IRAQI STATISTICIANS JOURNAL

https://isj.edu.iq/index.php/isj

ISSN: 3007-1658 (Online)

Enhancing Satellite Imagery: A Novel Approach to Gaussian Noise Reduction Using Convolutional Neural Networks and Nonlinear Filtering Techniques

Mohammed Abdul Wadood Mohammed[1], Assma Ghalib Jaber[2]

[1] mohamed.a.mohamed@coadec.uobaghdad.edu.iq, [2] Drasmaa.ghalib@coadec.uobaghdad.edu.iq

[1, 2] Department of Statistics, College of Administration and Economics, University of Baghdad, Baghdad, Iraq

ARTICLE INFO

Article history:

Received 26/11/2024
Revised 26/11/2024
Accepted 14/1/2025
Available online 15/5/2025

Keywords:

Image Denoising
Image Restoration
Gaussian Noise
Convolution Neural Network

ABSTRACT

Image denoising is one of the fundamental aspects of removing noise from an image and enhancing its features containing visual information. Based on this, Convolutional Neural Networks (CNNs) have a latest topic of study, with a wide range of applications in fields as diverse as diagnostic image denoising and low-light image denoising. In this paper, an image denoising method is proposed based on converting the noisy image to YUV colon space, extracting the noisy Y channel, and obtaining an appropriate smoothing parameter for the noisy Y channel using the cross-validation smoothing techniques that are used to estimate the Y density function, then using an appropriate noise reduction method (total variation denoising) on the estimated density function to extract the denoised Y channel by removing the Gaussian noise. The results showed that the new proposed method effectively removed noise from the image, which is attributed to the approach adopted in this proposed filter.

1. Introduction

Space telescopes have given us unprecedented views of distant stars, galaxies, and cosmic phenomena, expanding our understanding of the universe. However, one of the significant challenges faced in capturing these images is the presence of noise, particularly Gaussian noise, which distorts the clarity and detail of the images. This type of noise, characterized by random variations in pixel intensity, can obscure critical features, making it difficult for astronomers to extract accurate information from the images. Addressing this problem is crucial, as the quality of space telescope images directly impacts the precision of scientific discoveries and observations.

To combat Gaussian noise, modern advancements in deep learning, particularly Convolutional Neural Networks (CNNs), have opened up new avenues for enhancing image-denoising performance.

In this article, we explore the integration of a Convolutional Neural Network (CNN) with a proposed filter specifically designed to reduce Gaussian noise in space telescope images. CNNs have gained widespread use in image processing due to their ability to learn complex patterns and features in data, making them particularly effective for tasks such as

* Corresponding author: E-**mail** : mohamed.a.mohamed@coadec.uobaghdad.edu.iq
https://doi.org/10.62933/37j8tb47

denoising. By training a CNN to identify and suppress noise while preserving essential image details, we aim to significantly improve the clarity of space telescope images compared to traditional filtering techniques alone.

The proposed approach combines the learning capabilities of CNNs with a refined filter that adapts to different noise levels and image features. The CNN is trained to distinguish between noise and actual image content, applying the filter intelligently across the image to achieve better noise reduction without compromising detail. This hybrid method seeks to outperform classical techniques like spline and exponential spline filters by leveraging deep learning's ability to learn context-aware noise patterns while maintaining computational efficiency.

Through a comprehensive comparative analysis, we will assess the performance of this CNN-based method alongside the spline filter and exponential spline filter. The evaluation will focus not only on the visual quality of the denoised images but also on the accuracy of extracted features critical to scientific research. The ultimate goal of this study is to enhance the quality of space telescope imagery, allowing scientists to analyze celestial objects with greater precision and clarity. By improving image processing methods through the integration of CNNs and innovative filtering techniques, we aim to push the boundaries of space exploration and contribute to the ongoing pursuit of new astronomical discoveries.

In recent years, some researchers have addressed the use of these filters to reduce noise, including the study by Tian et al. (2018) [23] introduced a novel method called enhanced convolutional neural denoising network (ECNDNet), which uses residual learning and batch normalization techniques to overcome training difficulties and accelerate convergence. The network also uses dilated convolutions to increase context information and reduce computational costs.

Bora and Chaudhary (2021) [2] proposed a method to eliminate Gaussian noise from grey images by enhancing bilateral filters and The ECNDNet method achieved improved PSNR values up to 34.73 when combined with bilateral filters and deep

Thayammal et al. (2021) [21] studied the performance of a CNN-based denoising method for low-light images, presenting a model called DnCNNs, which implicitly removes image noise, providing better reference for application developers.

Zhang et al. (2022) [29] proposed a robust deformed denoising CNN (RDDCNN) to address the issue of convolutional operations altering noise distributions in corrupted images. The method includes three blocks: a deformable block, an enhanced block, and a residual block. Experimental results show the model outperforms popular methods in qualitative and quantitative analysis.

Zheng et al. (2022) [30] developed a hybrid denoising CNN (HDCNN) to address the issue of poor performance on complex screens. The HDCNN consists of a dilated block, a RepVGG block, a feature refinement block, and a single convolution, resulting in good performance in image denoising. The experiment demonstrated its effectiveness in public data sets.

The study by Xie et al. (2023) [25] introduced a multi-level information fusion CNN (MLIFCNN) for image denoising, which includes a fine information extraction block, a multi-level information interaction block, a coarse information refinement block, and a reconstruction block. The method is compared to other excellent methods in terms of both quantitative and qualitative performance.

2. Convolutional Neural Network (CNN):

Neural network-based denoising strategies are drawing in acceptable considerations for their efficient performance in image rebuilding. They first train the network, and then the network accepts input as noisy patches, and the noiseless clear patches are estimated from the noise patches [9]. Each network contains a set of non-linear activations and convolution operations. It distinguishes the hidden prior of the image from the training set for image recovery [28]. Deep learning approaches have the best learning capacity and adaptable network design, which improve the efficiency of denoising. CNN is a deep learning approach, which has pulled in more consideration in denoising noisy images. Rectifier Linear Unit (ReLU), residual learning, and batch normalization (BN) are utilized in CNN to quicken the training of the network and improve the efficiency of denoising [14]. A convolutional network is an alternating sequence of linear filtering and nonlinear transformation operations. The input and output layers include one or more images, while intermediate layers contain “hidden" units with images called feature maps that are the internal computations of the algorithm [7].

For good denoising tasks using (CNN), Several factors are of central importance in this progress:

(i) the efficient training implementation.
(ii) the Rectified Linear Unit (ReLU) which makes convergence much faster while still presenting good quality.
(iii) the easy access to an abundance of data for training larger models.

Patch extraction and representation operation extracts (overlapping) patches from the low-resolution image (Y) and represents each patch as a high-dimensional vector [22]. These vectors comprise a set of feature maps, of which the number equals the dimensionality of the vectors. patch extraction and representation are a popular strategy in image restoration to densely extract patches and then represent them by a set of pre-trained bases such as PCA, DCT, Haar, etc. This is equivalent to convolving the image by a set of filters, each of which is a basis. Formally, our first layer is expressed as an operation [9]:

$$\mathrm{F}_1(\mathrm{Y}) = \max(0, \mathrm{W}_1 * \mathrm{Y} + \mathrm{B}_1) \quad (1)$$

Where $\{W_1$ and $B_1\}$ represent the filters and biases respectively, $\{*\}$ denotes the convolution operation. Here, $\{W_1\}$ corresponds to $\{n_1\}$ filters of support $\{c \times f_1 \times f_1)$, where $\{c\}$ is the number of channels in the input image, $\{f_1\}$ is the spatial size of $\{a\}$ filter.

Non-linear mapping operation nonlinearly maps each high-dimensional vector onto another high-dimensional vector. Each mapped vector is conceptually the representation of a high-resolution patch. These vectors comprise another set of feature maps [13]. The first layer extracts an $\{n_1$-dimensional feature$\}$ for each patch. In the second operation, we map each of these $\{n_1$-dimensional vectors$\}$ into an $\{n_2$-dimensional one$\}$. This is equivalent to applying $\{n_2\}$ filters which have a trivial spatial support $\{1\times1\}$. This interpretation is only valid for $\{1\times1\}$ filters. But it is easy to generalize to larger filters like $\{3\times3\}$ or $\{5\times5\}$. In that case, the non-linear mapping is not on a patch of the input image; instead, it is on a $\{3\times3\}$ or $\{5\times5\}$ “patch” of the feature map. The operation of the second layer is [5]:

$$F_2(Y) = max\,(0, W_2 * F_2(Y) + B_2) \quad (2)$$

Here $\{W_2\}$ contains $\{n_2\}$ filters of size $\{n_1 \times f_2 \times f_2\}$, and $\{B_2\}$ is $\{n_2$-dimensional$\}$. Each of the output $\{n_2$-dimensional vectors$\}$ is conceptually a representation of a high-resolution patch that will be used for reconstruction.

Reconstruction operation aggregates the above high-resolution patch-wise representations to generate the final high-resolution image. This image is expected to be similar to the ground truth (X) [4]. In traditional methods, the predicted overlapping high-resolution patches are often averaged to produce the final full image. The averaging can be considered as a pre-defined filter on a set of feature maps (where each position is the “flattened” vector form of a high-resolution patch) [14]. Motivated by this, we define a convolutional layer to produce the final high-resolution image:

$$F(Y) = W_3 * F_2(Y) + B_3 \qquad (3)$$

Here $\{W_3\}$ corresponds to c filters of a size $\{n_2 \times f_3 \times f_3\}$, and $\{B_3\}$ is a {c-dimensional vector}.

3. Proposed Filter:

The proposed filter converts the image from RGB to YUV color space as step one, where the (Y, U, V) color space is a color representation used in digital image and video processing. It separates the luminance (brightness) information from the chrominance (color) information in an image [17]. then in step two, we split the image channels to extract the noisy (Y) channel and apply a bandwidth selection method on this (Y) channel, where the bandwidth selection, or smoothing parameter (h) is a crucial step in nonparametric estimation techniques, particularly kernel density estimation and kernel regression [8]. It involves determining an appropriate bandwidth parameter (h) that controls the smoothing or blurring effect of the kernel function [18].

Nonparametric estimation aims to estimate an underlying probability density function or regression function from a given set of data points. The kernel function is a smooth, symmetric function cantered at each data point, and the bandwidth determines the width of this kernel function [19]. The choice of bandwidth significantly influences the quality and accuracy of the estimated function. If the bandwidth is too large, the estimate may become overly smooth and fail to capture important features or structures in the image. On the other hand, if the bandwidth is too small, the estimate may exhibit excessive noise and reflect the specific characteristics of individual data points rather than the underlying pattern.

Cross-validation (CV) is a method that offers a criterion for optimality that works as an empirical analog of the (MISE) and so it allows us to estimate (h). There are three types of (CV), Least Squares Cross-Validation (LSCV), also called unbiased (UCV), involves the (ISE) [6].

$$ISE_h = \int_{-\infty}^{\infty} (\hat{f}_h(x) - f(x))^2 dx \qquad (4)$$

Where $(f(x))$ and $(\hat{f}_h(x))$ is the density and density estimator, which leads to [32].

$$\hat{h}_{LSCV} = \text{argmin}\,(LSCV_h) \qquad (5)$$

Biased Cross-Validation (BCV), where it attempts to directly minimize the (AMISE). This requires an estimation of the unknown R(f"), which requires selecting another bandwidth [11].

$$AMISE = \frac{K}{nh} + \frac{K_2^2\, R(f'')}{nh} \qquad (6)$$

By replacing the unknown values in the $\{R(f'')\}$ term with the estimate $\{\tilde{R}(f'')\}$, we obtain the $\{BCV_h\}$ estimator [10]:

$$BCV_h = \frac{K}{nh} + \frac{K_2^2}{nh}\left(R\left(\hat{f}_h''\right) - \frac{R(K'')}{nh^5}\right) \qquad (7)$$

$$\therefore \hat{h}_{BCV} = \text{argmin}\, CV(h) \qquad (8)$$

Maximum Likelihood Cross Validation (MLCV). The rationale behind this method is to estimate the log-likelihood of the density at observation (x_i) based on all observations except (x_i). Averaging this log-likelihood over all observations results in the following (MLCV) score [20].

$$\hat{h}_{MLCV} = \text{argmin}\, CV(h) \qquad (9)$$

MLCV seeks to test the hypothesis:

$$\left.\begin{matrix} H_0: \hat{f}_x = f_x \\ H_0: \hat{f}_x \neq f_x \end{matrix}\right\} \qquad (10)$$

The next step, bandwidth parameters (h) for the (Y) channel that we get by (CV), use it for density estimation, where the Kernel density estimation (KDE) is a non-parametric method used to estimate the probability density function (PDF) of a random variable based on a set of observed data points [1]. KDE works by placing a kernel (a smooth, symmetric, and non-negative function) on each data point and summing up these kernels to obtain the estimated PDF [26]. The estimated density at any point x is formulated as [3].

$$\hat{f}_h(x) = \frac{1}{n}\sum_{i=1}^{n} K\left(\frac{x - X(i)}{h}\right) \qquad (11)$$

where x(i) is a neighboring point to (x), (n) is the number of neighbors, K (·) is the kernel function, and (h) is the bandwidth. The kernel function can be considered a weighting factor that gives a larger value when x(i) is close to (x) [10]. This density estimation will reconstruct the (Y) channel then in the next step, we apply the denoising method to get the denoised (Y) channel which we use to rebuild the new denoised image in the final step. There are several denoising methods or filters for the denoising task.

Total Variation (TV) is used for image denoising and restoration. TV method effectively reduces noise while preserving edges and important image structures by minimizing the total variation of an image, which is a measure of the total amount of variation or changes between neighboring pixels [27]. For the image denoising task, TV assumes that the noisy image y(n) is of the form

$$y_n = x_n + w_n \qquad n = 0, \dots, N-1 \qquad (12)$$

where x(n) is a (approximately) piecewise constant signal and w(n) is white Gaussian noise. (TV) estimates the image x(n) by solving the optimization problem [33].

$$\text{argmin}_x = \left\{F_x = \frac{1}{2}\sum_{n=0}^{N-1}|y_n - x_n|^2 + \lambda \sum_{n=1}^{N-1}\left|x_n - x_{(n-1)}\right|\right\} \qquad (13)$$

The regularization parameter λ > 0 controls the degree of smoothing. Increasing λ gives more weight to the second term which measures the fluctuation of the signal x(n) [31]. the TV denoising in equation (2) can be written compactly as:

$$\text{argmin}_x = \left\{F_x = \frac{1}{2}\|y - x\|_2^2 + \lambda\|D_x\|_1\right\}$$
(14)

The N-point signal x is represented by the vector:

$$X = [x_0, x_1, \dots, x_{N-1}]^T \qquad (15)$$

Classical ℓ_1 TV computed independently on each color component [12].

$$\|X\|_1 = \sum\|X_k\|_1 \qquad (p = 1, q = 1) \qquad (16)$$

ℓ_2 TV computes the Euclidean norm of the vector

$$\|X\|_2 = \left(\sum_k X_k^2\right)^{\frac{1}{2}}(p = 2, q = 1) \qquad (17)$$

Squared ℓ2 TV computes the squared Euclidean norm of the vector [22].

$$\|X\|_2 = \left(\sum_k X_k^2\right)(p = 2, q = 2) \qquad (18)$$

The matrix D is defined as [13]

$$D = \begin{bmatrix} -1 & 1 & \cdots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & & \cdots & -1 \quad 1 \end{bmatrix} \qquad (19)$$

The first-order difference of an N-point image x is given by D_x where D is of size $(N-1) \times N$. Note, for later, that DD^T is a tridiagonal matrix of the form:

$$D = \begin{bmatrix} 2 & -1 & & \cdots & & 0 & \\ -1 & 2 & -1 & & & & \\ & \vdots & & \ddots & & \vdots & \\ & 0 & & \cdots & -1 & 2 & -1 \\ & & & & & -1 & 2 \end{bmatrix}$$
(20)

The total variation of the N-point image x(n) is given by

$$TV(x) = \|D_x\|_1 = \sum_{n=1}^{N-1}\left|x_n - x_{(n-1)}\right|$$
(21)

The main advantage of the TV formulation is the ability to preserve edges in the image due to the piecewise smooth regularization property of the TV norm [16].

Finally, after we get the new denoised Y channel, we can reconstruct the denoised image by merging the new Y channel with the U and V channels replace the denoised Y channel with the noise Y channel in the image.

4. Results and discussion

To compare the results of the filters used, we relied on two quality measurement criteria, The Peak Signal Noise Ratio (PSNR) which is the ratio of the maximum image values to the magnitude of noise affecting the image

$$PSNR_{(X,Y)} = 10 * \log_{10}\frac{\left(MAX_{pixels}^2\right)}{(MSE)} \qquad (22)$$

Where the original image (X) and the resulting image (Y) are compared using the Max brightness value (255) and the mean square error (MSE) between the two images.

And the SSIM index measures structural similarity between two images, with perfect quality indicating the quality of the other image being compared.

$$\mathrm{SSIM}_{(X,Y)} = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)} \qquad (23)$$

The experiment was carried out by adding AWGN with zero mean and 0.01 variance to the approved image as shown in Figure 1, which is a dumbbell nebula, Considering the significance of these images, we should work to eliminate any noise that may have been introduced during the transmission and acquisition process. So, in this experiment, we added different percentages of Gaussian noise to the adopted image and then applied the adopted filters, the code of these filters is written using MATLAB.

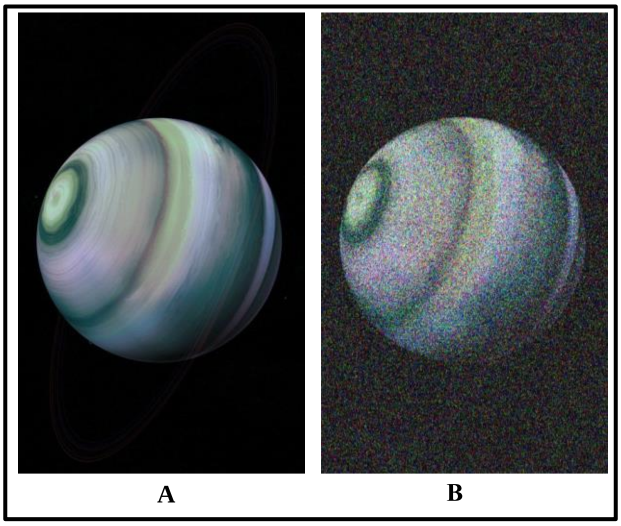

Figure 1: (**A**) Uranus Clean Image, (**B**) Uranus Noisy Image.

Table 1: PSNR and SSIM Values for The Restored Images for Each Filter.

Filters	**Image Quality Measurements**	
	PSNR	**SSIM**
Proposed	42.20	0.98
CNN	38.64	0.93

The results indicate that the proposed filter performs best in terms of both PSNR and SSIM when there is a noise density of 0.01 where it is given the value 42.20 PSNR, and 0.98 SSIM respectively, while the CNN filter ranks second with 38.64 PSNR and 0.93 SSIM. Figure 2 displays the images that have been restored.

When we implement the filters in different noise Ratio (**50%**, **75%**), to denoise the (Uranus) image, we get the following results:

Table 2: PSNR and SSIM Values of The Restored Images Calculated for Different Noise Ratio.

Filters	**Image Quality Measurements in Different Noise Ratios**			
	50%		**75%**	
	PSNR	**SSIM**	**PSNR**	**SSIM**
Proposed	40.30	0.97	38.69	0.96
CNN	36.98	0.92	34.52	0.90

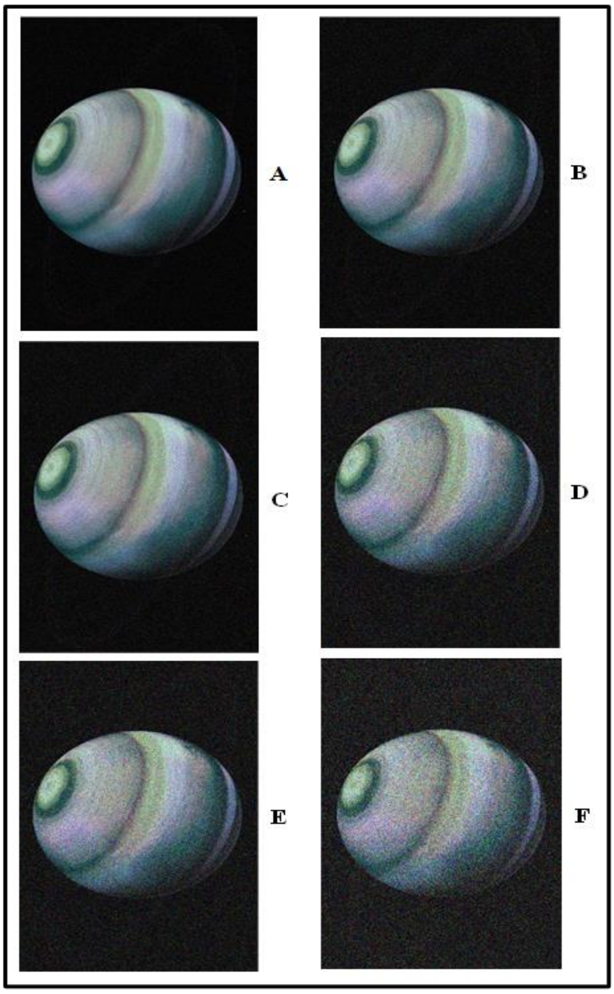

FIGURE 4. Showing Restored Image by (**A**) Proposed Filter in 10% noise ratio, (**B**) Proposed Filter in 50% noise ratio, (**C**) Proposed Filter in 75% noise ratio, (**D**) CNN in 10% noise ratio, (**E**) CNN in 50% noise ratio, (**F**) CNN in 75% noise ratio.

We can observe from the results in Table 2 that the order of filters about the restored image quality and the presence of various noise levels did not change. The first filter is the proposed one, which has 40.30 PSNR and 0.97 SSIM in 50% and 38.69 PSNR and 0.96 SSIM in 75% noise density, respectively. Whereas the CNN filter recorded 36.98 PSNR, 0.92 SSIM in 50% noise density, 34.52 PSNR, and 0.90 SSIM in 75% noise density.

The mechanism used by the proposed method to analyze the image, which was represented by a series of stages, is what accounts for its superiority. First, an appropriate bandwidth parameter was extracted using a plug-in method designed to minimize errors. The Gaussian density function was then estimated using this parameter by calculating the mean and variance of the noise contained in the image. Finally, a useful denoising method is to divide the image using thresholding.

5. Conclusions

This paper presents a new implementation approach for denoising Gaussian noise in satellite images, combining a convolutional neural network (CNN) with a novel filtering technique. The method converts noisy images to YUV color space, isolates the noisy Y channel, and uses cross-validation to estimate the density function of the Y channel. This method significantly enhances image quality by removing noise while preserving essential details. The method offers a robust framework for handling noise characteristics, demonstrating improvements in visual appearance and performance metrics.

References

1. Abdul Wadood, M., & Ghalib, A. (2024). Gaussian Denoising for the First Image from The James Webb Space Telescope “Carina Nebula” using Non-Linear Filters. *Journal of Economics and Administrative Sciences, 30*(143), 420-434.
2. Araveeporn, A. (2011). The Comparison of Bandwidth Selection Methods using Kernel Function. *KMITL Sci. Tech. J., 11* (2), 64-78. Retrieved from https://www.thaiscience.info/Journals/Article/KLST/10896091.pdf
3. Bora, P., & Chaudhary, K. (2021, 5). Improved Image Denoising Methodology using Deep CNN Bilateral Filter Compared to Additional Methods. *International Journal of Engineering and Advanced Technology (IJEAT), 10*(5), 191-196. doi:https://doi.org/10.35940/ijeat.d2372.0610521
4. Chac´on, J., & Duong, T. (2019). Multivariate plug-in bandwidth selection with unconstrained pilot bandwidth matrices. *International Journal of Development Research, 7*(9), 15048–15053. doi:https://doi.org/10.1007/s11749-009-0168-4
5. Dong, C., Loy, C., & Tang, X. (2015). Image Super-Resolution Using Deep Convolutional Networks. 1-14.
6. Dong, C., Loy, C., He, K., & Tang, X. (2014). Learning a Deep Convolutional Network for Image Super-Resolutio. 184–199.
7. Florence, K., Aggrey, A., & Leonard, K. (2018, 5 24). Efficiency of various Bandwidth Selection Methods across Different Kernels. *IOSR Journal of Mathematics (IOSR-JM), 15*(3), 55-62. doi:https://doi.org/10.9790/5728-1503015562
8. Ghalib, A., & Abdul Wadood, M. (2020). Using multidimensional scaling technique in image dimension reduction for satellite image. *Periodicals of Engineering and Natural Sciences, 8*(1), p.447-454.
9. Herbreteau, S., & Kervrann, C. (2021). DCT2net: an interpretable shallow CNN for image denoising. *Journal Of Latex Class Files, 14*(8), 1-13.
10. Hmood, Munaf. (2005). Comparing Nonparametric Estimators For Probability Density Estimation. PH.D. Dissertation University of Baghdad, College of Administration & Economics.
11. Hmood, Munaf. (2010), Comparing Variable Bandwidth Estimators Using Local Linear

Estimator with Application. Journal of Statistical Sciences, Vol. (3), PP1-21.

12. Horova, I., Kolacek, J., Zelinka, J., & Vopatova, K. (2008, 12). Bandwidth choice for kernel density estimates. 1-10. Retrieved from https://link.springer.com/article/10.1007/s10182-013-0216-y

13. Ilesanmi, A. E., & Ilesanmi, T. O. (2021). Methods for image denoising using convolutional neural network: a review. *Complex & Intelligent Systems, 7*, 2179–2198. doi:https://doi.org/10.1007/s40747-021-00428-4

14. Li, M.-M., & Li, B.-Z. (2021). A novel weighted total variation model for image denoising. *IET Image Process, 15*, 2749–2760. doi:https://doi.org/10.1049/ipr2.12259

15. Li, Y., & Chen, Y. (2023). NSNet: An N-Shaped Convolutional Neural Network with Multi-Scale Information for Image Denoising. *11*(2772), 1-19. doi:https://doi.org/10.3390/math11122772

16. Malik, J., Kiranyaz, S., & Gabbouj, M. (2020). Operational vs Convolutional Neural Networks for Image Denoising. 1-16.

17. Ortelli, F., & van de Geer, S. (2020). Adaptive Rates for Total Variation Image Denoising. *Journal of Machine Learning Research, 21*, 1-38.

18. Pana, H., Wena, Y.-W., & Zhu, H.-M. (2019, 11 23). A regularization parameter selection model for total variation-based image noise removal. *Applied Mathematical Modelling, 68*, 353-376. doi:https://doi.org/10.1016/j.apm.2018.11.032

19. Pm, N., & Chezian, R. M. (2013, 1). Various Colour Spaces And Colour Space Conversion Algorithms. *Journal Of Global Research In Computer Science, 4*(1), 44-48. Retrieved from http://www.jgrcs.info/

20. Rothfuss, J., Ferreira, F., Walther, S., & Ulrich, M. (2019, 4 3). Conditional Density Estimation with Neural Networks: Best Practices and Benchmarks. 1-36. Retrieved from https://arxiv.org/abs/1903.00954

21. Salgado-Ugarte, I. H., & Perez-Hernandez, M. A. (2003). Exploring the use of variable bandwidth kernel density estimators. *The Stata Journal, 3*(2), 133-137. Retrieved from https://www.stata-journal.com/abstracts/st0036.pdf

22. Sharma, A., Lall, U., & Tarboton, D. G. (1998). Kernel bandwidth selection for a first-order nonparametric streamflow simulation model. *Statistic Hydrology and Hydraulics, 12*, 33-52. Retrieved from https://link.springer.com/article/10.1007/s004770050008

23. Thayammal, Sankaramalliga, Priyadarsini, & Ramalakshmi. (2021). Performance Analysis of Image Denoising using Deep Convolutional Neural Network. *IOP Conf. Series: Materials Science and Engineering, 1070*, 1-8. doi:https://doi.org/10.1088/1757-899X/1070/1/012085

24. Tian, C., Xu, Y., & Zuo, W. (2020). Image denoising using deep CNN with batch renormalization. *Neural Networks, 121*, 461-473. doi:https://doi.org/10.1016/j.neunet.2019.08.022

25. Tian, C., Xu, Y., Fei, L., Wang, J., Wen, J., & Luo, N. (2018, 2 7). Enhanced CNN for image denoising. *CAAI Transactions on Intelligence Technology, 7*(4), 1-7. doi:https://doi.org/10.1049/trit.2018.1054

26. Varol, E., & Nejatbakhsh, A. (n.d.). Wasserstein Total Variation Filtering.

27. Xie, S., Song, J., Hu, Y., Zhang, C., & Zhang, S. (2023, 5 20). Using CNN with Multi-Level Information Fusion for Image Denoising. *Electronics, 12*(6), 2146–2146. doi:https://doi.org/10.3390/electronics12092146

28. You, S. J., & Cho, N. I. (2013). An adaptive bandwidth nonlocal means image denoising in the wavelet domain. *o EURASIP Journal on Image and Video Processing, 60*, 1-22. Retrieved from http://jivp.eurasipjournals.com/content/2013/1/60

29. Zeng, X., & Li, S. (2013). An efficient adaptive total variation regularization for image denoising. *Conference: Image and Graphics (ICIG), 7*, 1-6. doi:https://doi.org/10.1109/ICIG.2013.17

30. Zhang, K., Zuo, W., & Zhang, L. (2018). FFDNet: Toward a Fast and Flexible Solution for CNN-based Image Denoising. 1-15.

31. Zhang, Q., Xiao, J., Tian, C., Lin, J., & Zhang, S. (2022). A robust deformed convolutional neural network (CNN) for image denoising. *CAAI Transactions on Intelligence Technology*, 331-342. doi:https://doi.org/10.1049/cit2.12110

32. Zheng, M., Zhi, K., Zeng, J., Tian, C., & You, L. (2022). A Hybrid CNN for Image Denoising. *Journal of Artificial Intelligence and Technology*, *2*, 93-99. doi:https://doi.org/10.37965/jait.2022.0101

33. Zhu, J., Li, K., & Hao, B. (2019, 5 6). Image Restoration by Second-Order Total Generalized Variation and Wavelet Frame Regularization. *Complexity*, 1-17. doi:https://doi.org/10.1155/2019/3650128