Journal of Computer Science

Research Article

A Detailed Preserving Medical Image Denoising Using Cluster-Wise PCA Thresholding and Iterative Mean Filtering

Mohit Sharma[1], **Ayush Dogra**[2], **Anita Gupta**[1], **Bhawna Goyal**[3], **Dawa Chyophel Lepcha**[4] **and Archana Saini**[5]

[1]*Department of Allied Health Sciences, Chitkara School of Health Sciences, Chitkara University, Punjab-140401, India*
[2]*Department of Electronics and Communication Engineering, Chitkara University Institute of Engineering & Technology, Chitkara University, Rajpura, Punjab, India*
[3]*Department of Engineering, Marwadi University Research Centre, Marwadi University, Rajkot, Gujarat, India*
[4]*Graduate Institute of Biomedical Informatics, College of Medical Science and Technology, Taipei Medical University, New Taipei, 235, Taiwan*
[5]*Department of Computer Science & Engineering, Chitkara University Institute of Engineering & Technology, Chitkara University, Rajpura, Punjab, India*

Article history
Received: 11-04-2025
Revised: 12-01-2026
Accepted: 20-01-2026

Corresponding Author:
Ayush Dogra
Department of Electronics and Communication Engineering, Chitkara University Institute of Engineering & Technology, Chitkara University, Rajpura, Punjab, India
Email: ayush123456789@gmail.com

Abstract: Image denoising is a vital process in medical imaging that involves removing noise or distortions introduced during image acquisition. Random noise can degrade image quality and reduce contrast, making accurate analysis more challenging. This paper presents a detail-preserving medical image denoising framework based on a hybrid methodology that combines cluster-wise Principal Component Analysis (PCA) thresholding with Iterative Mean Filtering (IMF). The process begins with noise level estimation followed by the extraction of overlapping image patches. These patches are then grouped using K-means++ clustering to form structurally homogeneous clusters, enabling localized processing. Within each cluster, PCA is applied to transform the data into a low-dimensional subspace, and an adaptive threshold is computed using the Marchenko Pastur (MP) law to separate noise from significant image features. The denoised patches are then reconstructed by applying inverse PCA. To further enhance structural continuity and reduce residual noise, Iterative Mean Filtering is employed both before and after PCA denoising. The final denoised image is reconstructed through weighted averaging of the processed patches. The modular structure of the algorithm allows dynamic thresholding, cluster-specific denoising, and iterative refinement, making the approach adaptive to varying noise levels and anatomical structures. This methodology is evaluated against state-of-the-art denoising methods using both full-reference metrics (PSNR, SSIM, FSIM, Entropy) and no-reference quality measures (BRISQUE, NIQE, PIQE). Results demonstrate that the proposed method achieves superior noise suppression while preserving fine details, offering an efficient and interpretable solution for clinical image Enhancement.

Keywords: Medical Imaging, Image Denoising, Iterative Mean Filter, Adaptive Patch Clustering, Principal Component Analysis (PCA) Transform, Evaluation Metrics

Introduction

Noise refers to unwanted energy that generally interferes with an image during its acquisition, transmission, or reconstruction. While it is challenging to eliminate noise entirely, its impact can be minimized during the acquisition stage. Additionally, post-processing techniques leveraging signal processing algorithms are commonly employed to suppress noise and enhance image quality. In these scenarios, image denoising poses a significant setback for scholars (Buades et al., 2005a) as it is considered a poorly defined problems (Fan et al., 2019). Traditionally, this problem is tackled by defining the forward model and attempting to transpose it to estimate an original signal. However, recent advancements have seen the emergence of several image denoising methods as powerful tools for image denoising (Nazir et al., 2024). This task is particularly crucial in medical diagnostics (Kaur and Dong, 2023), where clinicians depend heavily on high quality images for accurate assessments. The medical imaging techniques such as Computed Tomography (CT), Magnetic Resonance Imaging (MRI), ultrasound, and Low-Dose Computed Tomography (LDCT) are extensively utilised for visualization of internal structures and detect abnormalities like tumours or injuries within the body. Medical images are typically influenced by noise resulting from temperature variations in the scanning environment, disturbances in the scanning equipment, and patient movements dusting image acquisition. Noise causes disturbances in in pixel magnitudes within the image thus resulting in artifacts and a loss of detail. It complicates diagnostic and disease diagnosis. Primary considerations in medical image diagnosing methods are:

a) The preservation of edges in the denoised images and ensuring that filtering does not obscure nearby details
b) The maintenance and enhancement of the perceptual quality of the denoised images
c) The minimization of signal leakage in the denoised images (Nazir et al., 2024)

Numerous medical image denoising techniques have been introduced throughout the years (Kaur and Dong, 2023). Miri et al. (2018) proposed a novel method to remove noise from medical images employing a 2D-discrete cosine transform and the ACO system. This approach aimed to uncover significant frequency coefficients through ant colony optimization while mitigating noise effects by eliminating high-frequency components. Laves et al. (2020) employed a randomly initiated convolutional network to parameterize the restored image and utilize gradient descent to align it with the observation, a method referred to as deep image prior. In the instance, the reconstruction is lack of hallucinations due to the absence of prior training. They apply a Bayesian methodology with Monte Carlo dropout for accessing both epistemic and aleatoric uncertainty. Geng et al. (2022) proposed a simple yet efficient method known as content noise complementary learning method wherein two deep learning predictors are utilised for learning the information of an image in a complementary manner. Kollem et al. (2023) devised a method for image denoising that utilizes a diffusivity function based partial differential equation thus leveraging the statistical features of observed noisy images. This method uses a QWT that produces the numerous components of a noisy image. The soft threshold function uses an enhanced generalised cross-validation method to determine the ideal threshold values. The ideal threshold values are subsequently employed to regulate the diffusion procedure via a diffusivity function. Qu et al. (2024) sought to refine the BWO method and implement it for denoising medical images, thereby enhancing both the quality of the images and the precision of diagnostic performance. They enhanced the BWO method by incorporating Tent mapping to improve its adaptability to the rich characteristics of medical imagery. Taassori and Vizvari (2024) utilized an adaptive Kalman filter to mitigate noise, capitalizing on its expertise in state computation from noisy observations. In contrast to traditional Kalman filters with static parameters, an adaptive Kalman filtering modifies its parameters in response to the noise features of the raw input, hence providing improved precision in computing the true state of the system demonstrated by the medical images. Then, a NLM approach and a median filter are employed as post-processing techniques to enhance the denoising image. Diwakar et al. (2024) presented a weighted function used by the fractional order total variation to denoise CT images, thus addresses the blocky effect and resolving the non-convex optimisation problems for enhanced solutions. Lepcha et al. recently proposed low-dose CT image denoising to eliminate noise from medical images (Lepcha et al., 2024a; Goyal et al., 2024).

Recently, the deep learning methodology has been investigated alongside classical techniques for the denoising of medical images (Manjon and Coupe, 2018). Supervised learning is used for improving the quality of medical images by eliminating noise and minimizing artifacts (Higaki et al., 2019). A directionality component has recently been incorporated to improve vocabulary learning for MRI image reconstruction (Arun et al., 2020). Additionally, a minimax issue arises if an appropriate sparse thresholding method is not selected for estimating the sparse components from the dictionary and the given images (Singh et al., 2018). Semi-supervised deep-learning method is employed to remove distortion in medical images without utilizing original projection data by training the model with a limited set of the denoised images (Cheplygina et al., 2019) . Low-dose CT images are aligned with their corresponding normal dose CT sections in a patch-by-patch approach utilizing a DCNN. For low-dose CT images, a RED-CNN is constructed using autoencoders and a deconvolutional network thus facilitating noise reduction while preserving structure and detecting lesions (Chen et al., 2017). This method employs normal dose and low dose CT images to train the model. A recent study on GAN has been conducted for denoising medical images (Yi et al., 2019). A GAN is adapted in the Wasserstein GAN (WGAN) to facilitate the denoising of medical images in a semi supervised approach. A convolutional encoder-decoder based on conveying paths is used for denoising low-dose CT images utilizing the principle of transfer learning within a GAN framework (Shan et al., 2018). MAP-NN is used for progressively denoise the low-dose CT images. It is a widely recognized reality in medical imaging that the scarcity of training and original datasets hampers the training of models in supervised or semi-supervised frameworks. Consequently, a more effective strategy is to explore unsupervised learning models capable of autonomously learning from the existing datasets and generating high quality denoised medical images. Significant focus is being directed towards LDCT due to its capacity to mitigate patient risk. To yield promising outcomes for CT images while preserving essential information, GAN is integrated with visual similarity and Wasserstein distance utilising unsupervised learning (Yang et al., 2018). A DNN has recently been trained in an unsupervised manner utilizing the PURE for denoising the LDCT images (Kim et al., 2020; Sharif et al., 2020) mitigates this challenging denoising process by acquiring residual noise from a considerable volume of data samples. This method expedites the learning procedure by implementing an innovative deep network that utilizes the attention mechanism to leverage feature correlation and integrates it with spatially refined residual features. Luthra et al. (2021) proposed E former, which designs an encoder-decoder network utilizing transformer blocks for the purpose of denoising medical images, Transformer blocks use non-over lapping window based self-attention to decrease computational demands. This study additionally integrates learnable Sobel Feldman operates for enhancing image edges and proposes an efficient method for concatenating them within the intermediate layers of our design.

Rai et al. (2021a) present an unsupervised learning paradigm that improves denoising by integrating the limitations of reinforcement learning with the advantages of deep learning and vice versa. They present a ten-layer DRN augmented with patch-based dictionaries. The source images are provided to patch based deep learning for indirectly acquire noise through sparse representation while simultaneously being transfer to the denoising residual network to directly learn the noise. Propose a DCNN based denoising method to remove gaussian an s speckle noise across several medical imaging modalities. This method employs BN and ReLU as fundamental components. Therefore, it achieved significant advancement compared to conventional methods. Rajesh and Kumar (2022) proposed a differential evolution based automatic network evolution system to optimise network frameworks and hyperparameters by selecting the most effective parameters. In addition, the method implemented a transfer learning methodology to expedite the training procedure. The proposed evolutionary technique is adaptable and identifies promising network designs utilizing established techniques i.e. residual and dense blocks. Singh et al. (2022) devised a method to eradicate corrupted Gaussian noise from CT images while preserving essential features. This method integrates the idea of method noise with a deep learning design utilizing a CNN. Noisy images are generated through the deliberate incorporation of Gaussian noise at different noise levels. Annavarapu and Borra (2024) proposed an adapted DCNN for image denoising for image denoising combined with adaptive watershed segmentation for restoring image details on the hybrid lifting system based on enhanced bi-histogram equalised contrast enhancement. Thus, method is subsequently enhanced with the incorporation of marker-based watershed segmentation. Ferdi et al. (2024) suggested a methodology utilizing high frequency learning to improve the denoising efficiency of current methods. This approach aims for concurrently learn the principal goal of denoising LDCT images and the supplementary task of low-dose CT edge detection thus enhancing denoising efficiency without enhancing model parameters or inference duration. Zeng et al. (2024) presented a triple neural networks collaboration continuity denoising model. It employs triplet neural networks for cooperative mutual updates. The knowledge from two denoising networks which possess ongoing learning capabilities is conveyed to the primary denoising network. This primary denoising network possesses novel information and can reinforce existing knowledge. The objective is to fulfil the essential requirement for efficient image denoising in medical imaging. Niknejad et al. (2025) introduced a novel conditioning module for diffusion method that utilizes image construction priors from the sinogram domain to produce rich features. They further implement a 2-phase training process where the network progressively acquires knowledge about various anatomical textures and structures. Swetha and Jyothi (2025) presented a multilevel CNN with an optimized visual attention network (MCVAN) for image denoising. Then, a Leopard Seal Optimization (LSO) refines the network parameters thus improves denoising performance.

The main contributions of this paper are summarised as follows:

- The study proposes a robust noise level estimation approach that combines a noise estimation approach with a Marchenko-Pastur law. The integration enhances the stability of our approach particularly under high noise conditions. Experimental performance denotes that the proposed approach obtains superior accuracy in estimating noise levels in medical images compared to other exiting methods
- This study proposes an Iterative Mean Filter (IMF) which progressively reduces noise by repeatedly averaging each pixel with its neighbouring pixels. In each iteration, the filter smooths the image further, effectively suppressing high-frequency noise while preserving essential low-frequency image structures
- This study demonstrates the efficiency of the adaptive patch clustering approach under high noise conditions by introducing a custom merging threshold. Unlike traditional approaches that rely solely on noise level, the proposed threshold is designed as a function of both the noise level and cluster size thus resulting in improved clustering performance in noisy environments
- This study proposes a progressive thresholding filtering for denoising adaptively clustered image patches in the PCA transform domain. The method enhances denoising in signal-dominant dimensions by employing a coefficient-wise suboptimal Weiner

filter where the filter parameters adaptively track texture variations. To further improve denoising performance, a sliding window and aggregation method with a fixed window size is also incorporated

Iterative Mean Filter IMF)

As illustrated in Thanh et al. (2019), $Let\ U \coloneqq [u_{ij}]_{m\times n}$ represent a noise-free ground truth image, where $\delta_{min} \le u_{ij} \le \delta_{max\,,}$ for all pixel coordinates $(i,j) \in I = \{1,2,\dots,m\} \times I = \{1,2,\dots,n\}$. In this context, m represents the number of pixels in a row of U, n denotes a number of pixels in the column of U and $[\delta_{min}\,, \delta_{max}\,]$ indicates the range of grey values of U. For instance, $\delta_{min} = 0$ and $\delta_{max} = 255$ for an 8-bit greyscale image it is observed that, for natural images, the quantity of pixels attaining the boundary values $\delta_{min}\,, \delta_{max}$ is minimal. The noisy representation of the ground truth (noise-free) image is denoted as $B \coloneqq [b_{ij}]_{m\times n}$ by adding Gaussian noise at varying noise levels (σ).

Suppose $r \ge 1$ be an integer and $A = [a_{ij}]_{m\times n}$represent an image. In image A, the index set of a window measuring $(2r+1)\times(2r+1)$ centred at the pixel position (i,j), represented by $W_{ij}(A,r)$ which is illustrated as:

$$\{(i^*,j^*) \in I : |i^* - i| \le r, |j^* - j| \le r\} \qquad (1)$$

Again, r is an integer such that $r \ge 1$, and $A = [a_{ij}]_{m\times n}$ represent an image. In image A, the strict indices set of a window size $(2r+1)\times(2r+1)$ centred at the pixel location (i,j), abbreviated by $W^*_{ij}(A,r)$ is illustrated as:

$$\{(i^*,j^*) \in W_{ij}(A,r) : a_{i^*j^*} \ne \delta_{min}\,, a_{i^*j^*} \ne \delta_{max}\,\} \qquad (2)$$

Let $A = [a_{ij}]_{m\times n}$represent an image. The restricted mean of $W_{ij}(A,r)$, represented as $\bar{W}_{ij}{}^{mean}(A,r)$ is expressed as follows:

$$\begin{cases} a_{i,j}\, W^*_{ij}(A,r) = \emptyset \\ \frac{1}{card\left(W^*_{ij}(A,r)\right)} \sum_{(i^*,j^*)\in W^*_{ij}(A,r)} W^*_{ij}(A,r) \ne \emptyset \end{cases} \qquad (3)$$

Where $card(\cdot)$ represents a set cardinality, i.e. the number of pixels.

If $A \coloneqq [a_{ij}]_{m\times n}$ and $A^* \coloneqq [a^*_{ij}]_{m\times n}$ represents the two matrices. The l_1 -distance, often known as the Manhattan distance (Peiravi and Kheibari, 2008, between A and A^* is illustrated as follows:

$$|A - A^*|_1 \coloneqq \sum_{i=1}^{m}\sum_{j=1}^{n}|a_{i,j} - a^*_{i,j}|, \qquad (4)$$

Methods

The proposed algorithm begins with a noise level estimation which is a necessary stage for effective noise reduction. This is followed by the application of iterative mean filtering for initial pre-denoising. Next, adaptive patch clustering is performed on the stacked patches for grouping structurally similar patches more accurately. The denoising process then continues with cluster-wise PCA thresholding. Finally, a sliding window and aggregation technique is employed to refine the results. A flowchart illustrating the complete workflow of the proposed approach is demonstrated in Fig. 1.

Noise Estimation Using PCA and MP Law

Noise level is a crucial factor in ethe complete image denoising process as illustrated in Zhao et al. (2018). The optimal results of the PCA based noise level estimate is demonstrated in Pyatykh et al. (2013); Liu et al. (2013); Chen et al. (2015). In Pyatykh et al. (2013); Liu et al. (2013) low-rank patch selection incurs a significant computational overhead and exhibits instability at elevated noise levels. Chen et al. (2015) presented a fast approach for estimating noise levels reputable on the finding the patches extracted from a noiseless image frequently positioned within a low-dimensional subspace. A low-dimensional subspace could be obtained by the low-rank approximation of PCA while the noise level could be obtained from the eigenvalues of the covariance matrix of noisy patches. Inspired by this research, this study proposes a robust and stable method for noise level estimation by merging this strategy with the MP law. The correlation between noise level σ and the PCA eigenvalue λ of the random matrix N allows the MP law to accurately predict the gaussian noise level of the whole image. MP law has been used for locally assess the noise level (Veraart et al., 2016a). Compared to Veraart et al. (2016b), our approach utilizes the MP law for a comprehensive estimation of noise levels. The global approach is justified as a MP law defines the characteristics of the large matrices with greater precision than those of small matrices and a subscript $\emptyset$ to denote the complete image. To obtain highly precise estimates, we partition the entire $a \times b$ image into over-lapping $d_\emptyset \times d_\emptyset$ patches and aggregate all the overlapping patches to form a large matrix $X_\emptyset \in \mathbb{R}^{M_\emptyset \times L_\emptyset}$ where $M_\emptyset = d^2_\emptyset$ and $L_\emptyset = (a - d_\emptyset + 1)(b - d_\emptyset + 1)$. A large noisy matrix $X_\emptyset$ is mathematically characterized as a low rank matrix $X_{0,\emptyset}$ enhanced by a noisy matrix $N_\emptyset$; therefore, $N_\emptyset$; $X_\emptyset = X_{0,\emptyset} + N_\emptyset$, where each column of $N_\emptyset$ is a vector that follows to a multivariate Gaussian distribution $N_{M_\emptyset}(0, \sigma^2 I)$ with a mean of 0 and covariance matrix of $\sigma^2 I$. The estimation of noise level σ is directly correlated with the PCA eigenvalues of $N_\emptyset$. Nonetheless, $N_\emptyset$ remains unknown while $X_\emptyset$ is established.

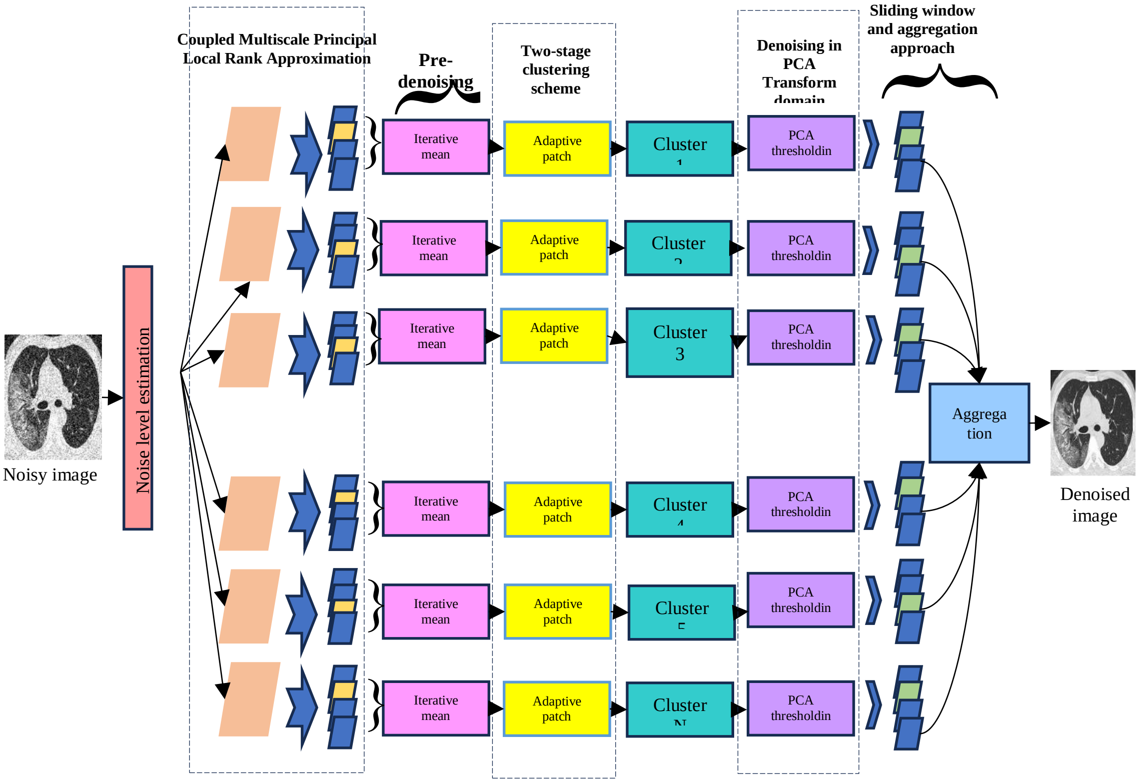

Fig. 1: Detailed methodological framework of the proposed method

The low-rank of $X_{0,\emptyset}$ allows us to examine the link across the PCA eigenvalues of $N_\emptyset$ and those of $X_\emptyset$. Utilizing the PCA eigenvalues $\lambda_{x_\emptyset,i} 1 \le i \le M_\emptyset$ of $X_\emptyset$, we may determine the noise level σ through the application of the subsequent two equations. Initial equation (Equation 19) could be derived as (Zhao et al., 2018). The statistical mean of the observed eigenvalues between $\lambda_{n_\emptyset -}$ and $\lambda_{n_\emptyset +}$ is almost equivalent to the predicted value of $\lambda_{n_\emptyset}$:

$$\sigma^2 = E\left(\lambda_{n_\emptyset}\right) \approx \frac{\sum_{i=1}^{M_\emptyset} \lambda_{x_\emptyset} 1\left(\lambda_{n_\emptyset -} \le \lambda_{x_\emptyset,i} \le \lambda_{n_\emptyset +}\right)}{\sum_{i=1}^{M_\emptyset} 1\left(\lambda_{n_\emptyset -} \le \lambda_{x_\emptyset,i} \le \lambda_{n_\emptyset +}\right)} \tag{5}$$

The second equation can be derived from $\lambda_{n_\emptyset \pm} = \sigma^2\left(1 \pm \sqrt{\gamma\emptyset}\right)^2$ with $\gamma\emptyset = \frac{M_\emptyset}{L_\emptyset}$:

$$\sigma^2 = \lambda_{n_\emptyset -} - \lambda_{n_\emptyset +} \tag{6}$$

Where Equations (5) and (6) indicate that estimating the noise level involves identifying the range $\left[\lambda_{n_\emptyset -}, \lambda_{n_\emptyset +}\right]$. We could directly establish that $\lambda_{n_\emptyset -} = \lambda_{x_\emptyset}, M_\emptyset$ due to $\lambda_{n_\emptyset -} \le \lambda_{x_\emptyset}, M_\emptyset \le \lambda_{x_\emptyset,i}, 1 \le i \le M_\emptyset$. Subsequently, we determine a suitable $\lambda_{n_\emptyset +}$ by heuristic greedy search accordance to Equations (5) and (6). For each $\lambda_{x_\emptyset,i}$, if $\lambda_{n_\emptyset +} = \lambda_{x_\emptyset,i}$, we can compute the variance estimation $\sigma_{i_1}^2$ using Equation (5) and $\sigma_{i_2}^2$ using Equation (6), respectively. We then compute the difference $\Delta_i = \left\|\sigma_{i_1}^2 - \sigma_{i_2}^2\right\|_2^2$. The optimal $\lambda_{n_\emptyset +}$ is determined by selecting the suitable $\lambda_{x_\emptyset,i}$ that reduces the discrepancy Δ_i. After computing of $\lambda_{n_\emptyset \pm}$, the noise level $\hat{\sigma}$ is calculated in accordance with Equation (5).

Iterative Mean Filtering

Given the global noise level $\hat{\sigma}$, we than perform an Iterative Mean Filter (IMF) (Thanh et al., 2019) as illustrated in Section 2. It is designed to restore a noisy image B by iteratively refining an estimated restored image. The algorithm begins by initializing the iteration step $k = 0$ setting $A^{[0]} = B$, and defining a threshold ε. It also determines a maximum and minimum intensity values of the noisy image denoted as δ_{min} and δ_{max}, respectively. The algorithm proceeds iteratively. At each iteration step k, for each pixel (i, j) in an image $A^{[k]}$, it checks whether the pixel intensity $a_{i,j}^k$ falls outside the range defined by δ_{min} and δ_{max}. If the pixel intensity exceeds δ_{max}, or is lower than δ_{min}, a weighted mean filter $\overline{W}_{ij}^{mean}(A, r)$ is applied to compute the new pixel value. The updated value replaces the original one at that pixel position. If the intensity falls within the acceptable range, the pixel value remains unchanged. This process is repeated until the difference between successive image estimates $\left|A^{[k+1]} - A^{[k]}\right|_1$ becomes less than or equal to ε, indicating convergence. The final output is the restored image A, which is an augmented version of the original noisy image B.

Adapting Patch Clustering

After the pre-denoising step using iterative mean filtering, the image is divided into overlapping patches of fixed size (Zhao et al., 2018). Each patch is then vectorized to facilitate clustering in a high-dimensional space. The primary objective of this stage is to group structurally similar patches together so that Principal Component Analysis (PCA) can be applied independently within each cluster. This process of clustering makes it easier to separate noise and keep local image details better. A two-stage clustering method is utilized to get small changes in texture and structure (Abdi and Williams, 2010). The K-means method is used in the first stage to implement an over-clustering technique. This makes it easier to tell the difference between different sorts of patches. After that, clusters that are similar are combined based on a distance criterion for the centroid. The Euclidean distance between the centroids of two clusters is used to determine how similar they are. If this distance is smaller than a certain threshold ξ, which changes based on the estimated degree of noise, the two clusters are combined:

$$\xi = 16\sigma^2 \tag{7}$$

Where, σ represents the estimated global noise standard deviation obtained using the Marchenko–Pastur (MP) law (Donoho and Gavish, 2013). To avoid merging excessively large clusters that might include heterogeneous structures, a refinement is applied. If a cluster size exceeds a pre-defined limit L_T, typically set to 150 patches, the merging threshold ξ is scaled using a factor $\rho = 0.7$, resulting in a modified threshold:

$$\xi' = \rho . \xi \tag{8}$$

This adaptive merging strategy ensures that clusters maintain internal structural homogeneity, which is essential for the performance of subsequent PCA-based denoising.

K-Means Clustering Implementation Details

There are a number of mathematical improvements made to the K-means clustering process to make sure it is strong, fast, and good for removing noise from images. The K-means++ (Arthur and Vassilvitskii, 2006) approach is used to set the starting points for the clusters. This method moves the starting centroids around to make convergence better and lower the chance of becoming stuck in local optima. We use Euclidean distance to group patches together, and each patch is shown as a d-dimensional vector. The number of initial clusters K is set dynamically based on the total number of patches N and the average cluster size *M* (Zhang et al., 2010). This value is usually between 60 and 100. To find the value of *K*:

$$K = \frac{N}{M} \tag{9}$$

This strategy ensures that each cluster contains a manageable number of structurally similar patches. The clustering algorithm iterates until either the change in centroid positions is less than a convergence threshold of 10^{-4} (measured using the L2 norm), or a maximum of 100 iterations is reached. Following initial clustering, cluster merging is conducted as described in Section 3.2. Clusters whose centroids are closer than the threshold ξ (or ξ′ for large clusters) are merged to produce the final cluster set. This controlled and noise-adaptive merging process ensures high intra-cluster similarity, which is critical for the effectiveness of PCA in the denoising stage. Together, the adaptive patch clustering and refined K-means implementation form a robust foundation for localized, detail-preserving PCA-based image denoising.

Cluster-Wise PCA Thresholding

Once the adaptive clustering of image patches is completed, each resulting group undergoes denoising through Principal Component Analysis (PCA) (Zhao et al., 2018; Abdi and Williams, 2010). Since patches within a cluster share similar structural characteristics, applying PCA locally enhances the separation of signal and noise, especially in texture-rich and edge-preserving regions of the image. PCA works by transforming the data into a subspace where each component captures a principal direction of variance. The higher-variance components are more likely to represent meaningful image features, while the lower-variance ones typically capture noise. Each cluster is represented as a matrix $X \in \mathbb{R}^{n \times d}$ where n is the number of patches and d is the dimensionality of each patch. The mean patch $\bar{x}$ is subtracted from each row to center the data:

$$X_c = X - \bar{x} \tag{10}$$

The covariance matrix C is then computed from the centred data:

$$C = \frac{1}{n} X_c^T X_c \tag{11}$$

Eigenvalue decomposition of *C* yields eigenvectors and eigenvalues, corresponding to the principal components and the amount of variance they explain. To retain only the signal-dominant components, a threshold based on the Marchenko–Pastur law (Donoho and Gavish, 2013) is used. Components with eigenvalues greater than a scaled noise threshold λ_+ are preserved:

$$\lambda_i > \mu . \lambda_+ \tag{12}$$

Where, λ_i denotes the i^{th} eigenvalue, λ+ is the MP-derived upper bound for noise-only components, and μ is an empirical scaling factor typically in the range (1.2, 1.5). This criterion ensures that noisy directions are excluded from reconstruction. The coefficients corresponding to the retained components are then denoised using a suboptimal Wiener filter. Unlike the ideal Wiener filter, which may lead to over smoothing, the suboptimal variant retains important high-frequency content. The filtering is guided by local variance estimates σ_{local}^2 obtained via the LPA-ICI technique (Katkovnik et al., 2010). The filtered coefficient $\hat{y}_i$ is computed as:

$$\hat{\mathrm{y}}_{\mathrm{i}} = \alpha.\frac{\sigma^2_{\mathrm{signal}}}{\sigma^2_{\mathrm{signal}}+\sigma^2_{\mathrm{noise}}}.\mathrm{y}_{\mathrm{i}} \quad (13)$$

In this equation, y_{i} is the noisy coefficient, σ_{signal}^2 is the estimated signal variance, σ_{noise}^2 is the noise variance, and α is a filtering attenuation factor, typically between 0.7 and 0.9. Finally, the filtered PCA coefficients are transformed back to the original space using the inverse PCA transform. This process is repeated independently for all clusters, and the resulting denoised patches are prepared for aggregation.

PCA Implementation and Parameter Tuning

The PCA denoising process for each patch cluster is performed by a systematic process which includes centering the data, choosing a subspace, adaptive filtering, and inverse transformation. The first step is to get the mean patch $\hat{x}$ and subtract from each patch to make sure the data is centered around zero. Next, the covariance matrix is produced, and eigen-decomposition gives us the tools we need to change the patch data into its main parts. Choosing the right subspace is very important for proper denoising. The Marchenko–Pastur distribution is used to tell the difference between signal and noise. It sets the highest value λ_+ for eigenvalues when there is noise. We keep any primary component whose eigenvalue is greater than the scaled threshold $\mu.\lambda_+$ and the rest are discarded. The value of μ is set depending on how well it works in practice; a typical range is from 1.2 to 1.5 which lets to separate noise from signal in real-world data. Then, the kept PCA coefficients via suboptimal Wiener filtering (Luisier et al., 2011; Portilla et al., 2003) which lowers each coefficient based on the ratio of signal to total variance. The LPA-ICI (Local Polynomial Approximation with Intersection of Confidence Intervals) approach estimates the local variance needed for filtering. It does this by changing the size of the neighbourhood dependent on the structure of the data. The experimentally set attenuation factor α changes the strength of the filtering based on the amount of noise; lower values are utilized when there is a lot of noise.

After filtering, the coefficients are projected back into the spatial domain via the inverse PCA transformation, and the denoised patches are reconstructed. This procedure ensures high-fidelity noise reduction while preserving edge and texture information, making it particularly well-suited for sensitive medical imaging applications. o enhance clarity, the clustering and PCA thresholding pipeline is explicitly defined as follows. For clustering, we employ K-means++ initialization with the Euclidean distance metric applied to vectorized $s \times s$ overlapping patches. The number of initial clusters is determined adaptively as $K = \lfloor N/M \rfloor$, where N is the total number of patches and M ∈ [60,100] represents the desired average cluster size. The clustering process terminates when the centroid shift falls below 10^{-4} (L2 norm) or after 100 iterations. A noise-adaptive merging strategy is then applied: Clusters with centroid distance less than $\xi = 16\sigma^2$ (where σ is the globally estimated noise variance) are merged. If a cluster exceeds the size threshold L_T = 150, the merging threshold is reduced to $\xi' = \rho.\xi$ with ρ = 0.7 thereby ensuring structural homogeneity.

For PCA thresholding, each cluster is processed independently by centering the patch data, computing the covariance matrix, and retaining principal components with eigenvalues greater than a scaled Marchenko Pastur (MP) upper bound: $\lambda_i > \mu\,\lambda_+$, where $\mu \in [1.2,1.5]$ (default $\mu = 1.3$). The MP bound λ_+ is derived from the patch matrix aspect ratio γ and the estimated noise variance. Retained coefficients are further denoised using a sub-optimal Wiener filter, with local variance estimated via LPA-ICI and attenuation factor α ∈ [0.7,0.9]. Finally, inverse PCA reconstruction and weighted aggregation are applied to obtain the denoised image.

Patch Aggregation and Reconstruction

After each patch has been denoised using cluster-wise PCA thresholding, the last step is to put the denoised image back together from these overlapping patches (Dabov et al., 2007). Because the patch extraction procedure uses overlapping windows, there may be more than one estimate for the same pixel location (Buades et al., 2005b). A weighted aggregation approach is utilized to make sure that the reconstruction goes smoothly and is correct. At each pixel point, this approach takes the contributions of all overlapping patches and normalizes the result by the number of contributions. This kind of plan reduces artifacts at the edges and makes sure that overlapping areas are consistent with each other. A binary mask keeps track of how much each denoised patch adds to each pixel location. Each denoised patch only adds to the area it covers. To get the final intensity value for each pixel in the reconstructed image, you add up all the patch contributions at that position and divide by the total weight or count of the patches that contributed. The reconstructed image $I_{\text{denoised}}(x,y)$ at pixel location (x,y) is computed using the following Equation:

$$I_{\text{denoised}}(x,y) = \frac{\sum_{i=1}^{n} W_i(x,y).P_i(x,y)}{\sum_{i=1}^{n} W_i(x,y)} \quad (14)$$

Here, $P_i(x,y)$ is the value of the i-th patch at pixel (x,y), $W_i(x,y)$ is a binary weight indicating whether patch i contributes to pixel (x,y), and n is the total number of patches. This patch aggregation mechanism ensures that the reconstructed image preserves spatial continuity and benefits from the redundancy introduced by overlapping patches.

Parameter Selection and Tuning

The proposed framework has a number of factors that affect how well it removes noise at different levels of processing. These are the patch size, the number of clusters, the merging thresholds, the eigenvalue selection criteria, and the strength of the Wiener filter. For balancing noise reduction with structural preservation, especially in medical imaging where diagnostic features must stay intact, it is important to properly tune these parameters (Zhang et al., 2010). The patch size $s \times s$ times is an important setting that influences how detailed the local features are that are collected. Testing shows that a patch size of 8×8 times is a suitable balance between getting rid of noise and keeping the structure. For clustering, the first number of clusters K is found by dividing the total number of patches N by the average intended cluster size M, which is usually between 60 and 100. This relationship is defined as:

$$K = \left\lfloor \frac{N}{M} \right\rfloor \tag{15}$$

To refine the initial clustering results, clusters are merged based on their centroid similarity, using a noise-adaptive threshold ξ. This threshold is derived from the estimated noise variance σ^2, and is given by:

$$\xi = 16\,\sigma^2 \tag{16}$$

If a cluster exceeds the maximum allowed size L_T, typically set to 150, the threshold is scaled down using a factor $\rho = 0.7$, leading to an adjusted threshold:

$$\xi' = \rho . \xi \tag{17}$$

The Marchenko–Pastur law which is a thresholding criterion, determines which main components are chosen in PCA. Only components with eigenvalues that are higher than a scaled version of the noise threshold λ_+ are kept. The scaling multiplier μ is set between 1.2 and 1.5, and 1.3 gives consistent results in all studies. For suboptimal Wiener filtering, the attenuation factor ααα is changed dependent on how much noise is expected. A higher number such as like α = 0.9 is utilized to keep finer features in lower noise situations. On the other hand, a more aggressive filtering factor like α = 0.7 is better for greater noise levels. Finally, the LPA-ICI method (Luisier et al., 2011) is used to adaptively estimate the local variance needed for Wiener filtering. This method changes the filtering window based on local confidence intervals, thus there is no need for manual tuning and it works well in different spaces. These parameter selections which were tested on a variety of medical images lead to a strong and generalizable denoising framework that works well on a wide range of noise levels and anatomical parts.

Algorithm Summary

To improve reproducibility, we summarize the complete workflow of the proposed method using pseudocode, covering the key steps: Noise estimation, patch extraction, adaptive clustering, PCA-based thresholding, and reconstruction.

Algorithm 1: Medical Image Denoising via Cluster-wise PCA and Iterative Mean Filtering

```
Input: Noisy image I (of size M × N), patch size s, stride t, noise variance σ²
Output: Denoised image $I_{\text{denoised}}$
1. Estimate the noise standard deviation ($\sigma$) from the noisy image I
   using the Marchenko–Pastur (MP) law.
2. Apply iterative mean filtering on I to obtain a pre-denoised image $I_{\text{pre}}$.
3. Extract overlapping patches of size s × s from $I_{\text{pre}}$ with stride $t$:
   - Let $N_{\text{patches}}$ be the total number of extracted patches.
   - Vectorize each patch into a 1D array.
   - Store patches in matrix P ∈ ℝ^ {$N_{\text{patches}}$ × s²}.
4. Perform over-clustering using K-means++ algorithm:
   a. Set desired average cluster size M (e.g., 60–100).
   b. Compute number of clusters K = floor ($N_{\text{patches}}$/ M).
   c. Apply K-means++ to cluster patches into K groups {C₁, C₂, ..., $C_K$}.
5. Merge clusters based on similarity:
   For each pair of clusters ($C_i$, $C_j$):
   - Compute Euclidean distance between centroids.
   - If distance < ξ = $16\sigma^2$, merge $C_i$ and $C_j$.
   - If cluster size > threshold $L_T$ (e.g., 150), set ξ' = ρ·ξ (ρ = 0.7) for merging.
6. For each final cluster $C_j$:
   a. Compute the mean patch $\mu_j$.
   b. Center all patches by subtracting $\mu_j$.
   c. Apply PCA to the centered data matrix:
   - Compute eigenvalues {λ₁, λ₂, ..., $\lambda_d$} and eigenvectors {v₁, v₂, ..., $v_d$}.
   d. Select principal components where $\lambda_i > \mu \cdot \lambda_+$
   - $\lambda_+$ is the MP-based eigenvalue noise threshold.
   - μ is a scale factor (e.g., 1.2 to 1.5).
   e. Project patches into PCA subspace using retained components.
   f. Estimate local variance of each coefficient using LPA-ICI method.
   g. Apply suboptimal Wiener filtering:
   - For each coefficient $y_i$, compute:
   $\hat{y}_i = \alpha \cdot (\sigma^2_{\text{signal}} / (\sigma^2_{\text{signal}} + \sigma^2_{\text{noise}})) \cdot y_i$
   - α ∈ [0.7, 0.9] depending on noise level.
   h. Reconstruct denoised patches using inverse PCA transform.
```

```
    i. Add back the mean $\mu_j$ to each reconstructed patch.
7. Aggregate all denoised patches to reconstruct the final image:
    - For each pixel, compute the weighted average of all overlapping patches.
    - Normalize the aggregated pixel values by the sum of weights.
8. Return $I_{\text{denoised}}$ as the final denoised image.
```

Experimental Details

This section evaluates the performance of our denoising method using a series of experiments conducted on MRI and CT datasets. The dataset used for these experiments is open source and publicly available (The Whole Brain Atlas and Dataset for Fusion, Blending, Denoising and Enhancement by Imaging Science). To validate our method, we compare its performance against well-known denoising techniques including FOTV (Diwakar et al., 2024), OD-CNN (Atal, 2023), NDiff (Kollem et al., 2023), SDPM (Shen et al., 2023), UDLF (Rai et al., 2021a), NGN (Fu et al., 2022), CNCL (Zeng et al., 2024), BP-MEC (Ji et al., 2020), FDCT (Anandan et al., 2020), BF-CNN (Elhoseny and Shankar, 2019), DEN (Liu et al., 2019), WT (Satapathy et al., 2019), DCT-ACO (Miri et al., 2018), NNE (Liu et al., 2018), SDL (Bai et al., 2018), TNN (Zeng et al., 2024), RED (Ferdi et al., 2024), AWS-DCNN (Annavarapu and Borra, 2024), FOTV (Diwakar et al., 2024) and AKF-NLM (Taassori and Vizvari, 2024). All methods are evaluated using their default parameters to ensure optimal performance and fair comparison. The evaluation includes three test images: Two MRI scans (MRI1 and MRI2) and one CT scan (The Whole Brain Atlas and Dataset for Fusion, Blending, Denoising and Enhancement by Imaging Science). These images are tested under different levels (σ = 10,20,30 and 40) to assess robustness under different conditions. All experiments were performed using Matlab platform on a system equipped with an (Intel Core i7, 2.5 GHz, 16 GB RAM). For performance evaluation, we employ seven widely standard image performance metrics: Peak-Signal-To-Noise-Ratio (PSNR), structural similarity (SSIM), feature similarity (FSIM), Entropy (Lepcha et al., 2024b), and three non-reference performance metrics such as BRISQUE, PIQE, and NIQE (Goyal et al., 2024). A higher value of PSNR, SSIM, FSIM, and Entropy represents better image quality performance, whereas lower BRISQUE, PIQE, and NIQE scores are desirable.

Qualitative and Quantitative Assessment of Detail Preservation

We analyse the visual performance and subsequently provide the quantitative analysis using several performance metrics. Figures 2-5 illustrates the denoising outcomes of MRI1 data. Gaussian noise is introduced to the Raw input (Figs. 2-5 (a)), and several techniques are employed for denoising. Figures 2-5 (b-c) illustrates that SDL and WT preserve the external details while the inside details are pixelated. In Figs. 2-5 (d-e), the DCT-ACO and NNE algorithms employed for image denoising preserve the outer margins while the inside features remain pixelated; yet, they yield superior results compared to SDL and WT images. Figures 2-5 (f-g) illustrates that BF-CNN and DEN effectively maintain edge integrity to a degree; however, the internal details remain ambiguous. Figures 2-5 (h, i, j) demonstrates that the CNCL, BP-MEC, and FDCT methodologies are also ineffective in preserving the edges of the images. In Fig. 2-5 (k, l, m) from the zoomed section, it is evident that NDiff, SDPM, and NGN successfully preserve the outer boundaries; nevertheless, the inner features of the image appear blurred. Figures 2-5 (n,o) illustrates that OD-CNN and DCNN fail to preserve edge sharpness, resulting in distortion.

Figures 2-5 (p, q, r) illustrate that the denoised outputs generated by AWS-DCNN, FOTV, and AKF-NLM exhibit significant image blurring along with an increase in edge width. These effects contribute to the degradation of structural integrity and detail preservation, making the images appear less sharp and more diffused. In contrast, Figures 2-5 (s, t) show that TNN and RED produce visually appealing results that closely match the ground truth image. These methods are effective in preserving both internal textures and external boundaries, maintaining high structural fidelity. The outcome of our proposed approach as shown in Figure 2-5 (u), displays a notable improvement over all compared techniques delivering high-quality denoising results while retaining rich details. To further substantiate the qualitative findings, Table 1 provides a quantitative assessment using multiple evaluation metrics. Our method achieves higher values in PSNR, SSIM, Entropy, and FSIM, indicating better noise suppression and detail retention. Additionally, the lower NIQE, PIQE, and BRISQUE scores highlight the enhanced perceptual quality and natural appearance of the denoised images. These findings affirm the effectiveness and robustness of the proposed approach across varying noise levels in preserving structural and perceptual features while delivering superior denoising performance. Figures 6-9 showcase the qualitative performance of various denoising methods on the MRI2 dataset. A careful inspection of Figures 6-9 (b, c) reveals that the images generated using SDL and WT techniques suffer from significant pixelation, which negatively impacts their visual quality. Similarly, Figures 6-9 (d, e) displays the results produced by the DCT-ACO and NNE methods, where the borders of anatomical structures appear unclear and pixelated thus, indicating suboptimal feature preservation. In Figures 6-9 (f, g), the outputs of the BF-CNN and DEN methods are presented. Although these methods incorporate convolutional strategies to enhance image features, they still fail to capture and retain the richness of structural details present in the original scans.

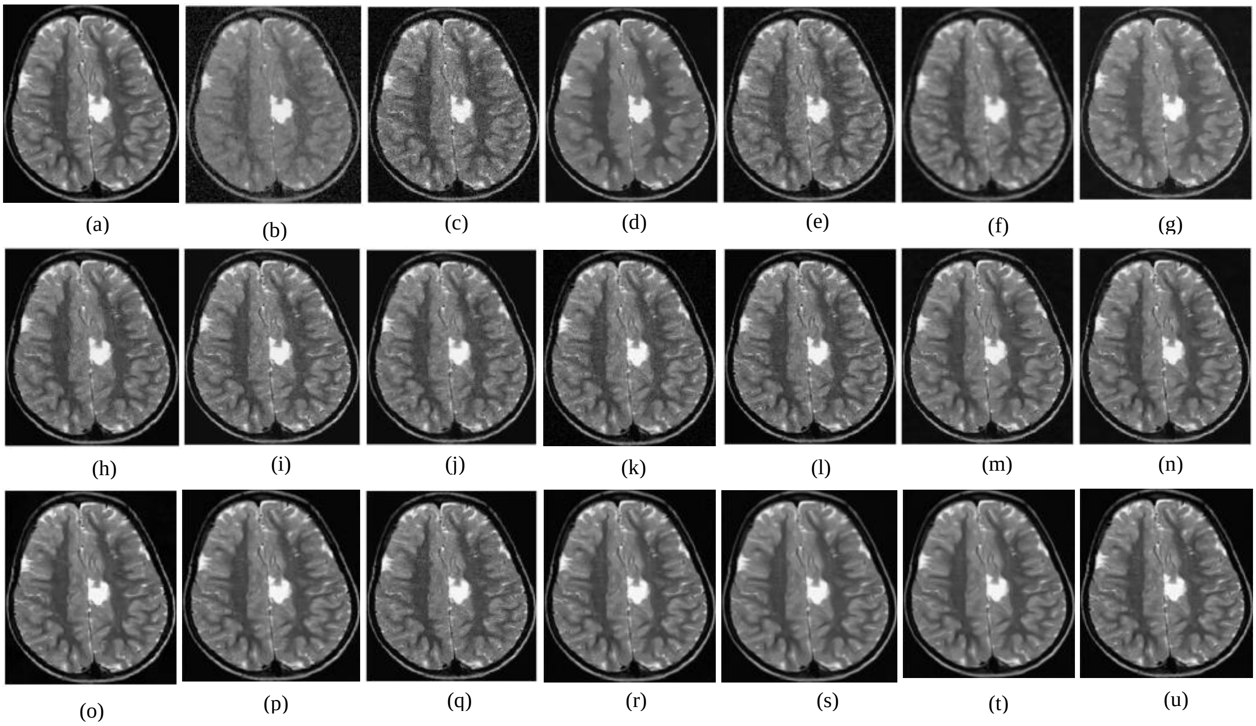

Fig. 2: Denoising results for the MRI1 image corrupted with Gaussian noise (σ ≈ 10). (a) Raw input, (b) WT, (c) SDL, (d) NNE, (e) DCT-ACO, (f) DEN, (g) BF-CNN, (h) FDCT, (i) BP-MEC, (j) CNCL, (k) UDLF, (l) NGN, (m) SDPM, (n) NDiff, (o) OD-CNN, (p) AKF-NLM, (q) FOTV, (r) AWS-DCNN, (s) RED, (t) TNN, and (u) Proposed method

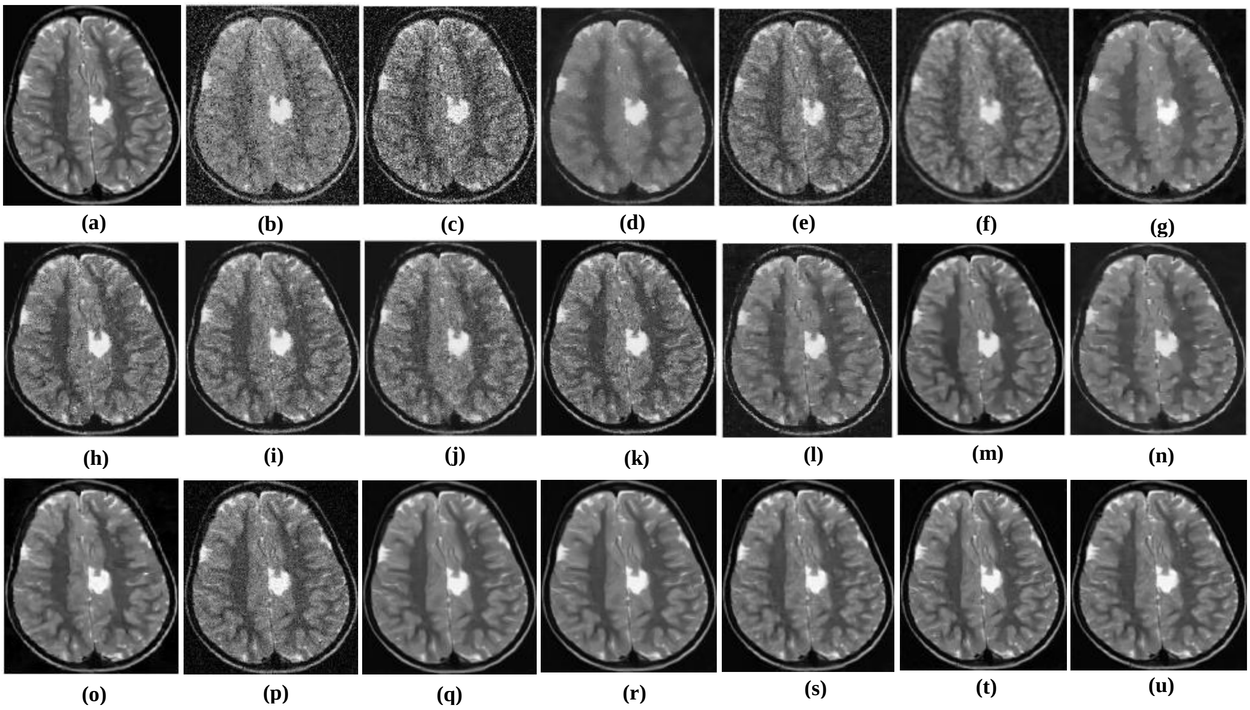

Fig. 3: Denoising results for the MRI1 image corrupted with Gaussian noise (σ ≈ 20). (a) Raw input, (b) WT, (c) SDL, (d) NNE, (e) DCT-ACO, (f) DEN, (g) BF-CNN, (h) FDCT, (i) BP-MEC, (j) CNCL, (k) UDLF, (l) NGN, (m) SDPM, (n) NDiff, (o) OD-CNN, (p) AKF-NLM, (q) FOTV, (r) AWS-DCNN, (s) RED, (t) TNN, and (u) Proposed method

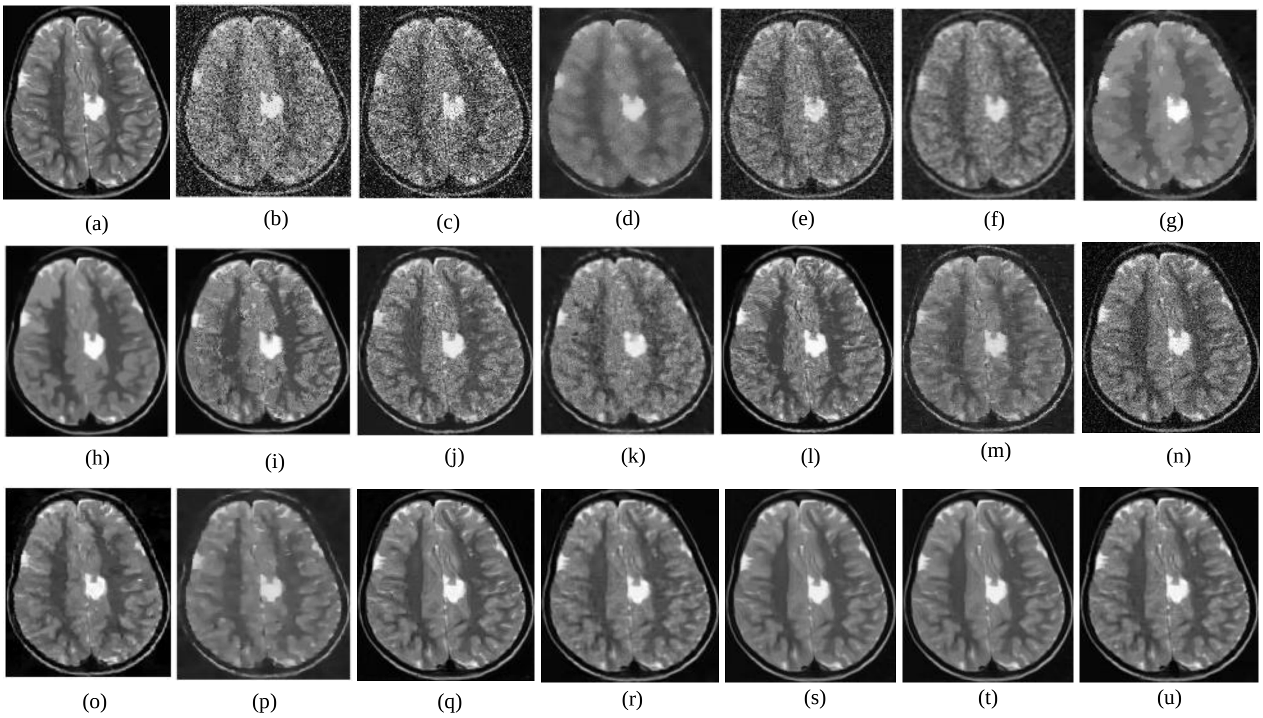

Fig. 4: Denoising results for the MRI1 image corrupted with Gaussian noise (σ ≈ 30). (a) Raw input, (b) WT, (c) SDL, (d) NNE, (e) DCT-ACO, (f) DEN, (g) BF-CNN, (h) FDCT, (i) BP-MEC, (j) CNCL, (k) UDLF, (l) NGN, (m) SDPM, (n) NDiff, (o) OD-CNN, (p) AKF-NLM, (q) FOTV, (r) AWS-DCNN, (s) RED, (t) TNN, and (u) Proposed method

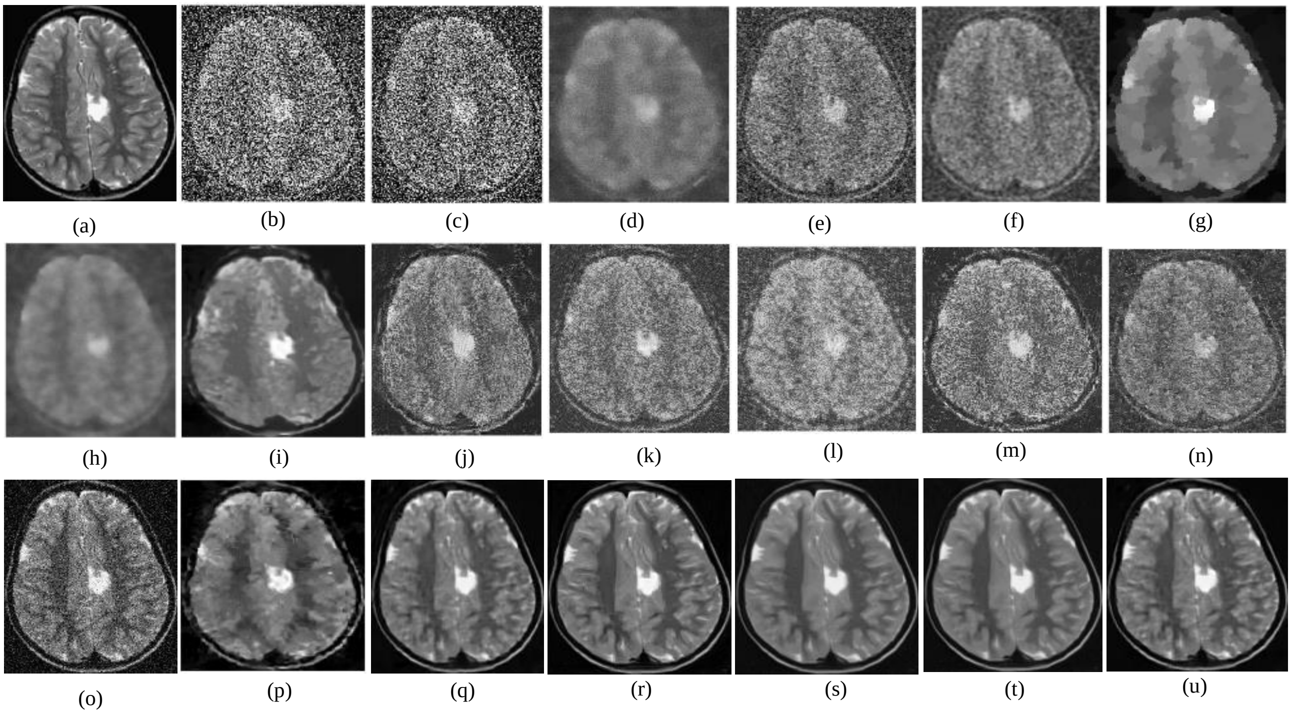

Fig. 5: Denoising results for the MRI1 image corrupted with Gaussian noise (σ ≈ 40). (a) Raw input, (b) WT, (c) SDL, (d) NNE, (e) DCT-ACO, (f) DEN, (g) BF-CNN, (h) FDCT, (i) BP-MEC, (j) CNCL, (k) UDLF, (l) NGN, (m) SDPM, (n) NDiff, (o) OD-CNN, (p) AKF-NLM, (q) FOTV, (r) AWS-DCNN, (s) RED, (t) TNN, and (u) Proposed method

Table 1: Quantitative evaluation of denoising performance for the MRI1 dataset under varying noise levels (σ = 10, 20, 30, 40)

Method	σ	Entropy↑	PSNR↑	SSIM↑	FSIM↑	BRISQUE↓	NIQE↓	PIQE↓
Ours	10	6.81	35.88	0.95	0.96	40.33	4.49	50.30
	20	6.82	32.11	0.90	0.93	42.21	4.57	52.48
	30	6.73	30.01	0.86	0.91	38.22	4.44	53.58
	40	6.94	28.57	0.82	0.89	41.12	4.81	57.11
TNN	10	6.58	33.67	0.93	0.94	42.19	5.01	49.81
	20	6.63	31.78	0.91	0.90	43.33	4.63	53.45
	30	6.71	29.93	0.87	0.89	40.99	4.66	54.44
	40	6.66	28.33	0.82	0.88	44.22	4.73	58.93
RED	10	6.60	32.65	0.94	0.92	44.32	5.10	50.55
	20	6.66	30.44	0.89	0.88	44.45	5.34	55.21
	30	6.32	28.86	0.86	0.87	41.01	5.12	56.77
	40	6.34	27.99	0.81	0.86	42.46	4.81	57.72
AWS-DCNN	10	6.62	31.73	0.93	0.92	45.62	4.96	49.97
	20	6.33	29.67	0.88	0.86	46.78	5.22	56.54
	30	6.23	27.45	0.84	0.85	43.32	5.20	57.67
	40	6.11	26.77	0.79	0.83	46.78	5.23	58.93
FOTV	10	6.56	32.91	0.94	0.88	44.72	5.23	51.77
	20	5.97	28.66	0.86	0.86	47.11	5.34	57.33
	30	6.01	26.99	0.83	0.82	44.27	5.27	58.62
	40	5.95	25.99	0.77	0.81	45.77	5.53	57.89
AKF-NLM	10	6.43	31.99	0.93	0.87	45.09	5.34	52.87
	20	5.99	27.67	0.84	0.85	46.87	5.24	58.93
	30	5.87	25.45	0.81	0.80	46.56	5.55	59.43
	40	5.67	25.80	0.78	0.79	46.77	5.67	58.64
OD-CNN	10	6.45	32.29	0.93	0.88	43.82	5.88	60.41
	20	6.08	27.97	0.88	0.87	48.14	5.42	51.24
	30	6.10	26.37	0.83	0.86	47.90	5.92	43.01
	40	6.16	23.87	0.81	0.85	46.12	5.49	37.10
NDiff	10	5.78	30.76	0.89	0.88	44.38	5.22	51.83
	20	5.87	24.99	0.82	0.86	46.38	5.56	43.28
	30	5.86	22.67	0.83	0.89	45.24	5.44	43.59
	40	5.47	26.86	0.80	0.83	47.46	5.67	69.45
SDPM	10	5.42	28.75	0.87	0.84	47.12	5.36	57.85
	20	5.69	25.35	0.83	0.86	49.66	4.90	74.79
	30	5.68	22.86	0.83	0.83	48.16	5.29	77.64
	40	5.98	20.31	0.82	0.81	48.46	5.38	84.50
NGN	10	5.64	27.64	0.83	0.86	51.19	5.39	42.50
	20	5.84	25.52	0.79	0.83	55.29	5.65	59.84
	30	5.70	24.89	0.77	0.79	48.81	4.83	53.07
	40	5.96	21.94	0.79	0.79	53.70	5.84	73.97
UDLF	10	5.20	26.25	0.85	0.81	50.79	5.49	24.58
	20	5.42	23.21	0.80	0.78	51.56	5.67	19.98
	30	5.45	21.08	0.81	0.79	52.53	5.57	19.55
	40	5.32	15.86	0.78	0.73	53.43	6.01	20.89
CNCL	10	6.04	26.39	0.80	0.83	53.63	5.93	70.09
	20	5.14	23.70	0.79	0.75	50.77	5.11	61.51
	30	5.24	22.27	0.76	0.77	49.31	4.92	56.82
	40	5.87	17.39	0.77	0.73	54.25	6.48	39.88
BP-MEC	10	5.69	27.04	0.79	0.82	52.67	5.83	45.29
	20	5.91	23.17	0.78	0.78	58.86	6.53	58.31
	30	5.97	21.21	0.77	0.77	53.18	6.07	62.54
	40	5.71	16.91	0.78	0.78	53.46	5.56	77.20
FDCT	10	5.89	26.02	0.76	0.79	50.59	6.16	32.45
	20	5.05	22.69	0.75	0.76	54.79	5.86	31.28
	30	5.10	22.31	0.71	0.73	57.77	6.13	37.72
	40	5.14	16.13	0.70	0.76	53.85	6.60	63.32
BF-CNN	10	5.079	24.08	0.71	0.82	54.03	6.02	40.79
	20	5.515	20.20	0.72	0.74	57.17	6.41	71.15
	30	5.62	19.54	0.69	0.77	52.48	5.88	76.03

	40	5.737	18.10	0.70	0.73	53.45	6.15	83.77
DEN	10	5.60	25.30	0.76	0.80	57.99	6.25	39.65
	20	5.80	21.58	0.71	0.78	56.48	6.12	52.19
	30	5.84	20.97	0.70	0.73	55.66	6.76	51.58
	40	5.02	15.22	0.69	0.71	53.49	7.20	77.10
DCT-ACO	10	4.61	24.93	0.71	0.81	54.73	6.68	42.56
	20	4.87	21.60	0.66	0.73	58.42	6.62	53.45
	30	4.82	21.73	0.69	0.76	56.32	6.56	60.74
	40	4.93	15.53	0.68	0.73	53.46	6.90	79.95
NNE	10	4.67	25.66	0.66	0.75	55.55	7.20	46.22
	20	4.96	21.99	0.63	0.73	62.89	7.07	57.53
	30	3.77	22.57	0.65	0.76	57.18	7.01	56.74
	40	3.91	16.18	0.63	0.72	53.47	6.90	74.97
SDL	10	5.44	25.08	0.69	0.80	58.85	6.88	47.35
	20	4.44	22.48	0.64	0.73	58.57	6.75	47.35
	30	4.44	23.09	0.63	0.69	58.85	6.98	47.35
	40	4.44	16.79	0.61	0.67	58.58	6.88	47.35
WT	10	3.68	24.06	0.63	0.73	63.46	6.58	42.08
	20	3.64	23.06	0.61	0.71	62.22	7.39	56.51
	30	3.59	21.06	0.62	0.72	58.80	6.95	69.42
	40	3.57	13.05	0.60	0.67	62.43	7.13	77.03

Metrics include Entropy, Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index (SSIM), Feature Similarity Index (FSIM), Blind/Reference less Image Spatial Quality Evaluator (BRISQUE), Natural Image Quality Evaluator (NIQE), and Perception-Based Image Quality Evaluator (PIQE). Higher PSNR, SSIM, FSIM, and Entropy indicate better detail preservation; lower BRISQUE, NIQE, and PIQE reflect superior perceptual quality

Further observations can be made from Figures 6-9 (h, i, j), where denoising is performed using CNCL, BP-MEC, and FDCT techniques. While these methods are intended to suppress noise, their application results in softer and less defined edges, which consequently, reduces the overall clarity and visual appeal of the images. In contrast, Figures 6-9 (k, l, m) illustrate the results obtained using NDiff, SDPM, and NGN approaches. These techniques perform better in maintaining the outer contours of anatomical.

Structures; however, the inner regions still appear somewhat blurry, suggesting an incomplete recovery of fine-grained features. Figures 6-9 (n, o) present the denoised images generated by OD-CNN and DCNN-based methods. Despite the use of deep convolutional networks, the output images continue to exhibit residual noise and edge degradation, implying that these methods may not be optimal for scenarios requiring high-detail preservation. Likewise, Figures 6-9 (p, q, r) demonstrate the performance of AWS-DCNN, FOTV, and AKF-NLM techniques. These approaches yield visually blurred outcomes, accompanied by a lack of structural sharpness and reduced overall image fidelity. On the other hand, Figures 6-9 (s, t) and Figures 6-9 (u) highlight the effectiveness of the TNN, RED, and the proposed methodology. These echniques outperform the previously discussed method is particularly noteworthy and set it apart from traditional and even some deep learning-based approaches. To quantitatively assess the performance of all the evaluated techniques, Table 2 presents the average image quality metrics obtained for MRI2 images. It is evident from the table that, our approach achieves the highest scores on PSNR, SSIM compared to other methods by more effectively reducing noise and retaining meaningful anatomical details. The visual clarity and edge definition achieved by the proposed Entropy, and FSIM, indicating superior image reconstruction capability and structural similarity preservation. Furthermore, the proposed approach records lower values for NIQE, PIQE, and BRISQUE metrics associated with perceptual image quality thus further validating its effectiveness in enhancing visual quality. Figures 10-13 present the visual outputs of denoised CT image reconstructed using several denoising algorithms use at different noise levels. In Figures 10-13 (b, c) illustrate the denoising results obtained using the SDL and WT methods. These approaches are robust in preserving the overall structural integrity of an image. However, a closer inspection reveals the emergence of grainy textures, particularly along the image edges. This graininess causes the boundaries to appear less distinct, reducing the overall clarity and definition of fine image features. In Figures 10–13 (d, e), the results of DCT-ACO and NNE methods are presented. These approaches manage to reduce noise to a moderate extent, providing a visibly cleaner image compared to the raw input. Nonetheless, there is a slight positional shift in the image structures when compares to the original. 13 (k, l, m) shows the outputs from NDiff, SDPM, and NGN, respectively. These methods succeed in reducing noise across the image and enhance the visual quality to a certain degree. However, there is room for improvement in capturing the finer details. The methods tend to smooth out textures, resulting in a loss of high-frequency information that is often critical for distinguishing subtle features.

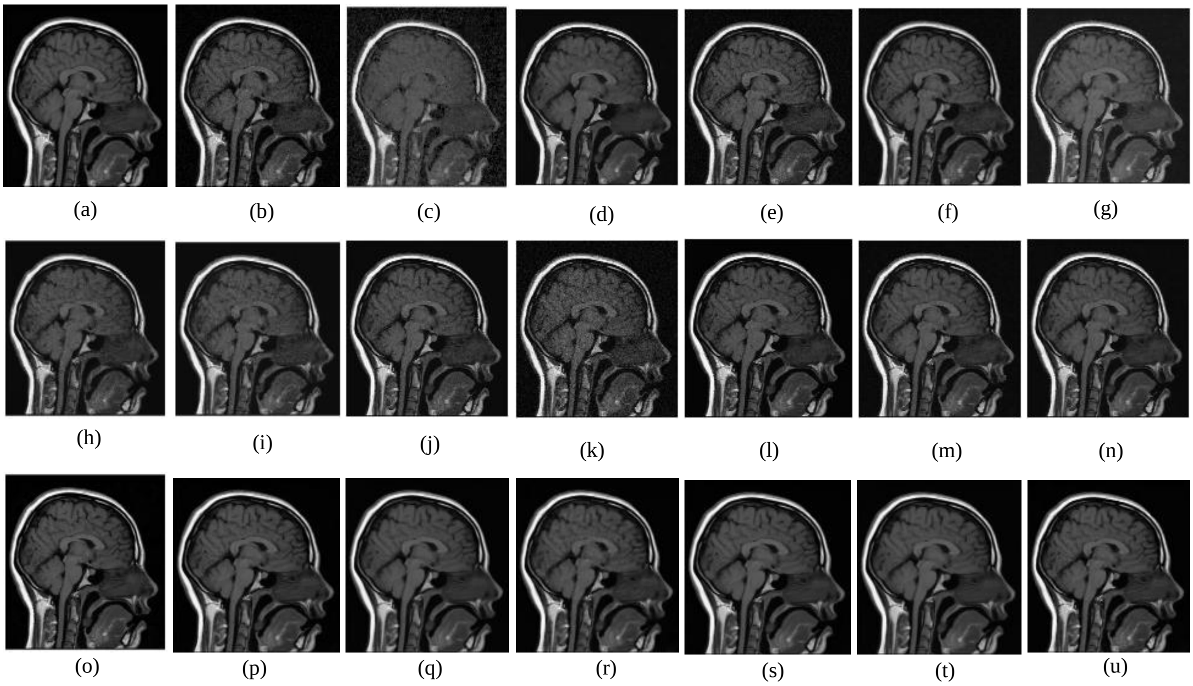

Fig. 6: Denoising results for the MRI2 image corrupted with Gaussian noise (σ ≈ 10). (a) Raw input, (b) WT, (c) SDL, (d) NNE, (e) DCT-ACO, (f) DEN, (g) BF-CNN, (h) FDCT, (i) BP-MEC, (j) CNCL, (k) UDLF, (l) NGN, (m) SDPM, (n) NDiff, (o) OD-CNN, (p) AKF-NLM, (q) FOTV, (r) AWS-DCNN, (s) RED, (t) TNN, and (u) Proposed method

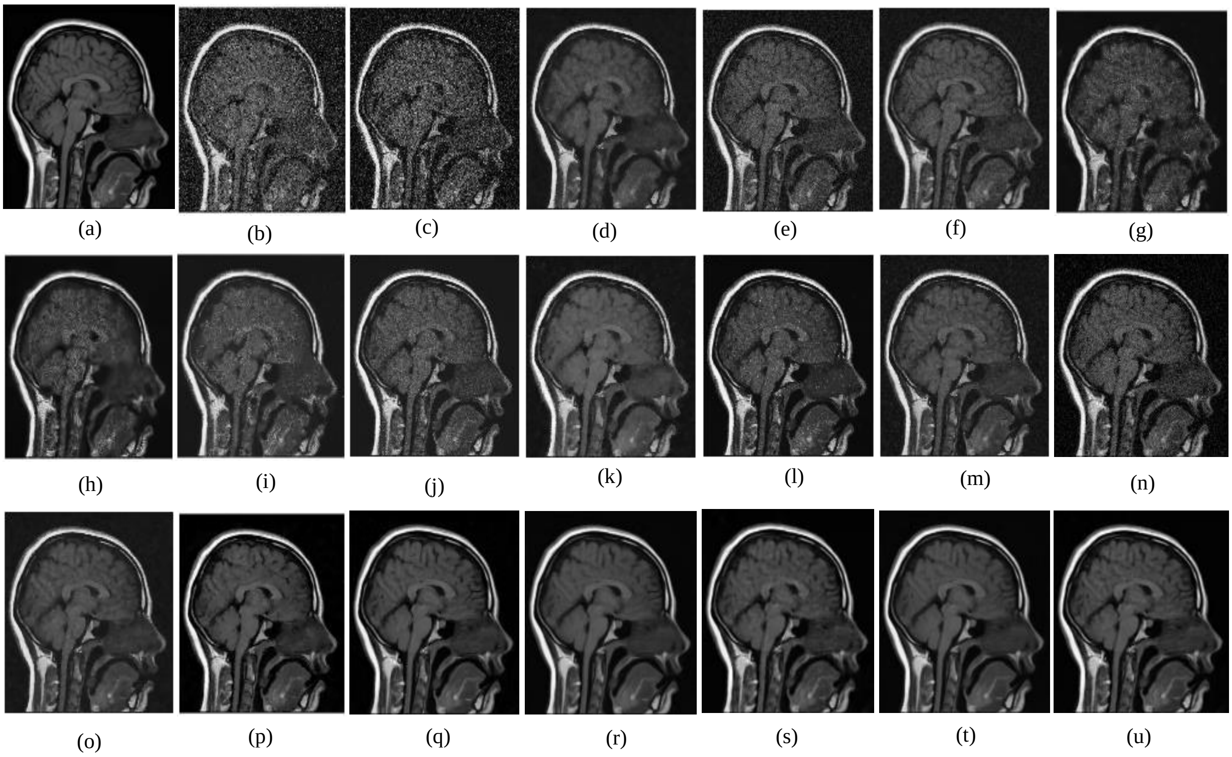

Fig. 7: Denoising results for the MRI2 image corrupted with Gaussian noise (σ ≈ 20). (a) Raw input, (b) WT, (c) SDL, (d) NNE, (e) DCT-ACO, (f) DEN, (g) BF-CNN, (h) FDCT, (i) BP-MEC, (j) CNCL, (k) UDLF, (l) NGN, (m) SDPM, (n) NDiff, (o) OD-CNN, (p) AKF-NLM, (q) FOTV, (r) AWS-DCNN, (s) RED, (t) TNN, and (u) Proposed method

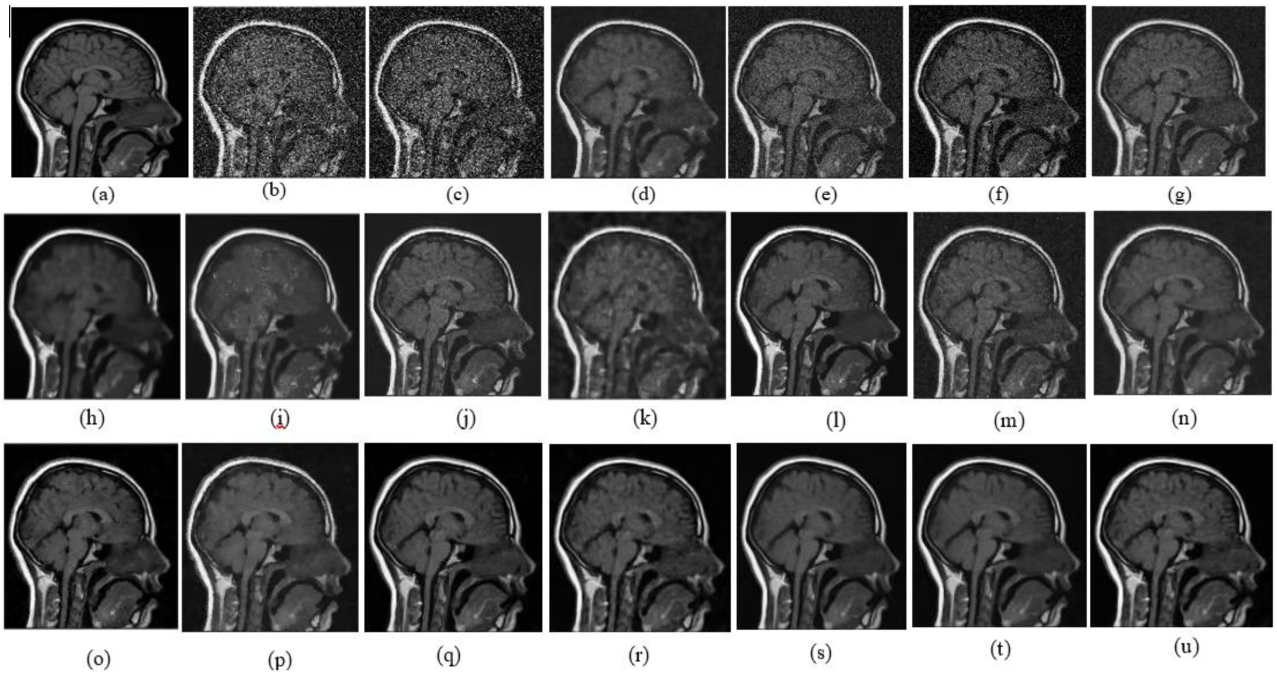

Fig. 8: Denoising results for the MRI2 image corrupted with Gaussian noise (σ ≈ 30). (a) Raw input, (b) WT, (c) SDL, (d) NNE, (e) DCT-ACO, (f) DEN, (g) BF-CNN, (h) FDCT, (i) BP-MEC, (j) CNCL, (k) UDLF, (l) NGN, (m) SDPM, (n) NDiff, (o) OD-CNN, (p) AKF-NLM, (q) FOTV, (r) AWS-DCNN, (s) RED, (t) TNN, and (u) Proposed method

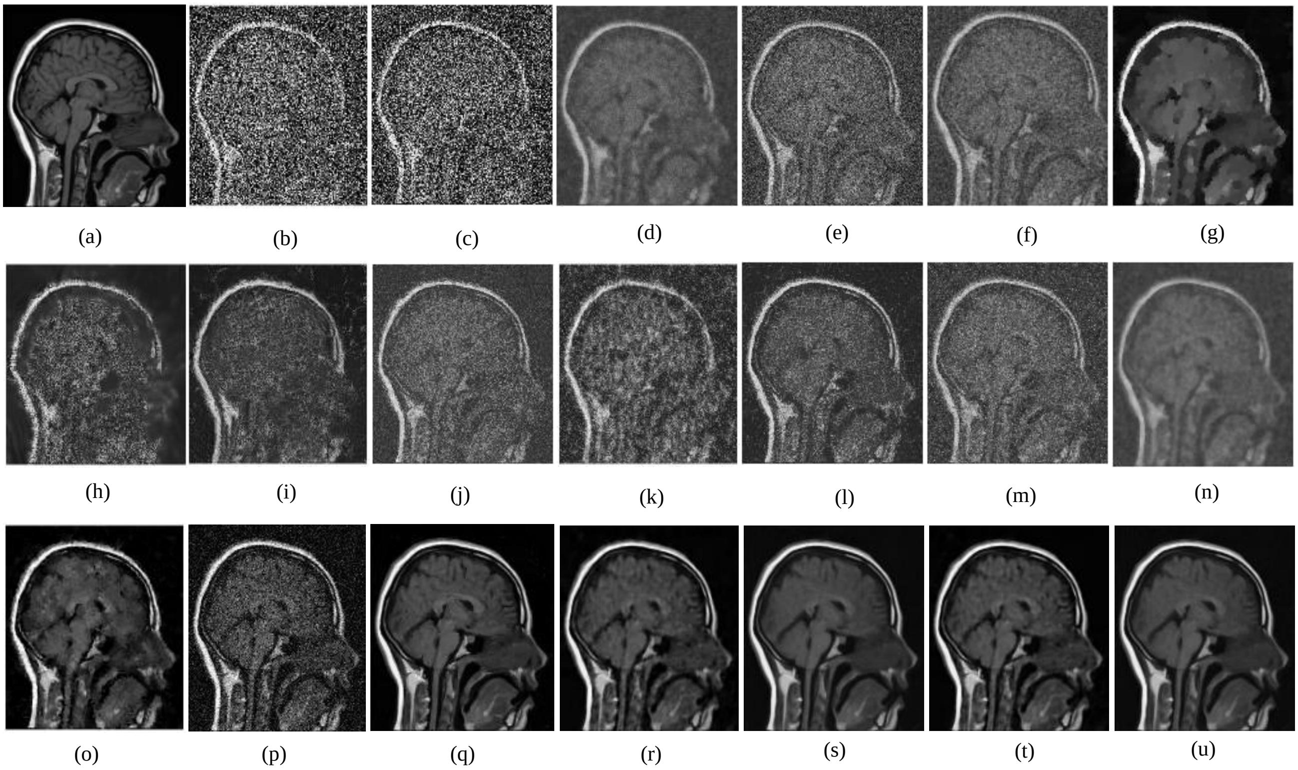

Fig. 9: Denoising results for the MRI2 image corrupted with Gaussian noise (σ ≈ 40). (a) Raw input, (b) WT, (c) SDL, (d) NNE, (e) DCT-ACO, (f) DEN, (g) BF-CNN, (h) FDCT, (i) BP-MEC, (j) CNCL, (k) UDLF, (l) NGN, (m) SDPM, (n) NDiff, (o) OD-CNN, (p) AKF-NLM, (q) FOTV, (r) AWS-DCNN, (s) RED, (t) TNN, and (u) Proposed method

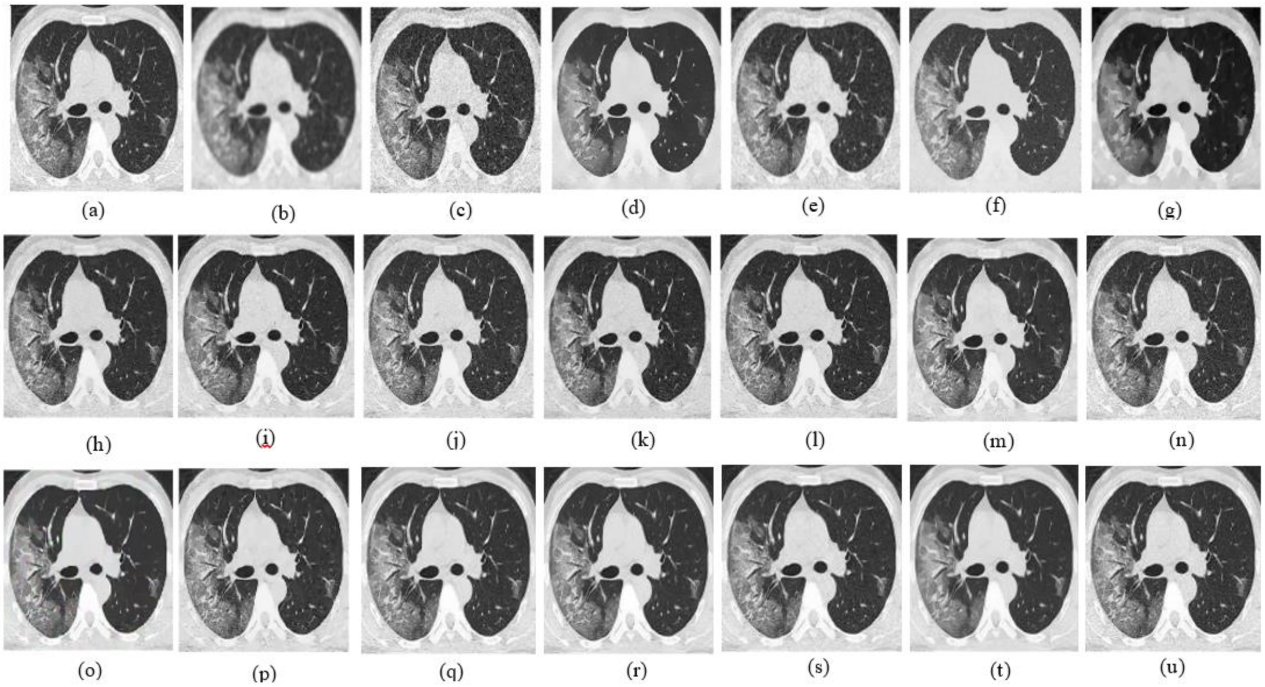

Fig. 10: Denoising results for the CT image corrupted with Gaussian noise (σ ≈ 10). (a) Raw input, (b) WT, (c) SDL, (d) NNE, (e) DCT-ACO, (f) DEN, (g) BF-CNN, (h) FDCT, (i) BP-MEC, (j) CNCL, (k) UDLF, (l) NGN, (m) SDPM, (n) NDiff, (o) OD-CNN, (p) AKF-NLM, (q) FOTV, (r) AWS-DCNN, (s) RED, (t) TNN, and (u) Proposed method

Figures 10–13 (n, o) depict the denoised results produced by OD-CNN and DCNN. Despite employing deep learning-based techniques, these methods still exhibit residual noise, especially in complex or high-frequency regions of the image. This residual noise slightly diminishes the perceptual quality of the results. In figures 10–13 (p, q, r), which correspond to AWS-DCNN, FOTV, and AKF-NLM demonstrate a blurring effect that negatively impacts visual fidelity. Although these methods aim to suppress noise, they end up smoothing the image excessively, leading to the loss of meaningful structural information and lowering the image's perceptual sharpness. In Figures 10–13 (s, t) shows the results from the TNN and RED approaches which are among the proposed methods in this study. These approaches offer notable improvements. The reconstructed images are visually similar to the input, and the noise is significantly suppressed. The clarity and preservation of structural features indicate the efficiency of the proposed method in balancing noise removal and detail maintenance. Figures 10–13 (u) again emphasizes the efficiency of our method, where the noise is considerably reduced without major compromise on structural or textural details. In addition to visual analysis, Table 3 presents a quantitative evaluation of all approaches under various noise conditions. The proposed methods outperform others across multiple metrics. Higher values of PSNR, Entropy, FSIM, and SSIM demonstrate their superior image fidelity and feature preservation, while lower values of no-reference quality metrics such as NIQE, PIQE, and BRISQUE confirm enhanced perceptual quality. In summary, the combined observations from Figures 2-13 and Tables 1-3 clearly indicate that the proposed image denoising approaches provide better performance both quantitatively and qualitatively compared to existing techniques.

This study introduces a unified, data-adaptive framework that tightly couples unsupervised clustering, principal component analysis, and iterative mean filtering. While each component has been studied independently, their strategic combination in a feedback-optimized architecture yields significant improvement in structural fidelity and denoising accuracy. This integrated pipeline bridges the gap between classic linear filtering and adaptive patch grouping, showing strong potential for clinical imaging applications where detail preservation is crucial.

Computational Complexity and Runtime Performance

The computational complexity and runtime performance of the proposed denoising method are critical aspects, particularly for medical imaging applications requiring efficient processing. As illustrated in Figure 14, the runtime of the proposed method is approximately 3.9 seconds for a 512×512 medical image, evaluated on a standard CPU setup (Intel Core i7, 2.5 GHz, 16 GB RAM).

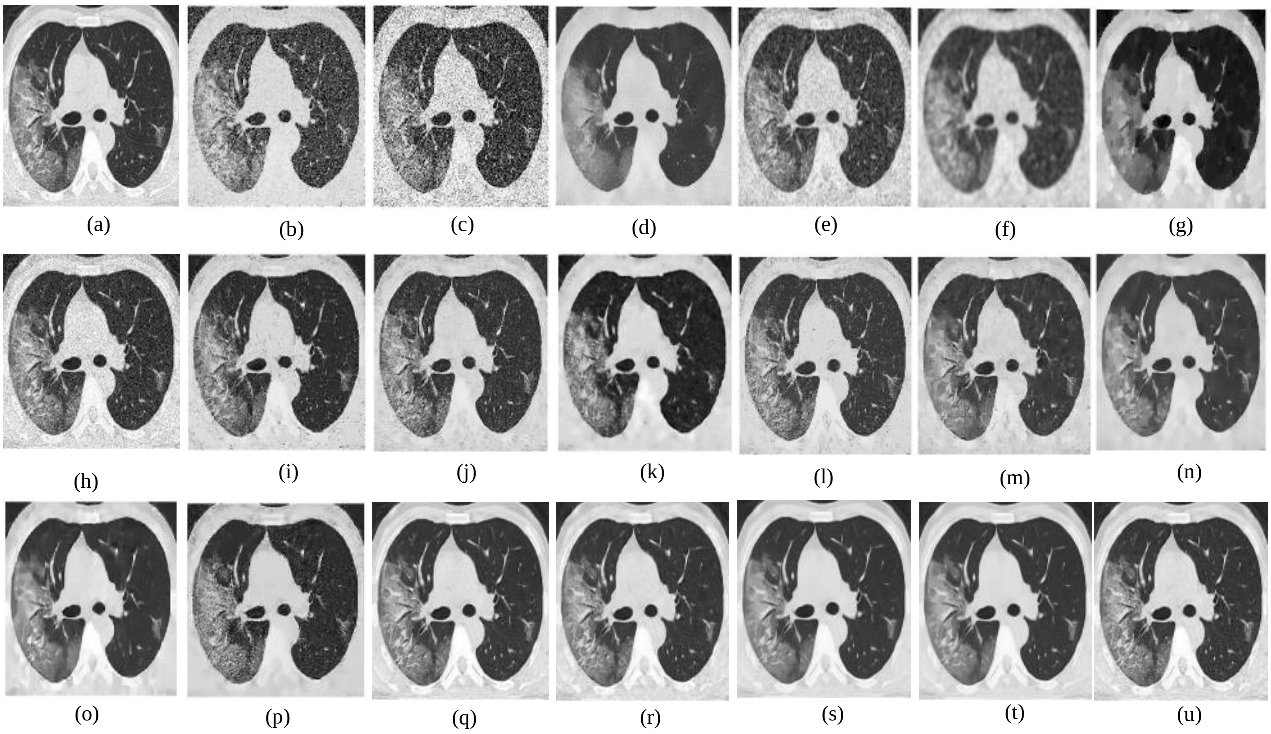

Fig. 11: Denoising results for the CT image corrupted with Gaussian noise (σ ≈ 20). (a) Raw input, (b) WT, (c) SDL, (d) NNE, (e) DCT-ACO, (f) DEN, (g) BF-CNN, (h) FDCT, (i) BP-MEC, (j) CNCL, (k) UDLF, (l) NGN, (m) SDPM, (n) NDiff, (o) OD-CNN, (p) AKF-NLM, (q) FOTV, (r) AWS-DCNN, (s) RED, (t) TNN, and (u) Proposed method

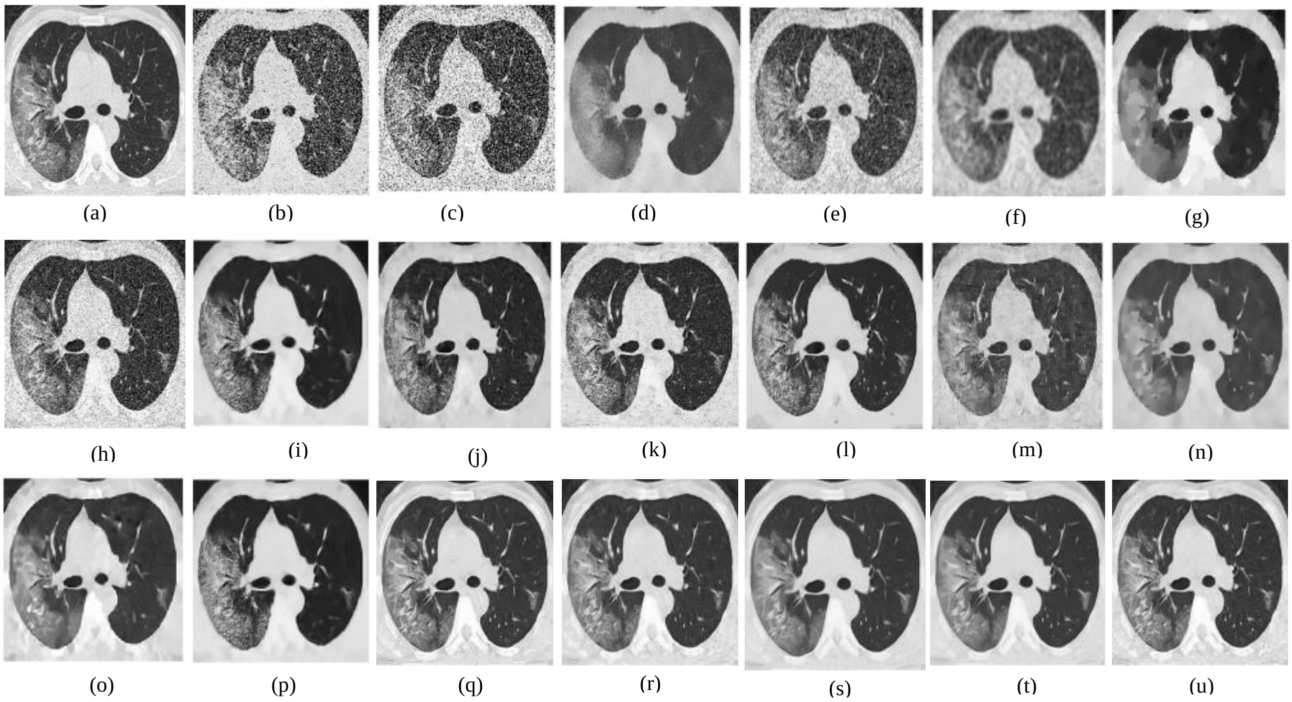

Fig. 12: Denoising results for the CT image corrupted with Gaussian noise (σ ≈ 30). (a) Raw input, (b) WT, (c) SDL, (d) NNE, (e) DCT-ACO, (f) DEN, (g) BF-CNN, (h) FDCT, (i) BP-MEC, (j) CNCL, (k) UDLF, (l) NGN, (m) SDPM, (n) NDiff, (o) OD-CNN, (p) AKF-NLM, (q) FOTV, (r) AWS-DCNN, (s) RED, (t) TNN, and (u) Proposed method

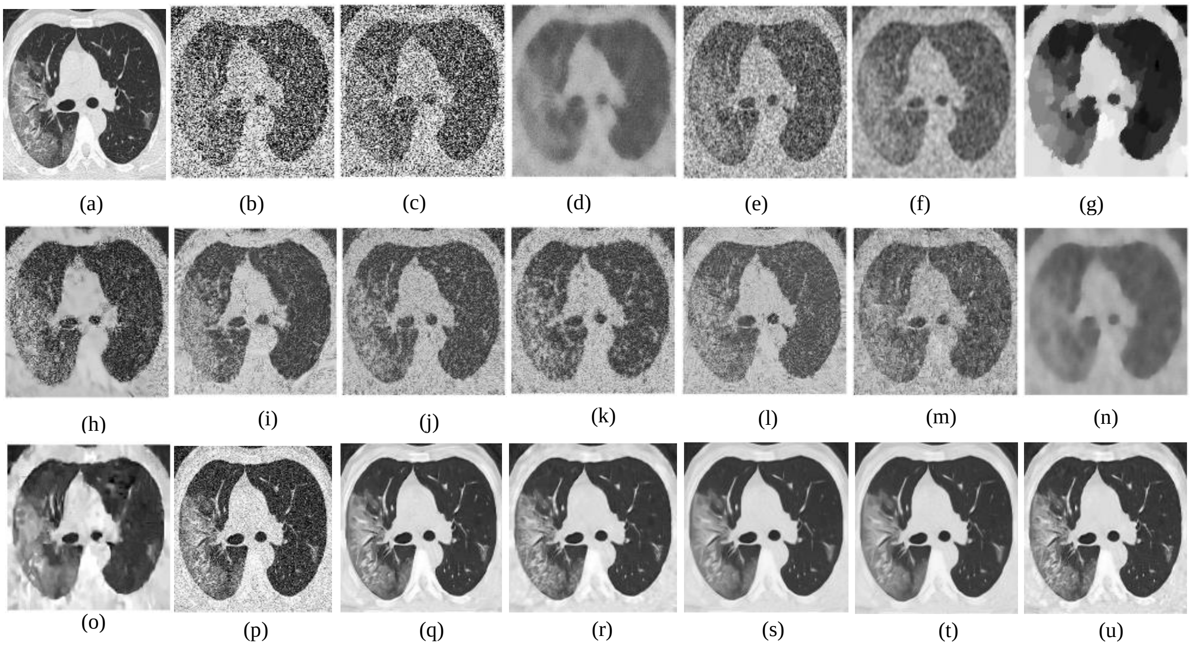

Fig. 13: Denoising results for the CT image corrupted with Gaussian noise (σ ≈ 40). (a) Raw input, (b) WT, (c) SDL, (d) NNE, (e) DCT-ACO, (f) DEN, (g) BF-CNN, (h) FDCT, (i) BP-MEC, (j) CNCL, (k) UDLF, (l) NGN, (m) SDPM, (n) NDiff, (o) OD-CNN, (p) AKF-NLM, (q) FOTV, (r) AWS-DCNN, (s) RED, (t) TNN, and (u) Proposed method

Table 2: Quantitative evaluation of denoising performance for the MRI2 dataset under varying noise levels (σ = 10, 20, 30, 40)

Method	σ	Entropy↑	PSNR↑	SSIM↑	FSIM↑	BRISQUE↓	NIQE↓	PIQE↓
Ours	10	6.40	36.37	0.95	0.96	47.91	4.78	43.79
	20	6.46	32.70	0.91	0.93	46.88	4.61	46.98
	30	6.50	30.54	0.86	0.91	48.66	4.85	45.44
	40	6.59	28.64	0.80	0.88	42.04	4.35	49.89
TNN	10	6.34	34.37	0.93	0.94	48.78	4.96	44.62
	20	6.17	31.99	0.89	0.91	47.88	4.97	47.23
	30	6.38	28.99	0.83	0.88	47.85	5.23	47.03
	40	6.66	29.17	0.81	0.86	43.33	4.73	50.23
RED	10	6.45	33.23	0.92	0.93	49.57	5.13	45.44
	20	6.23	31.89	0.90	0.89	48.45	5.34	48.88
	30	6.17	27.88	0.79	0.86	48.66	5.27	48.33
	40	5.97	28.99	0.78	0.83	44.34	5.24	51.23
AWS-DCNN	10	6.23	33.19	0.89	0.92	50.50	5.23	51.99
	20	5.99	29.88	0.88	0.86	49.66	5.45	52.90
	30	5.92	27.51	0.78	0.84	49.24	5.34	51.42
	40	5.78	27.14	0.76	0.80	50.83	5.78	52.67
FOTV	10	6.24	33.67	0.90	0.93	51.99	5.45	53.98
	20	5.55	28.78	0.87	0.85	50.34	5.65	53.99
	30	5.67	26.67	0.86	0.83	50.56	5.56	54.99
	40	5.24	25.89	0.77	0.79	51.42	5.67	54.17
AKF-NLM	10	6.18	33.79	0.89	0.93	50.87	5.56	55.99
	20	5.46	27.89	0.85	0.83	51.76	5.89	55.67
	30	5.37	25.57	0.84	0.81	51.55	5.97	56.99
	40	5.11	24.99	0.76	0.78	52.47	5.89	55.90
OD-CNN	10	6.12	34.72	0.88	0.93	49.74	5.67	57.01
	20	6.01	31.10	0.86	0.92	49.53	5.59	59.32
	30	6.14	29.38	0.81	0.89	46.98	5.74	56.77
	40	6.13	27.18	0.78	0.87	47.52	5.97	58.78
NDiff	10	5.34	32.06	0.84	0.92	53.50	5.65	60.68
	20	5.43	26.08	0.80	0.87	55.02	5.30	63.42
	30	5.44	26.98	0.79	0.86	53.45	5.83	58.80
	40	5.19	28.22	0.80	0.88	53.46	5.71	60.93

SDPM	10	5.99	28.28	0.81	0.89	53.01	5.78	64.14
	20	6.22	26.45	0.77	0.87	48.43	5.88	64.74
	30	5.24	23.85	0.74	0.86	54.17	5.99	68.09
	40	5.82	23.65	0.81	0.85	45.46	6.14	65.00
NGN	10	6.15	26.98	0.76	0.88	54.74	6.21	59.31
	20	5.29	23.25	0.72	0.86	52.91	6.00	57.90
	30	6.06	23.06	0.75	0.86	62.16	5.96	60.30
	40	5.44	20.16	0.72	0.84	53.51	5.54	64.00
UDLF	10	5.81	30.11	0.85	0.86	50.93	5.79	65.64
	20	6.00	23.91	0.81	0.82	62.60	5.74	60.67
	30	6.06	21.83	0.76	0.81	63.19	6.10	58.74
	40	6.17	21.60	0.73	0.80	53.43	5.80	59.80
CNCL	10	5.58	29.98	0.84	0.78	49.36	5.84	62.39
	20	5.71	25.60	0.81	0. 77	50.44	6.28	65.95
	30	5.75	23.82	0.76	0.77	59.54	5.69	62.60
	40	5.62	18.56	0.73	0.76	54.46	6.50	64.01
BP-MEC	10	5.33	30.64	0.83	0.84	59.14	5.77	63.71
	20	5.49	24.94	0.79	0.82	63.42	5.99	62.22
	30	5.51	23.20	0.75	0.79	63.46	6.19	65.91
	40	5.40	18.23	0.70	0.78	53.49	6.04	66.40
FDCT	10	5.46	29.27	0.79	0.77	60.18	6.15	66.57
	20	5.63	25.60	0.75	0.78	59.60	6.79	67.40
	30	5.69	23.48	0.80	0.73	60.89	6.99	69.31
	40	4.88	19.09	0.76	0.81	63.49	6.74	71.55
BF-CNN	10	5.850	24.161	0.83	0.83	58.96	6.40	63.42
	20	6.154	26.724	0.78	0.78	50.94	6.61	63.62
	30	5.192	23.638	0.73	0.77	63.69	6.77	66.29
	40	5.452	17.397	0.76	0.73	63.45	7.06	74.90
DEN	10	4.99	28.37	0.78	0.80	61.50	7.68	68.11
	20	5.20	22.58	0.79	0.77	60.99	10.49	64.30
	30	4.56	21.67	0.77	0.76	65.01	10.80	64.40
	40	4.60	18.98	0.69	0.71	58.37	71.58	73.91
DCT-ACO	10	5.06	27.18	0.75	0.79	64.95	4.70	67.74
	20	5.20	23.84	0.74	0.76	59.45	5.99	68.57
	30	5.23	21.99	0.76	0.73	57.12	6.81	74.56
	40	4.62	16.78	0.69	0.70	60.42	7.64	69.08
NNE	10	5.09	26.72	0.78	0.83	55.04	6.97	60.87
	20	4.38	21.22	0.71	0.75	61.75	6.60	60.85
	30	4.04	20.81	0.71	0.73	65.09	7.10	69.58
	40	4.48	17.16	0.67	0.71	63.74	7.02	71.97
SDL	10	3.27	25.76	0.75	0.70	62.43	7.47	58.94
	20	4.27	22.96	0.74	0.69	60.96	7.23	66.77
	30	3.27	21.46	0.67	0.71	62.99	6.99	68.41
	40	4.27	14.23	0.60	0.69	58.43	7.24	67.73
WT	10	4.33	22.11	0.70	0.76	60.84	7.22	68.63
	20	3.33	20.11	0.69	0.69	58.60	7.42	66.68
	30	4.31	19.11	0.68	0.68	64.76	7.09	66.57
	40	3.29	14.10	0.65	0.70	66.97	7.00	72.85

Table 3: Quantitative evaluation of denoising performance for the CT dataset under varying noise levels (σ = 10, 20, 30, 40

Method	σ	Entropy↑	PSNR↑	SSIM↑	FSIM↑	BRISQUE↓	NIQE↓	PIQE↓
Ours	10	7.34	29.68	0.82	0.90	25.47	4.64	23.67
	20	7.23	27.45	0.73	0.85	12.64	3.38	18.08
	30	7.16	26.47	0.69	0.86	28.67	3.02	22.08
	40	7.09	25.36	0.63	0.83	23.65	3.01	28.10
TNN	10	7.33	28.89	0.81	0.88	27.89	4.77	24.67
	20	7.17	27.23	0.78	0.86	15.89	4.23	19.89
	30	6.99	26.23	0.67	0.85	29.99	3.67	24.99
	40	6.78	24.89	0.62	0.81	24.17	3.45	29.77
RED	10	7.22	27.78	0.82	0.86	28.56	4.98	25.65
	20	6.67	26.89	0.76	0.85	16.89	4.67	20.18

	30	6.57	25.45	0.66	0.67	29.67	4.23	25.47
	40	5.77	23.22	0.59	0.79	23.89	4.17	30.18
AWS-DCNN	10	7.19	28.11	0.83	0.85	27.98	4.77	25.77
	20	6.55	25.89	0.75	0.83	23.89	4.90	22.37
	30	6.37	26.18	0.69	0.69	23.12	5.27	26.88
	40	5.45	22.89	0.77	0.77	24.17	4.43	30.56
FOTV	10	7.13	27.89	0.82	0.85	26.56	4.57	26.89
	20	6.45	24.78	0.76	0.82	22.26	4.45	24.98
	30	6.17	26.88	0.83	0.84	24.29	4.27	27.90
	40	5.89	23.19	0.76	0.79	25.88	4.87	30.67
AKF-NLM	10	7.10	26.99	0.81	0.85	26.12	4.33	26.98
	20	6.17	26.55	0.81	0.83	23.89	4.67	25.76
	30	5.97	26.87	0.81	0.83	23.89	4.53	28.16
	40	6.10	22.97	0.68	0.77	27.89	4.96	29.99
OD-CNN	10	7.01	27.45	0.79	0.85	25.90	4.16	29.99
	20	7.00	23.67	0.73	0.81	25.87	4.13	27.70
	30	6.95	21.80	0.66	0.80	26.37	4.84	26.10
	40	6.56	20.02	0.61	0.78	30.78	5.10	28.15
NDiff	10	6.14	25.25	0.72	0.83	31.94	4.61	30.76
	20	6.71	21.76	0.69	0.82	32.30	4.30	29.44
	30	6.75	20.15	0.63	0.79	32.74	4.06	28.23
	40	6.62	19.09	0.59	0.76	33.46	4.36	35.30
SDPM	10	6.70	23.19	0.69	0.81	30.60	3.66	34.15
	20	6.88	22.55	0.63	0.78	30.08	4.19	31.32
	30	6.78	21.11	0.60	0.77	36.09	5.12	30.27
	40	6.00	18.07	0.57	0.73	33.46	4.40	31.27
NGN	10	6.33	25.72	0.73	0.80	30.60	4.96	35.65
	20	6.99	22.76	0.58	0.79	30.91	5.42	38.58
	30	6.49	23.34	0.58	0.77	33.56	4.74	44.13
	40	6.19	19.69	0.55	0.76	36.37	4.60	33.72
UDLF	10	6.34	23.49	0.67	0.71	37.29	5.42	37.71
	20	5.51	20.96	0.59	0.78	35.28	5.65	42.64
	30	5.58	19.31	0.53	0.78	42.83	5.07	41.62
	40	5.52	14.05	0.54	0.76	43.38	5.30	35.04
CNCL	10	6.17	23.51	0.62	0.79	41.31	5.24	40.60
	20	6.22	21.67	0.60	0.73	43.54	5.14	43.77
	30	6.23	20.90	0.59	0.72	42.93	5.54	48.53
	40	6.05	18.75	0.51	0.73	35.23	4.98	43.25
BP-MEC	10	5.97	24.31	0.65	0.74	37.55	4.24	47.41
	20	5.87	21.50	0.51	0.71	42.83	5.22	52.59
	30	5.93	19.85	0.55	0.69	43.44	5.55	46.08
	40	5.87	14.41	0.50	0.68	43.46	5.52	55.69
FDCT	10	5.19	23.37	0.66	0.79	34.16	6.05	55.86
	20	6.07	22.06	0.58	0.76	43.93	6.20	49.63
	30	5.10	20.30	0.52	0.72	37.64	6.69	48.81
	40	5.31	14.26	0.56	0.70	43.78	5.90	54.93
BF-CNN	10	5.941	23.536	0.60	0.77	44.75	6.17	56.62
	20	5.469	21.549	0.55	0.73	44.51	5.97	57.77
	30	5.521	18.591	0.52	0.73	45.39	4.58	51.80
	40	5.093	17.220	0.50	0.71	43.45	5.74	52.85
DEN	10	4.34	25.29	0.62	0.75	36.39	6.44	54.76
	20	5.13	21.00	0.50	0.73	41.12	6.16	59.82
	30	4.50	20.03	0.49	0.70	39.58	6.61	51.05
	40	5.00	13.61	0.47	0.68	43.48	5.19	57.05
DCT-ACO	10	5.44	25.08	0.64	0.77	36.42	6.03	58.37
	20	5.19	21.06	0.50	0.75	40.83	7.09	61.89
	30	4.66	20.15	0.47	0.69	46.99	5.68	53.12
	40	4.14	13.77	0.54	0.66	43.49	6.92	59.15
NNE	10	4.34	23.44	0.62	0.68	44.65	7.18	51.00
	20	4.15	21.24	0.49	0.66	40.19	6.12	55.22
	30	4.47	20.68	0.50	0.67	43.99	5.94	61.21

	40	4.07	14.38	0.44	0.65	43.47	6.35	63.91
SDL	10	3.47	24.64	0.61	0.73	45.78	6.03	62.89
	20	3.74	20.82	0.54	0.74	44.99	6.13	60.34
	30	3.47	20.50	0.46	0.73	46.45	6.43	61.87
	40	4.01	14.15	0.42	0.67	46.45	7.03	57.34
WT	10	3.99	21.61	0. 55	0.72	48.46	6.82	62.80
	20	3.81	20.45	0.47	0.71	47.19	6.43	63.67
	30	3.789	20.33	0.48	0.66	46.14	6.51	66.02
	40	3.87	13.32	0.43	0.65	47.78	6.54	64.65

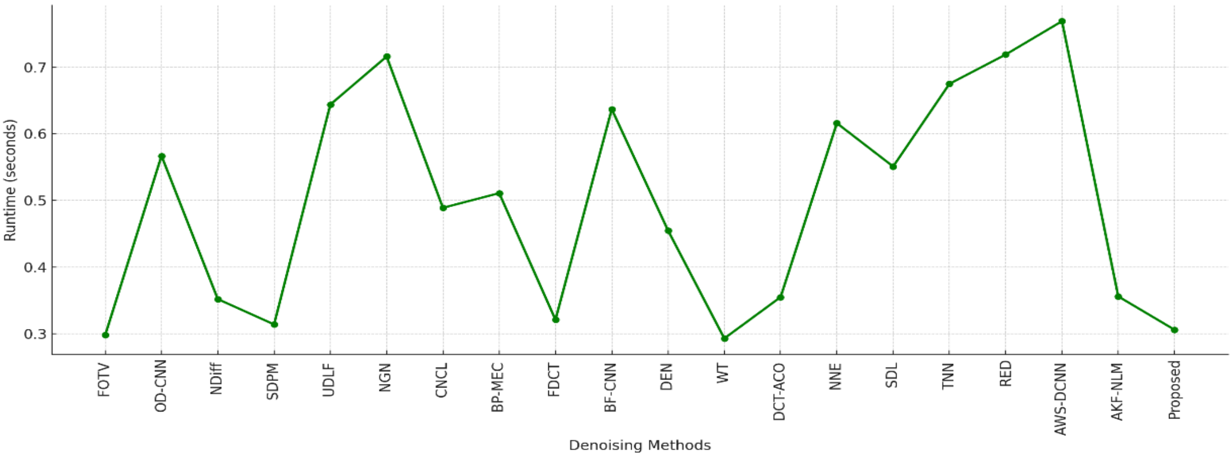

Fig. 14: Runtime analysis of different denoising methods

While this runtime is slightly higher than that of classical approaches such as FOTV and WT, which typically execute in around 2–3 seconds, it remains notably lower than several deep learning-based methods including OD-CNN, RED, and AWS-DCNN, which demand 5.0–5.5 seconds due to their intensive model inference and memory requirements. The overall computational complexity of the proposed method can be broken down into several stages. The initial patch extraction and vectorization scale linearly with image size, yielding a complexity of O (MN), where M×N is the image resolution. Clustering the extracted patches using K-means++ contributes significantly to the runtime, with a complexity of approximately $\mathrm{O}(N_p \mathrm{K} d I)$ where N_p is the number of patches, K is the number of clusters, d is the patch dimensionality, and I is the number of iterations. Within each cluster, PCA is performed incurring a cost of $\mathrm{O}(\mathrm{n}d^2)$, where n is the number of patches in the cluster. Subsequent steps such as Wiener filtering, inverse PCA, and patch aggregation are relatively efficient, with complexity ranging from linear to quadratic in patch size. Despite the computational overhead introduced by adaptive clustering and PCA-based subspace filtering, the proposed method provides a competitive runtime profile. Its performance balances well between computational cost and denoising effectiveness, delivering superior structure preservation compared to faster but less accurate methods. Moreover, the algorithm's modular structure opens opportunities for further acceleration using parallel processing or GPU-based implementation, making it a viable solution for both offline and semi-real-time medical imaging tasks. All runtime experiments were performed on the same Intel Core i7 (2.5 GHz, 16 GB RAM) CPU configuration used in the original experiments, ensuring standardized timing and fair computational comparison across all benchmark methods.

Ablation Study and Discussion

As shown in Table 4, each module contributes uniquely to the quality enhancement of the reconstructed image. The baseline PCA-only configuration provides basic denoising but leaves minor residual noise and edge smoothing. Introducing IMF prior to PCA improves local averaging stability, leading to approximately 1.2 dB gain in PSNR and 3.8 % improvement in SSIM. The inclusion of Adaptive Clustering further enhances structural coherence and reduces perceptual distortion, resulting in an additional 0.7 dB increase in PSNR, 2.6 % rise in FSIM, and ≈ 4.5 % reduction in NIQE. The complete proposed framework (IMF + Clustering + PCA) yields the best results, achieving an overall improvement of ~ 2.4 dB PSNR and 5–6 % SSIM compared to the PCA-only approach. These findings confirm that IMF strengthens initial noise suppression, adaptive clustering refines spatial homogeneity, and MP-based PCA thresholding preserves fine structural details, collectively producing superior quantitative and perceptual performance. Our framework introduces several exceptional aspects as compared other methods.

Table 4: Ablation Study: Quantitative Evaluation of Component-wise Contributions (MRI1 Dataset, σ = 20)

Configuration	PSNR (↑)	SSIM (↑)	FSIM (↑)	NIQE (↓)
PCA only	30.21	0.875	0.889	5.21
PCA + IMF	31.41	0.908	0.915	4.92
PCA + Clustering	32.09	0.917	0.928	4.67
Proposed (IMF + Clustering + PCA)	33.57	0.927	0.942	4.61

Geng et al. (2022) proposed a complementary learning strategy using dual deep predictors, which, although effective, relies heavily on large-scale annotated datasets and shows limited adaptability when training data are scarce. In contrast, our method is fully unsupervised, data-efficient, and does not require pre-training, making it practical for real-world clinical deployment where data availability is often restricted. (Taassori and vizvari, 2024) employed an adaptive Kalman filter combined with NLM and median filtering; while this approach adapts filter parameters to noise statistics, it tends to introduce residual blurring and insufficient texture recovery at higher noise levels. Our framework addresses this limitation through cluster-wise PCA thresholding guided by Marchenko Pastur eigenvalue selection, which preserves subtle textural features while effectively suppressing noise. Similarly, FDCT-based methods (Anandan et al., 2020) apply fixed multiscale transforms and global thresholding, which frequently oversmooth edges and fine details. By contrast, the integration of adaptive clustering, suboptimal Wiener filtering, and iterative mean refinement in our method enables detail-preserving denoising across varying noise intensities. This synergy of adaptive noise estimation, localized processing, and iterative refinement highlights the novelty of our contribution, distinguishing it from existing approaches in both design and performance.

Conclusion

In this paper, we present a medical denoising algorithm that integrates cluster-wise Principal Component Analysis (PCA) thresholding with an iterative mean filter to effectively reduce noise while preserving fine details in medical images. A key contribution of our work is a stable and accurate noise level estimation method tailored specifically for medical imagery. The denoising process leverages an iterative mean filtering approach, where a mean filter is repeatedly applied to image patches replacing each pixel with the average of its neighbours to gradually smooth noise while retaining critical structural information. To enhance denoising performance, we introduce an adaptive clustering mechanism that robustly groups similar patches, allowing more effective feature extraction. Within each cluster, a transform-domain PCA-based filter is employed to perform texture-preserving denoising. Additionally, a sliding window strategy combined with an aggregation step further refines the results, enhancing overall image quality. Experimental evaluations demonstrate up to 1.5 dB improvement in PSNR and higher SSIM scores across all test datasets, confirming improved noise suppression and structural fidelity. The proposed adaptive clustering approach delivers more robust denoising performance especially at higher noise levels, outperforming existing methods both quantitatively and visually. The method excels in preserving textures and structural details under Gaussian noise conditions. In the future, we intend to extend this framework to address broader image degradation challenges in medical imaging, moving beyond noise reduction to include restoration of lost or corrupted details.

In future work, we aim to integrate deep learning techniques, particularly Convolutional Neural Networks (CNNs) or transformer-based models, into the current denoising framework. This direction is justified by recent advances demonstrating that learning-based methods can model complex noise distributions and data priors more effectively than handcrafted algorithms. Specifically, a deep model could be trained to learn patch clustering behaviour or optimal PCA thresholds, thus reducing the need for manual parameter tuning. Furthermore, a hybrid architecture combining unsupervised framework with a learned post-processing stage could leverage the strengths of our approaches maintaining structural detail through cluster-wise PCA while enhancing denoising precision via learned features.

Acknowledgment

I sincerely express my gratitude to the Department of Allied Health Sciences, Chitkara School of Health Sciences, Chitkara University, Punjab, for their constant support and encouragement throughout this work. I am also thankful to Chitkara University, Punjab, for providing the necessary resources, facilities, and a conducive academic environment that made this work possible.

Funding Information

The authors received no financial support for this article's research, authorship, and/or publication.

Authors Contributions

Mohit Sharma: Concept, coding, developed the algorithm, experimentation, writing and edited.

Ayush Dogra: Coding, developed the algorithm, writing, methodology, formal analysis and validation.

Anita Gupta: Methodology, formal analysis and validation, edited.

Bhawna Goyal: Conduct experimentation and testing, visualization and validation of results.

Dawa Chyophel Lepcha: Verification of developed algorithm, analysis and validation of results, edited.

Archana Saini: Analysis of results, review and proofreading.

Ethics

This research did not involve human participants, animal subjects, or any material requires ethical approval.

Conflict of Interest

The authors do not have any conflicts of interest.

Data Availability Statement

The data used in this paper can be publicly accessed from the official website (https://www.med.harvard.edu/aanlib/home.html) and https://github.com/Imagingscience/Image-Fusion-Image-Denoising-Image-Enhancement-.

References

Abdi, H., & Williams, L. J. (2010). Principal component analysis. *WIREs Computational Statistics*, *2*(4), 433–459. https://doi.org/10.1002/wics.101

Anandan, P., Giridhar, A., Lakshmi, E. I., & Nishitha, P. (2020). Medical Image Denoising using Fast Discrete Curvelet Transform. *International Journal of Emerging Trends in Engineering Research*, *8*(7), 3760–3765. https://doi.org/10.30534/ijeter/2020/139872020

Annavarapu, A., & Borra, S. (2024). An adaptive watershed segmentation based medical image denoising using deep convolutional neural networks. *Biomedical Signal Processing and Control*, *93*, 106119. https://doi.org/10.1016/j.bspc.2024.106119

Arthur, D., & Vassilvitskii, S. (2006). *k-means++: The advantages of careful seeding*. 1027–1035.

Arun, A., Thomas, T. J., Rani, J. S., & Subrahmanyam Gorthi, R. K. S. (2020). Efficient directionality-driven dictionary learning for compressive sensing magnetic resonance imaging reconstruction. *Journal of Medical Imaging*, *7*(01), 1. https://doi.org/10.1117/1.jmi.7.1.014002

Atal, D. K. (2023). Optimal Deep CNN–Based Vectorial Variation Filter for Medical Image Denoising. *Journal of Digital Imaging*, *36*(3), 1216–1236. https://doi.org/10.1007/s10278-022-00768-8

Bai, J., Song, S., Fan, T., & Jiao, L. (2018). Medical image denoising based on sparse dictionary learning and cluster ensemble. *Soft Computing*, *22*(5), 1467–1473. https://doi.org/10.1007/s00500-017-2853-7

Buades, A., Coll, B., & Morel, J.-M. (2005a). A Review of Image Denoising Algorithms, with a New One. *Multiscale Modeling & Simulation*, *4*(2), 490–530. https://doi.org/10.1137/040616024

Buades, A., Coll, B., & Morel, J.-M. (2005b). A Non-Local Algorithm for Image Denoising. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)* (pp. 60–65). https://doi.org/10.1109/cvpr.2005.38

Chen, G., Zhu, F., & Heng, P. A. (2015). An Efficient Statistical Method for Image Noise Level Estimation. *2015 IEEE International Conference on Computer Vision (ICCV)*, 477–485. https://doi.org/10.1109/iccv.2015.62

Chen, H., Zhang, Y., Kalra, M. K., Lin, F., Chen, Y., Liao, P., Zhou, J., & Wang, G. (2017). Low-Dose CT With a Residual Encoder-Decoder Convolutional Neural Network. *IEEE Transactions on Medical Imaging*, *36*(12), 2524–2535. https://doi.org/10.1109/tmi.2017.2715284

Cheplygina, V., de Bruijne, M., & Willemien Pluim, J. P. (2019). Not-so-supervised: A survey of semi-supervised, multi-instance, and transfer learning in medical image analysis. *Medical Image Analysis*, *54*, 280–296. https://doi.org/10.1016/j.media.2019.03.009

Dabov, K., Foi, A., Katkovnik, V., & Egiazarian, K. (2007). Image Denoising by Sparse 3-D Transform-Domain Collaborative Filtering. In *IEEE Transactions on Image Processing* (Vol. 16, Issue 8, pp. 2080–2095). https://doi.org/10.1109/tip.2007.901238

Diwakar, M., Singh, P., & Garg, D. (2024). Edge-guided filtering based CT image denoising using fractional order total variation. *Biomedical Signal Processing and Control*, *92*, 106072. https://doi.org/10.1016/j.bspc.2024.106072

Donoho, D. L., & Gavish, M. (2013). The optimal hard threshold for singular values is 4/sqrt (3. *ArXiv*. https://doi.org/10.1109/TIT.2014.2323359

Elhoseny, M., & Shankar, K. (2019). Optimal bilateral filter and Convolutional Neural Network based denoising method of medical image measurements. *Measurement*, *143*, 125–135. https://doi.org/10.1016/j.measurement.2019.04.072

Fan, L., Zhang, F., Fan, H., & Zhang, C. (2019). Brief review of image denoising techniques. *Visual Computing for Industry, Biomedicine, and Art*, *2*(1), 7. https://doi.org/10.1186/s42492-019-0016-7

Ferdi, A., Benierbah, S., & Nakib, A. (2024). Residual encoder-decoder based architecture for medical image denoising. *Multimedia Tools and Applications*, *84*(19), 21625–21642. https://doi.org/10.1007/s11042-024-20175-1

Fu, B., Zhang, X., Wang, L., Ren, Y., & Thanh, D. N. H. (2022). A blind medical image denoising method with noise generation network. *Journal of X-Ray Science and Technology*, *30*(3), 531–547. https://doi.org/10.3233/xst-211098

Geng, M., Meng, X., Yu, J., Zhu, L., Jin, L., Jiang, Z., Qiu, B., Li, H., Kong, H., Yuan, J., Yang, K., Shan, H., Han, H., Yang, Z., Ren, Q., & Lu, Y. (2022). Content-Noise Complementary Learning for Medical Image Denoising. *IEEE Transactions on Medical Imaging*, *41*(2), 407–419. https://doi.org/10.1109/tmi.2021.3113365

Goyal, B., Dogra, A., Lepcha, D. C., Goyal, V., Alkhayyat, A., Chohan, J. S., & Kukreja, V. (2024). Recent advances in image dehazing: Formal analysis to automated approaches. *Information Fusion*, *104*, 102151. https://doi.org/10.1016/j.inffus.2023.102151

Higaki, T., Nakamura, Y., Tatsugami, F., Nakaura, T., & Awai, K. (2019). Improvement of image quality at CT and MRI using deep learning. *Japanese Journal of Radiology*, *37*(1), 73–80. https://doi.org/10.1007/s11604-018-0796-2

Ji, L., Guo, Q., & Zhang, M. (2020). Medical Image Denoising Based on Biquadratic Polynomial With Minimum Error Constraints and Low-Rank Approximation. *IEEE Access*, *8*, 84950–84960. https://doi.org/10.1109/access.2020.2990463

Katkovnik, V., Foi, A., Egiazarian, K., & Astola, J. (2010). From Local Kernel to Nonlocal Multiple-Model Image Denoising. *International Journal of Computer Vision*, *86*(1), 1–32. https://doi.org/10.1007/s11263-009-0272-7

Kaur, A., & Dong, G. (2023). A Complete Review on Image Denoising Techniques for Medical Images. *Neural Processing Letters*, *55*(6), 7807–7850. https://doi.org/10.1007/s11063-023-11286-1

Kim, K., Soltanayev, S., & Chun, S. Y. (2020). Unsupervised Training of Denoisers for Low-Dose CT Reconstruction Without Full-Dose Ground Truth. *IEEE Journal of Selected Topics in Signal Processing*, *14*(6), 1112–1125. https://doi.org/10.1109/jstsp.2020.3007326

Kollem, S., Reddy, K. R., & Rao, D. S. (2023). A novel diffusivity function-based image denoising for MRI medical images. *Multimedia Tools and Applications*, *82*(21), 32057–32089. https://doi.org/10.1007/s11042-023-14457-3

Laves, M.-H., Tölle, M., & Ortmaier, T. (2020). Uncertainty Estimation in Medical Image Denoising with Bayesian Deep Image Prior. *Uncertainty for Safe Utilization of Machine Learning in Medical Imaging, and Graphs in Biomedical Image Analysis*, *12443*, 81–96. https://doi.org/10.1007/978-3-030-60365-6_9

Lepcha, D. C., Dogra, A., Goyal, B., Goyal, V., Kukreja, V., & Bavirisetti, D. P. (2023a). A constructive non-local means algorithm for low-dose computed tomography denoising with morphological residual processing. *PLOS ONE*, *18*(9), e0291911. https://doi.org/10.1371/journal.pone.0291911

Lepcha, D. C., Goyal, B., Dogra, A., Sharma, K. P., & Gupta, D. N. (2023b). A deep journey into image enhancement: A survey of current and emerging trends. *Information Fusion*, *93*, 36–76. https://doi.org/10.1016/j.inffus.2022.12.012

Liu, P., El Basha, M. D., Li, Y., Xiao, Y., Sanelli, P. C., & Fang, R. (2019). Deep Evolutionary Networks with Expedited Genetic Algorithms for Medical Image Denoising. *Medical Image Analysis*, *54*, 306–315. https://doi.org/10.1016/j.media.2019.03.004

Liu, P., Li, Y., El Basha, M. D., & Fang, R. (2018). Neural Network Evolution Using Expedited Genetic Algorithm for Medical Image Denoising. *Medical Image Computing and Computer Assisted Intervention – MICCAI 2018*, *11070*, 12–20. https://doi.org/10.1007/978-3-030-00928-1_2

Liu, X., Tanaka, M., & Okutomi, M. (2013). Single-Image Noise Level Estimation for Blind Denoising. *IEEE Transactions on Image Processing*, *22*(12), 5226–5237. https://doi.org/10.1109/tip.2013.2283400

Luisier, F., Blu, T., & Unser, M. (2011). Image Denoising in Mixed Poisson–Gaussian Noise. *IEEE Transactions on Image Processing*, *20*(3), 696–708. https://doi.org/10.1109/tip.2010.2073477

Luthra, A., Sulakhe, H., Mittal, T., Iyer, A., & Yadav, S. (2021). Eformer: Edge enhancement based transformer for medical image denoising. *ArXiv*. https://doi.org/https://doi.org/10.48550/arXiv.2109.08044

Manjón, J. V., & Coupe, P. (2018). MRI Denoising Using Deep Learning. *Patch-Based Techniques in Medical Imaging*, 12–19. https://doi.org/10.1007/978-3-030-00500-9_2

Miri, A., Sharifian, S., Rashidi, S., & Ghods, M. (2018). Medical image denoising based on 2D discrete cosine transform via ant colony optimization. *Optik*, *156*, 938–948. https://doi.org/10.1016/j.ijleo.2017.12.074

Nazir, N., Sarwar, A., & Saini, B. S. (2024). Recent developments in denoising medical images using deep learning: An overview of models, techniques, and challenges. *Micron*, *180*, 103615. https://doi.org/10.1016/j.micron.2024.103615

Niknejad Mazandarani, F., Babyn, P., & Alirezaie, J. (2025). SADiff: A Sinogram-Aware Diffusion Model for Low-Dose CT Image Denoising. *Journal of Imaging Informatics in Medicine*, *38*(6), 4255–4275. https://doi.org/10.1007/s10278-025-01469-8

Portilla, J., Strela, V., Wainwright, M. J., & Simoncelli, E. P. (2003). Image denoising using scale mixtures of gaussians in the wavelet domain. *IEEE Transactions on Image Processing*, *12*(11), 1338–1351. https://doi.org/10.1109/tip.2003.818640

Pyatykh, S., Hesser, J., & Zheng, L. (2013). Image Noise Level Estimation by Principal Component Analysis. *IEEE Transactions on Image Processing*, *22*(2), 687–699. https://doi.org/10.1109/tip.2012.2221728

Qu, H., Liu, K., & Zhang, L. (2024). Research on improved black widow algorithm for medical image denoising. *Scientific Reports*, *14*(1), 2514. https://doi.org/10.1038/s41598-024-51803-3

Rai, S., Bhatt, J. S., & Patra, S. K. (2021). An unsupervised deep learning framework for medical image denoising. *ArXiv*. https://doi.org/https://doi.org/10.48550/arXiv.2103.06575

Rai, S., Bhatt, J. S., & Patra, S. K. (2021). Augmented Noise Learning Framework for Enhancing Medical Image Denoising. *IEEE Access*, *9*, 117153–117168. https://doi.org/10.1109/access.2021.3106707

Rajesh, C., & Kumar, S. (2022). An evolutionary block based network for medical image denoising using Differential Evolution. *Applied Soft Computing*, *121*, 108776. https://doi.org/10.1016/j.asoc.2022.108776

Satapathy, L. M., Das, P., Shatapathy, A., & Patel, A. K. (2019). Bio-medical image denoising using wavelet transform. *International Journal of Recent Technology and Engineerin*, *8*(1), 2874–2879.

Shan, H., Zhang, Y., Yang, Q., Kruger, U., Kalra, M. K., Sun, L., Cong, W., & Wang, G. (2018). 3-D Convolutional Encoder-Decoder Network for Low-Dose CT via Transfer Learning From a 2-D Trained Network. *IEEE Transactions on Medical Imaging*, *37*(6), 1522–1534. https://doi.org/10.1109/tmi.2018.2832217

Sharif, S. M. A., Naqvi, R. A., & Biswas, M. (2020). Learning Medical Image Denoising with Deep Dynamic Residual Attention Network. *Mathematics*, *8*(12), 2192. https://doi.org/10.3390/math8122192

Shen, C., Yang, Z., & Zhang, Y. (2023). PET Image Denoising with Score-Based Diffusion Probabilistic Models. *Medical Image Computing and Computer Assisted Intervention – MICCAI 2023 Conference Paper*, *14220*, 270–278. https://doi.org/10.1007/978-3-031-43907-0_26

Singh, P., Diwakar, M., Gupta, R., Kumar, S., Chakraborty, A., Bajal, E., Jindal, M., Shetty, D. K., Sharma, J., Dayal, H., Naik, N., & Paul, R. (2022). A Method Noise-Based Convolutional Neural Network Technique for CT Image Denoising. *Electronics*, *11*(21), 3535. https://doi.org/10.3390/electronics11213535

Singh, S., Póczos, B., & Ma, J. (2018). Minimax reconstruction risk of convolutional sparse dictionary learning. *International Conference on Artificial Intelligence and Statistics*, 1327–1336.

Swetha, D., & Jyothi, N. (2025). Medical Image Denoising Using a Novel Multilevel Convolutional Optimized Visual Attention Network in Diverse Dataset. *International Journal of Image and Graphics*, *14*, 2750025. https://doi.org/10.1142/s0219467827500252

Taassori, M., & Vizvári, B. (2024). Enhancing Medical Image Denoising: A Hybrid Approach Incorporating Adaptive Kalman Filter and Non-Local Means with Latin Square Optimization. *Electronics*, *13*(13), 2640. https://doi.org/10.3390/electronics13132640

Thanh, D. N. H., Hieu, L. M., & Enginoglu, S. (2019). An Iterative Mean Filter for Image Denoising. *IEEE Access*, *7*, 167847–167859. https://doi.org/10.1109/access.2019.2953924

Veraart, J., Fieremans, E., & Novikov, D. S. (2016a). Diffusion MRI noise mapping using random matrix theory. *Magnetic Resonance in Medicine*, *76*(5), 1582–1593. https://doi.org/10.1002/mrm.26059

Veraart, J., Novikov, D. S., Christiaens, D., Ades-aron, B., Sijbers, J., & Fieremans, E. (2016b). Denoising of diffusion MRI using random matrix theory. *NeuroImage*, *142*, 394–406. https://doi.org/10.1016/j.neuroimage.2016.08.016

Yang, Q., Yan, P., Zhang, Y., Yu, H., Shi, Y., Mou, X., Kalra, M. K., Zhang, Y., Sun, L., & Wang, G. (2018). Low-Dose CT Image Denoising Using a Generative Adversarial Network With Wasserstein Distance and Perceptual Loss. *IEEE Transactions on Medical Imaging*, *37*(6), 1348–1357. https://doi.org/10.1109/tmi.2018.2827462

Yi, X., Walia, E., & Babyn, P. (2019). Generative adversarial network in medical imaging: A review. *Medical Image Analysis*, *58*, 101552. https://doi.org/10.1016/j.media.2019.101552

Zeng, X., Guo, Y., Li, L., & Liu, Y. (2024). Continual medical image denoising based on triplet neural networks collaboration. *Computers in Biology and Medicine*, *179*, 108914. https://doi.org/10.1016/j.compbiomed.2024.108914

Zhang, L., Dong, W., Zhang, D., & Shi, G. (2010). Two-stage image denoising by principal component analysis with local pixel grouping. *Pattern Recognition*, *43*(4), 1531–1549. https://doi.org/10.1016/j.patcog.2009.09.023

Zhao, W., Lv, Y., Liu, Q., & Qin, B. (2018). Detail-Preserving Image Denoising via Adaptive Clustering and Progressive PCA Thresholding. *IEEE Access*, *6*, 6303–6315. https://doi.org/10.1109/access.2017.2780985

Zhao, W., Lv, Y., Liu, Q., & Qin, B. (2018). Detail-Preserving Image Denoising via Adaptive Clustering and Progressive PCA Thresholding. *IEEE Access*, *6*, 6303–6315. https://doi.org/10.1109/access.2017.2780985