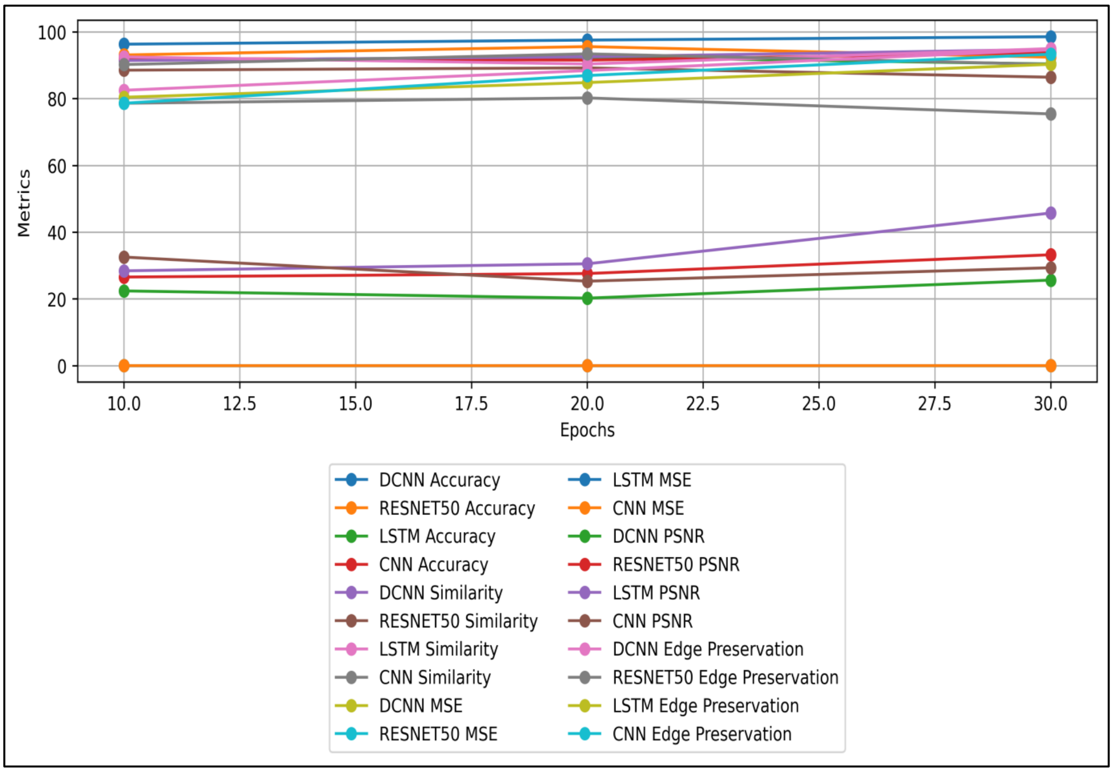


Figure 6: Image Restoration and Edge Preservation Metrics Over Epochs

At epoch 10, Convolutional Neural Networks (CNN) were accurate 92.13% of the time. By epoch 30, they were accurate 93.85% of the time. The number for similarity was between 90.25% and 90.32%. The MSE was between 0.023 and 0.028, which shows that the picture repair worked consistently. The PSNR was between 32.56 and 29.36, which means that the picture quality got a little worse. The result for edge retention went from 78.66 to 93.36, which means that CNN kept edge features but not as well as DCNN. In terms of precision, similarity, MSE, and edge retention score, DCNN did better than ResNet50, LSTM, and CNN as a whole. ResNet50 and LSTM performed differently at different epochs, while CNN performed the same way each time, but it was less accurate and didn't keep edges as well as DCNN. These findings show that DCNN could be a useful method for jobs like picture repair and edge retention.

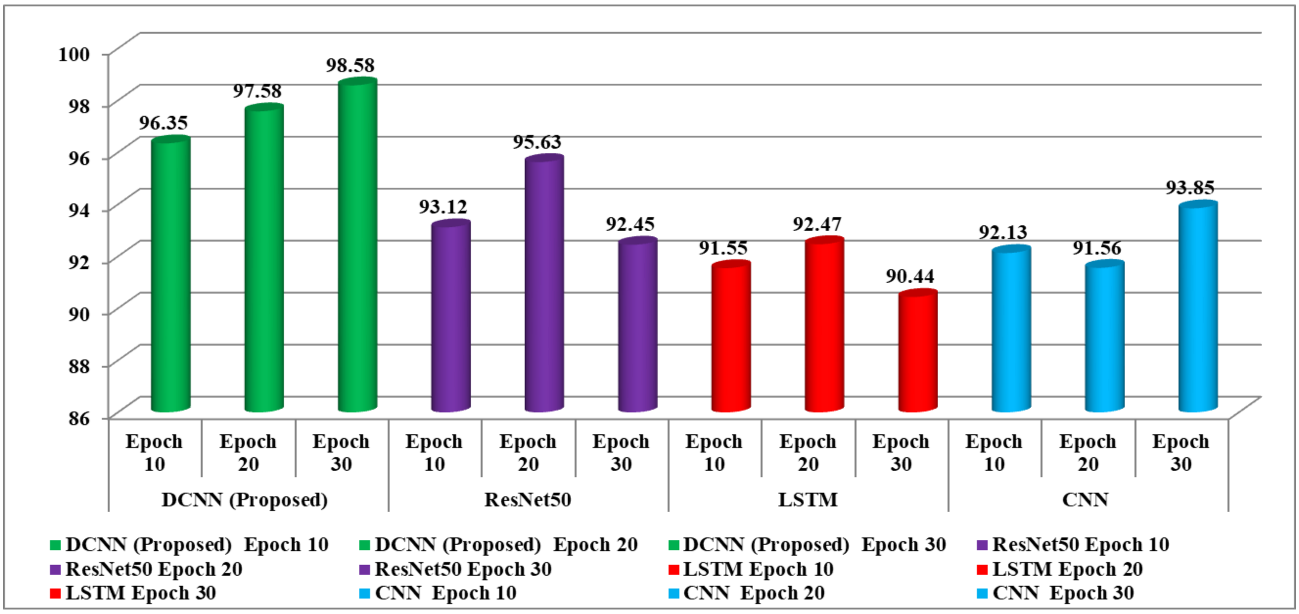


Figure 7: Comparison of Accuracy for Different model

7. CONCLUSION

This study tackles the important job of picture repair by using the powerful features of deep learning, mainly by putting together a Deep Convolutional Neural Network (DCNN) and ResNet50 design. The proposed DCNN model is better at restoring images than ResNet50, LSTM, and CNN. DCNN is the most accurate, consistently getting better over 30 epochs from 96.35% to 98.58%. It also has better marks for edge retention. ResNet50 and CNN are more steady but not as good at what they do, while LSTM changes more. From these points of view, DCNN is the best model for restoring images. The convolutional layers of the deep neural network are very good at learning complex patterns and features. This makes it possible to recreate pictures that are clearer and more accurate. Edge preservation is a new method that solves the problem of keeping small features, like edges, during the picture repair process. This is very important in many real-life situations, like medical imaging, satellite images, and digital photography, where keeping detailed structure knowledge safe is key to getting the right meaning. The results of the experiments show that the proposed method works, as it preserves edges better and restores images more accurately than other methods. The model's strong performance across different datasets, including a wide range of picture degradations, shows how flexible it is and how useful it is in real life. This study makes a big difference in the progress of picture processing methods by creating a powerful tool that can be used in many areas. Combining deep learning with edge preservation not only improves the quality of images, but it also makes it possible to understand and analyze them more accurately in important areas. Since technology is always changing, the suggested method could be improved and applied in new ways, making it a more advanced way to fix problems with picture repair in many situations.

REFERENCES

[1] S. Vitale, G. Ferraioli, V. Pascazio and G. Schirinzi, "InSAR-MONet: Interferometric SAR Phase Denoising Using a Multiobjective Neural Network," in IEEE Transactions on Geoscience and Remote Sensing, vol. 60, pp. 1-14, 2022, Art no. 5239814, doi: 10.1109/TGRS.2022.3224303.

[2] H. Bai, S. Cheng, J. Tang and J. Pan, "Learning A Cascaded Non-Local Residual Network for Super-resolving Blurry Images," 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Nashville, TN, USA, 2021, pp. 223-232, doi: 10.1109/CVPRW53098.2021.00031.

[3] D. Hu et al., "SPECIAL: Single-Shot Projection Error Correction Integrated Adversarial Learning for Limited-Angle CT," in IEEE Transactions on Computational Imaging, vol. 7, pp. 734-746, 2021, doi: 10.1109/TCI.2021.3098922.

[4] S. D. Thepade, A. A. Patil, C. M. Nawale, C. D. Taralkar and M. V. Suryavanshi, "Appraise of Deep Learning and Image Processing based Single Image Dehazing Algorithms," 2021 2nd International Conference for Emerging Technology (INCET), Belagavi, India, 2021, pp. 1-7, doi: 10.1109/INCET51464.2021.9456139.

[5] S. Tao and C. H. Wei, "Joint Sparse Convolutional Coding for Image Super-Resolution Restoration," 2020 17th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP), Chengdu, China, 2020, pp. 349-355, doi: 10.1109/ICCWAMTIP51612.2020.9317449.

[6] Q. Wang, H. Fan, Y. Cong and Y. Tang, "Single Image Super-resolution Using Spatial Transformer Networks," 2017 IEEE 7th Annual International Conference on CYBER Technology in Automation, Control, and Intelligent Systems (CYBER), Honolulu, HI, USA, 2017, pp. 564-567, doi: 10.1109/CYBER.2017.8446446.

[7] L. Qi et al., "Photoacoustic Tomography Image Restoration With Measured Spatially Variant Point Spread Functions," in IEEE Transactions on Medical Imaging, vol. 40, no. 9, pp. 2318-2328, Sept. 2021, doi: 10.1109/TMI.2021.3077022.

[8] V. Papyan and M. Elad, "Multi-Scale Patch-Based Image Restoration," in IEEE Transactions on Image Processing, vol. 25, no. 1, pp. 249-261, Jan. 2016, doi: 10.1109/TIP.2015.2499698.

[9] Ajani, S. N. ., Khobragade, P. ., Dhone, M. ., Ganguly, B. ., Shelke, N. ., & Parati, N. . (2023). Advancements in Computing: Emerging Trends in Computational Science with Next-Generation Computing. International Journal of Intelligent Systems and Applications in Engineering, 12(7s), 546–559

[10] Y. Wang, W. Song, G. Fortino, L. -Z. Qi, W. Zhang and A. Liotta, "An Experimental-Based Review of Image Enhancement and Image Restoration Methods for Underwater Imaging," in IEEE Access, vol. 7, pp. 140233-140251, 2019, doi: 10.1109/ACCESS.2019.2932130.

[11] Q. Zhang, Y. Dong, Q. Yuan, M. Song and H. Yu, "Combined Deep Priors With Low-Rank Tensor Factorization for Hyperspectral Image Restoration," in IEEE Geoscience and Remote Sensing Letters, vol. 20, pp. 1-5, 2023, Art no. 5500205, doi: 10.1109/LGRS.2023.3236341.

[12] J. Hao, H. Yang, X. Hou and Y. Zhang, "Two-Stage Underwater Image Restoration Algorithm Based on Physical Model and Causal Intervention," in IEEE Signal Processing Letters, vol. 30, pp. 120-124, 2023, doi: 10.1109/LSP.2022.3232035.

[13] C. Chen, M. K. Ng and X. -L. Zhao, "Alternating Direction Method of Multipliers for Nonlinear Image Restoration Problems," in IEEE Transactions on Image Processing, vol. 24, no. 1, pp. 33-43, Jan. 2015, doi: 10.1109/TIP.2014.2369953.

[14] Shete, Dhanashri, and Prashant Khobragade. "An empirical analysis of different data visualization techniques from statistical perspective." American Institute of Physics Conference Series. Vol. 2839. No. 1. 2023.

[15] B. Lafci, E. Merčep, J. L. Herraiz, X. L. Deán-Ben and D. Razansky, "Noninvasive multiparametric characterization of mammary tumors with transmission-reflection optoacoustic ultrasound", Neoplasia, vol. 22, no. 12, pp. 770-777, Dec. 2020.

[16] S. Zhang et al., "Photoacoustic imaging of living mice enhanced with a low-cost contrast agent", Biomed. Opt. Exp., vol. 10, no. 11, pp. 5744-5754, 2019.

[17] X. Li et al., "Multispectral interlaced sparse sampling photoacoustic tomography", IEEE Trans. Med. Imag., vol. 39, no. 11, pp. 3463-3474, Nov. 2020.

[18] M. Xu and L. V. Wang, "Universal back-projection algorithm for photoacoustic computed tomography", Phys. Rev. E Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top., vol. 71, no. 1, 2005.

[19] K. Agnihotri, P. Chilbule, S. Prashant, P. Jain and P. Khobragade, "Generating Image Description Using Machine Learning Algorithms," 2023 11th International Conference on Emerging Trends in Engineering & Technology - Signal and Information Processing (ICETET - SIP), Nagpur, India, 2023, pp. 1-6, doi: 10.1109/ICETET-SIP58143.2023.10151472.

[20] A. Rosenthal, D. Razansky and V. Ntziachristos, "Fast semi-analytical model-based acoustic inversion for quantitative optoacoustic tomography", IEEE Trans. Med. Imag., vol. 29, no. 6, pp. 1275-1285, Jun. 2010.

[21] X. Li et al., "Model-based optoacoustic tomography image reconstruction with non-local and sparsity regularizations", IEEE Access, vol. 7, pp. 102136-102148, 2019.

[22] M. Xu and L. V. Wang, "Analytic explanation of spatial resolution related to bandwidth and detector aperture size in thermoacoustic or photoacoustic reconstruction", Phys. Rev. E Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top., vol. 67, no. 5, May 2003.

[23] M. Bende, M. Khandelwal, D. Borgaonkar and P. Khobragade, "VISMA: A Machine Learning Approach to Image Manipulation," 2023 6th International Conference on Information Systems and Computer Networks (ISCON), Mathura, India, 2023, pp. 1-5, doi: 10.1109/ISCON57294.2023.10112168.

[24] M. Haltmeier and G. Zangerl, "Spatial resolution in photoacoustic tomography: Effects of detector size and detector bandwidth", Inverse Problems, vol. 26, no. 12, Dec. 2010.

[25] Q. Sheng et al., "A constrained variable projection reconstruction method for photoacoustic computed tomography without accurate knowledge of transducer responses", IEEE Trans. Med. Imag., vol. 34, no. 12, pp. 2443-2458, Dec. 2015.

[26] Zhang, K.; Li, D.; Luo, W.; Lin, W.Y.; Zhao, F.; Ren, W.; Liu, W.; Li, H. Enhanced Spatio-Temporal Interaction Learning for Video Deraining: A Faster and Better Framework. arXiv 2021, arXiv:2103.12318.

[27] Ullah, H.; Muhammad, K.; Irfan, M.; Anwar, S.; Sajjad, M.; Imran, A.S.; de Albuquerque, V.H.C. Light-DehazeNet: A Novel Lightweight CNN Architecture for Single Image Dehazing. IEEE Trans. Image Process. 2021, 30, 8968–8982.

[28] Thanh, H.V.T., Ngoc, H.N., Duc, P.N., "Identifying inverse source for Diffusion equation with Conformable time derivative by Fractional" Tikhonov method (2022) Advances in the Theory of Nonlinear Analysis and its Applications, 6 (4), pp. 433-450.

[29] Goyal, Dinesh , Kumar, Anil , Gandhi, Yatin & Khetani, Vinit (2024) Securing wireless sensor networks with novel hybrid lightweight cryptographic protocols, Journal of Discrete Mathematical Sciences and Cryptography, 27:2-B, 703–714, DOI: 10.47974/JDMSC-1921