Enhancing Image Denoising Efficiency: Dynamic Transformations in BM3D Algorithm

Ayyub Hamdanu Budi Nurmana, Mars Caroline Wibowo*, and Sarwo Nugroho

Department of Visual Communication Design, University of Science and Computer Technology, Semarang, Indonesia
Email: nurmana@stekom.ac.id (A.H.B.N.); caroline@stekom.ac.id (M.C.W.); sarwo@stekom.ac.id (S.N.)
*Corresponding author

***Abstract*—This research aims to enhance the performance of image-denoising algorithms, particularly in the context of Block Matching 3D (BM3D) usage, focusing on improving image quality and retaining important information in noisy images. The novelty of this research lies in developing more effective and efficient image-denoising techniques by considering the characteristics of image blocks to improve denoising results. The method employed in this research involves the development of a new approach enabling the application of adaptive 2D and 3D transformations depending on the characteristics of the image block being processed. The research develops a new approach enabling the application of adaptive 2D and 3D transformations depending on the characteristics of the image block being processed. The results of this research indicate that the proposed adaptive approach in the BM3D image denoising algorithm can significantly improve denoising performance. Experimental results show that performing 2D transformations on blocks that do not have sufficiently similar blocks can yield better denoising results, especially at high noise levels.**

***Keywords*—image denoising, adaptive transformation, image processing, Block Matching 3D (BM3D) algorithm**

I. Introduction

Denoising images serves as a critical pre-processing step, essential for a myriad of applications including image restoration, visual tracking, and image segmentation to function effectively. Within the realm of image processing, the enhancement of denoising algorithms holds substantial importance, aiding in the enhancement of image quality and retention of vital information [1]. Block Matching 3D (BM3D) has emerged as the most advanced algorithm in image denoising, providing superior performance compared to previous methods [2, 3]. However, there is still room for improvement in BM3D's performance. The widespread need to address noise interference in images cannot be ignored, as such interference often reduces visual quality and blurs important information stored within, especially in high-quality images across various applications such as medical imaging, surveillance, and multimedia. On the other hand, although the BM3D algorithm is one of the latest algorithms in image denoising, there are shortcomings in the approach used, and there are still gaps in the analysis that need to be addressed in the approach used by this algorithm. Specifically, the default approach of BM3D involves 3D transformation on all image blocks without considering whether similar blocks are available enough to be used as references. The impact of this approach is particularly noticeable at high noise levels, which can result in a significant decrease in denoising performance.

The BM3D algorithm by default performs 3D transformation on all image blocks without considering whether there are enough similar blocks that can be found. This results in a decrease in denoising performance, especially at high levels of noise. In the analysis of several kinds of literature, there has been no similar approach to improving the performance of BM3D. Considering the shortcomings in the BM3D approach related to the universally performed 3D transformation, this research attempts to contribute to improving the effectiveness of image denoising. Therefore, this study attempts to identify existing gaps while proposing a novel method enabling the flexible application of both 2D and 3D transformations depending on the characteristics of the image blocks in question with the hope that the implementation of the proposed approach will bring significant improvements in image denoising performance, expanding the coverage and applicability of the BM3D algorithm in the context of handling more complex noise. The main contribution of this research is the enhancement of BM3D algorithm performance in image denoising through the merging of adaptive 2D and 3D transformations, thereby improving denoising performance by considering the availability of similar blocks in the image. Thus, this research provides new insights into developing more effective and efficient image-denoising techniques. Additionally, this research also demonstrates that the proposed approach can result in lower computational time, which is an additional contribution to the development of more efficient denoising algorithms.

II. Literature Review

A. Image Noise

Image noise refers to random disturbances or artifacts that appear in images, which can degrade visual quality

Manuscript received April 1, 2024; revised July 4, 2024; accepted July 16, 2024; published October 8, 2024.

and blur the conveyed information. Types of image noise can vary, ranging from noise generated by recording devices, such as cameras or sensors [4, 5], to noise that arises during data processing and transmission [6]. Image noise can pose a significant challenge in various applications, including photography, medical processing, and image recognition, as it can disrupt the interpretation and analysis of visual data [5, 7–9]. The importance of addressing image noise lies in efforts to maintain the integrity of visual information, improve image quality, and enhance accuracy in the analysis of various applications. Various techniques have been developed to reduce image noise, including spatial filters, the use of adaptive algorithms, and statistical model-based approaches. For instance, Karanam *et al.* [10] used a statistical model approach to reduce image noise in the healthcare field, and the results showed that their proposed model had good accuracy for binary and multiclass classification of fractures. Additionally, noise reduction techniques on images using algorithms were also used in [11, 12], and their results indicated they could detect the impact of applying the Sobel algorithm. Finally, other approaches such as variations of spatial filters were also used in [13, 14]. Additionally, a profound understanding of the sources of noise and their characteristics is crucial for selecting and implementing denoising techniques that suit the specific needs of the application.

1) Non-spatially dependent noise

Non-spatially dependent noise is a type of noise that is uniform across the entire image area without regard to the structure or visual content. This is often caused by factors such as electronic noise in image sensors or signal disturbances during data transmission. Examples include white noise, which has a uniform frequency spectrum across the entire frequency range and can degrade overall image quality. Such noise is difficult to remove because there is no spatial pattern that can be exploited to distinguish it from the original image content. One of the main challenges in addressing non-spatially dependent noise is separating the noise from relevant signals in the image. Because such noise is spread throughout the image without a specific pattern, the appropriate approach must take into account the statistical characteristics of the noise. Effective denoising techniques for such noise often involve the use of linear or non-linear filters specifically designed to reduce noise without disturbing desired image content. The implementation of denoising for non-spatially dependent noise often requires a compromise between reducing noise and preserving important image details [15]. Several denoising filters used to address such noise include Gaussian filters and median filters. Gaussian filters emphasize gradual smoothing without sacrificing detail, while median filters are effective in removing isolated noise points without blurring significant image content. In addition to denoising techniques focused on reducing spatial noise, the use of data compression techniques can also help mitigate the effects of non-spatially dependent noise. By reducing the amount of data that needs to be stored or transmitted, data compression can reduce imperfections caused by noise without sacrificing important information in the image. Compression techniques such as JPEG, for example, use a combination of spatial transformation and calculations to create a more compact representation of the image that minimizes the effects of non-spatially dependent noise.

2) Spatially dependent noise

Spatially dependent noise is a type of noise that exhibits specific patterns or structures within digital images. This type of noise is often caused by factors such as damage to image sensors or disturbances during the data transmission process. Common examples of spatial noise include noise points, vertical or horizontal lines, or regular noise patterns. This noise tends to disrupt visual interpretation and can significantly degrade image quality. Spatially dependent noise often becomes the primary focus in digital image processing due to its more visible and directly noticeable impact. Denoising techniques used to address spatial noise often leverage spatial information from the image to identify and remove such noise. Spatial filters such as Gaussian or median filters are often used to smooth images and remove spatial noise without sacrificing important details in the image [16, 17]. Additionally, spatially dependent noise can affect the performance of more complex image processing algorithms, such as image segmentation or object recognition. Disturbances in spatial structure can blur object boundaries and disrupt accurate feature extraction [18]. Therefore, reducing spatial noise becomes an important step in data preparation before running advanced processing algorithms. Although spatially dependent noise can pose challenges, appropriate denoising techniques can significantly improve image quality and analysis results. Common approaches involve detecting and removing noise based on spatial characteristics such as size, shape, or pattern. By understanding the nature of spatial noise and using suitable denoising algorithms, image processing can be performed more efficiently and accurately.

3) Related work

Each type of noise has different denoising techniques. In this case, we employ supplementary White Gaussian noise alongside various denoising methodologies to address noise-related issues in images. This study considers noise with three properties. Firstly, the noise examined in this investigation remains separate from the signal, and systematically, this noise will be added to the pixel values. Furthermore, the noise samples under examination exhibit independence, thereby necessitating that all noise samples be sourced from a uniform distribution.

Recent research has shown significant improvements in image-denoising techniques by integrating deep learning and adaptive filtering methods. For instance, Ju *et al.* [8] utilized dual-uncertainty estimation to improve the classification of medical images with noise. This approach combines deep learning with adaptive noise reduction to enhance classification accuracy [8]. Kong *et al.* [11] proposed an improved Non-Local Means (NLM) algorithm for CT image denoising. The method adapts the search window size based on the noise level, thereby enhancing the denoising effectiveness [11].

Similarly, [19] introduced a self-guided deep-learning technique for Magnetic Resonance Imaging (MRI) noise reduction. This technique combines deep learning with adaptive filters to improve image quality, demonstrating significant performance gains [19].

B. Image Noise Removal using

1) Spatial filtering method

a) Arithmetic Mean Filter (ArMF)

Average Mean Filter (ArMF), is a fundamental technique in image denoising. This filter operates by replacing each pixel value with the average value of its neighboring pixels within a specified kernel or window. By averaging pixel values, the filter aims to reduce noise while preserving the general structure and features of the image. One of the primary advantages of the Arithmetic Mean Filter is its simplicity and computational efficiency, making it widely used in various denoising applications. However, while Arithmetic Mean Filter can effectively reduce noise, it may also blur the image and reduce sharpness, especially in areas with high frequency or intricate details. Additionally, this filter is sensitive to outliers and may not adequately handle impulsive noise, such as salt-and-pepper noise. Consequently, while the Arithmetic Mean Filter serves as a basic denoising approach, it is often combined with more sophisticated techniques or used as a pre-processing step in image enhancement pipelines to achieve better results. ArMF strives to discover an estimation that reduces the average squared difference between the noisy values and their corresponding estimates within the spatial domain. For the ArMF equation, it employs an equation from [1], as in Eq. (1).

$$F(x,y) = \frac{1}{mn}\sum_{(s,t)\in S_{xy}} g(s,t) \quad (1)$$

where *(x, y)* are the coordinates of the pixel being processed, and *F* is the filter function applied at the coordinates *(x, y). 1/mn* is the normalization factor, where m and n are the dimensions of the applied filter. S_{xy} is a group of pixel coordinates within the neighborhood of *(x, y)* according to the filter size, and $\sum(s,t) \in S_{xy}$ denotes the summation operation for all pixel coordinates (s, t). The average filter efficiently reduces local fluctuations in the image. However, as a consequence, the resulting cleaned image might exhibit significant blurring.

b) Wiener filtering

This evaluation of error is expressed as [1] in Eq. (2). Where e^2 is the Mean Square Error (MSE) that will be minimized in the Wiener Filtering process, while *E* represents the expectation indicating the average value of a random variable. *f* is the original pixel value (the undistorted original signal), and *F* is the pixel value after filtering (the filtered signal)

$$e^2 = E\{(f-F)^2\} \quad (2)$$

2) Non-linear spatial filter

a) Median filter and Adaptive Median Filter (AMF)

The Median Filter stands as a widely utilized order-statistic filter in image processing. Order-statistic filters operate spatially, relying on the arrangement of pixels within the filter's coverage area. At each point, the filter's response is influenced by the pixel ranking outcome. By substituting the pixel value with the median of the surrounding gray levels, the Median Filter proves notably efficient, particularly when dealing with unipolar and bipolar impulse noise. However, the performance of the Median Filter will decrease with a high spatial density of impulse noise [20]. The AMF demonstrates robust performance even in scenarios with a high density of impulse noise. It functions within a rectangular window area contingent upon the distribution of grey-level values, dynamically adjusting the window size to accommodate variations in the image.

C. Method Based on Domain Filtering Transformation

In this research, this method is used to transform pixel intensity values into another transformation domain. The transformation domain will be chosen in such a way that noise and signal values can be separated as best as possible. Subsequently, filtering is performed on the transformed coefficients to obtain a sparse representation. Finally, the transformation is reversed to obtain the actual pixel values.

1) Spatial frequency filtering and Anisotropic Diffusion Filtering (AnDF)

Subsequently, a low-pass filter is employed to eliminate noise, with the cut-off frequency carefully selected to dissociate noise from the valuable signal. Low-pass filtering entails permitting low-frequency content to pass through while impeding high-frequency elements. The identification of low-frequency and high-frequency elements depends on the selected cut-off frequency. Normally, the low-frequency elements within the image are associated with uniform areas, whereas the high-frequency elements pertain to characteristics such as edges and noise. Afterward, the transformed coefficients undergo an inverse Fourier transform to obtain the cleaned image in the spatial domain. However, a significant limitation of this filtering method arises from the dispersion of edge information across frequencies, creating difficulties in accurately discerning between edges and noise in the frequency spectrum, particularly when they share similar frequencies. Consequently, in the denoising process, this filtering approach often sacrifices edge information, resulting in the degradation of edge quality. AnDF [21–23] is used to clean the image while preserving edges. So, it is utilized to distribute the image uniformly across regions while constraining spreading in front of the edges, thus allowing better cleaning results.

D. Non-Local Means (NLM) Algorithm

NLM [24, 25], attempts to exploit redundancy in natural images. Natural images likely contain many examples of similar blocks. The NLM algorithm achieves pixel cleaning by averaging all pixel values in the image during smoothing, with a greater emphasis placed on pixels

sharing a similar neighborhood with the current pixel. Recent years have seen advancements in the traditional NLM algorithm. However, the original NLM's effectiveness can be compromised if a pixel resides in either a smooth or rough region, depending on the window size applied [26]. When using a small window size in a smooth area, the cleaning might not be sufficiently effective, whereas employing a larger search window in a rough area may lead to loss of features. Hence, adapting the search window size to the context under consideration can potentially enhance cleaning effectiveness. This adaptive window size approach tailors the window size for each pixel, aiding in the preservation of enhancing image intricacies to yield marginally improved results in terms of Peak Signal-to-Noise Ratio (PSNR) compared to the original NLM, albeit at the expense of increased computational complexity. The initial NLM algorithm employs a Gaussian-weighted template with consistent coefficients across all adjacent pixels [26]. However, these fixed weights are susceptible to noise interference, impacting the cleaning performance during averaging. Adapting these weighted templates based on the noise level can potentially yield improved results. This method dynamically adjusts Gaussian weight coefficients using the Laplace operator. Initially, the noisy image is cleaned using traditional NLM, yielding the base image. If the noise persists at a point, the Laplace operator at that point exhibits high intensity. Subsequently, the neighbors are segregated into disjoint regions, and the weights of each region are computed using gradient and Laplace information. Ultimately, NLM is implemented utilizing this adjusted weighting factor matrix. This method not only exhibits improved visual results and higher PSNR values but also conserves more textures and edges when compared to the original NLM. However, it does require the initial use of traditional NLM.

E. BM3D Filtering Based Denoising

The BM3D algorithm consists of two steps and it will summarized into simpler steps as illustrated in Fig. 1. Eq. (3) describes the observation model in the BM3D algorithm, where the signal (observed image) *z(x)* is the sum of the original signal *y(x)* and noise $\eta(x)$ from the observed signal *z(x)*. This results in the original signal *y(x)* being cleaner and of higher quality.

$$z(x) = y(x) + \eta(x) \tag{3}$$

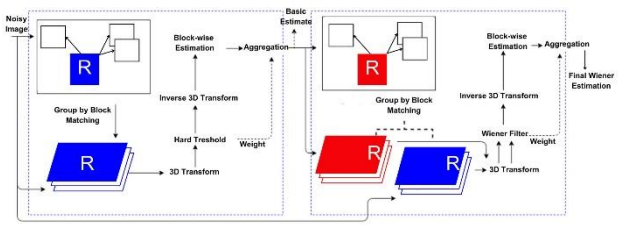

Fig. 1. Block diagram of the BM3D algorithm.

1) Step 1

The first step is to obtain the baseline image. In this process, block-wise estimation occurs, involving collaborative grouping and hard-thresholding. Grouping involves the search for blocks akin to the presently processed block, which are then aggregated to constitute a group. Subsequently, a 3D transformation is applied to this group, followed by hard thresholding on the transformation coefficients to mitigate noise. The transformation is then reversed to generate estimates for all grouped blocks, which are restored to their initial positions. Furthermore, aggregation is achieved by computing the baseline estimate for all pixels through a weighted average of all overlapping block estimates. Distance measurement d uses the following Eq. (4). For distance measurement *d*, it is done using Eq. (4), where R_{XR} and Z_X are the blocks being compared, while N_1^{ht} is the block size used in hard-thresholding.

$$d\_noisy(R_{XR}, Z_X) = \frac{||Z_{XR} - Z_X||_2^2}{(N_1^{ht})^2} \tag{4}$$

BM3D creates a collection of comparable patches for the present processing patch using the subsequent equation. To form a set of similar patches S_{xR}^{ht}, it is calculated using Eq. (5), where $\tau_{max-patch}^{ht}$ is the maximum threshold for the patch.

$$S_{xR}^{ht} = \left(\frac{x \in X}{d(Z_{xR,}\, Z_X) \leq \tau_{max-patch}^{ht}} \right) \tag{5}$$

For the inverse 3D transformation, calculate using Eq. (6), where γ is the function applied to the transformation result.

$$\Upsilon_{S_{xR}^{ht}} = \tau_3 D^{(ht)^{-1}} \left(\gamma \left(\tau_3 D^{ht} \left(Z_{S_{xR}^{ht}} \right) \right) \right) \tag{6}$$

In the aggregation stage to obtain the baseline estimates for each pixel, the considered weights as W_{xR}^{ht} in Eq. (7), σ is the noise deviation standard and N_{har}^{xR} is a total block on the group.

$$W_{xR}^{ht} = \begin{Bmatrix} \frac{1}{\sigma^{-2} N_{har}^{xR}}, & if\ N_{har}^{xR} \geq 1 \\ 1, & otherwise \end{Bmatrix} \tag{7}$$

For the baseline estimation at each pixel as $\Upsilon^{basic}(x)$, this calculation follows Eq. (8), where $X_m \in S_{xR}^{ht}$ is a group of patch similar to the patch being processed, and, $X_{x_m}^{(x)}$ is an initial patch from x_m.

$$\Upsilon^{basic}(x) = \frac{\sum_{XR \in X} \sum_{X_m \in S_{xR}^{ht}} W_{xR}^{ht} \Upsilon_{x_m}^{(ht,xR)}}{\sum_{XR \in X} \sum_{X_m \in S_{xR}^{ht}} W_{xR}^{ht} X_{x_m}^{(x)}},\ \forall x \in X \tag{8}$$

2) Step 2

After completing the first step, the baseline estimates for all pixels have been obtained, having baseline estimates for all pixels, and the image for the initial input has been acquired. The second step is the final estimation, where the baseline image is used for collaborative grouping and Wiener filtering. This process is similar to the first step but involves two groups formed from locations of blocks similar to the block being processed, one obtained from the original image and one with noise. Finally, the projected blocks are returned to their original placements, and the process of combining them is carried out by calculating the ultimate estimations for every pixel via averaging. Collaborative grouping and filtering are performed using Eq. (9). S_{xR}^{wie} *i*s group patch processing by Wiener, $\Upsilon_{xR}^{basic}$and Υ_{x}^{basic} is *xR* and *x*pixels estimation. N_1^{wie} is the block size that is used in Wiener grouping, and $\tau_{max-patch}^{wie}$ is the maximum threshold for the patch in the Wiener grouping.

$$S_{xR}^{wie} = \frac{||\Upsilon_{xR}^{basic} - \Upsilon_{x}^{basic}||_2^2}{(N_1^{wie})^2} < \tau_{max-patch}^{wie} \ \forall x \in X \quad (9)$$

For the Wiener shrinkage coefficients using Eq. (10), where $W_{S_{xR}^{wien}}$ is the Wiener shrinkage coefficient for the set N_{xR}^{wie}, and τ_{3D}^{wie} is the 3D transform used in Wiener grouping, and finally, σ^2 is noise variance.

$$W_{\left(S_{xR}^{wien}\right)} = \frac{|\tau_{3D}^{wie}(\Upsilon_{(S_{xR})}^{basic})|^2}{|\tau_{3D}^{wie}(\Upsilon_{(S_{xR})}^{basic})|^2 + \sigma^2} \quad (10)$$

To obtain the actual coefficients from the modified coefficients, inverse transformation is performed using the following Eq. (11), $\Upsilon_{S_{xR}^{wie}}^{wie}$ is the group of block-wise estimates.

$$\Upsilon_{\left(S_{xR}^{wie}\right)}^{wie} = \tau_{3D}^{(wie)^{-1}}\left(W_{\left(S_{xR}^{wie}\right)} \times \tau_{3D}^{wie}\left(Z_{(S_{xR}^{wie})}\right)\right) \quad (11)$$

In the aggregation of the second step, the weights to be used are calculated using Eq. (12), $W_{S_{xR}^{wie}}|\ |_2^{-2}$ is the best norm of the Wiener shrinkage coefficient.

$$W_{xR}^{wie} = \sigma^2 \times ||W_{(S_{xR}^{wie})}|\ |_2^{-2} \quad (12)$$

F. PSNR

While BM3D generally outperforms many denoising methods documented in the literature across various image scenarios, recent years have witnessed notable advancements in this field. Block Matching 3D (BM3D) has involved a series of significant innovations since its introduction in 2007 [19]. The original BM3D algorithm, which utilizes a block-matching and collaborative approach, has laid the groundwork for further development in image-denoising techniques. Since then, efforts have been made to improve the computational efficiency of BM3D, expand its capability to handle various types of noise encountered in digital images, and integrate deep learning techniques to enhance its performance [8, 19, 27, 28]. Some researchers aim to make BM3D more robust against variations in image content while providing user-friendly implementations for easy use by various users. With the continuous advancement of technology and research, BM3D remains a major focus in improving the quality of image denoising and continues to undergo innovation to meet new challenges and demands in this domain.

The ongoing innovation of the Block Matching 3D (BM3D) filter continues to enhance its performance in image-denoising tasks. Several notable areas of improvement include algorithm efficiency, noise reduction quality, adaptability to various types of noise, robustness to image variations, integration with deep learning, and user-friendly implementation [29, 30]. Efforts have been made to optimize the computational efficiency of BM3D, enabling faster processing without sacrificing the quality of denoising results. Many researchers are working to refine the denoising capabilities of BM3D to achieve better noise reduction while preserving image details and textures [27, 31]. BM3D has been adapted and optimized to handle various types of noise encountered in real-world scenarios, such as Gaussian noise, salt-and-pepper noise, or mixed noise types [30, 32].

Improvements aim to make BM3D more robust against variations in image content, resolution, and characteristics, ensuring consistent denoising performance across different types of images. Recent advancements have explored the integration of deep learning techniques with BM3D to further enhance its denoising capabilities. Efforts are made to provide user-friendly implementations of BM3D, offering easy-to-use software packages or libraries that allow researchers and practitioners to utilize the algorithm efficiently without requiring extensive expertise in image processing. Innovations in BM3D filtering also result in an improvement in the PSNR value. PSNR is a metric used to measure the quality of image restoration after denoising processes. A higher PSNR value indicates fewer distortions or errors occurring in the restoration process, thus indicating better restoration quality. Therefore, in the context of BM3D, improvements leading to an increase in PSNR values suggest that the denoising algorithm has effectively reduced noise without sacrificing important details in the image. Thus, in the development and improvement of BM3D, increasing PSNR is often considered an indicator of success in enhancing denoising quality.

III. Proposed Methods

A. 3D Processing in BM3D

Image processing in Three Dimensions (3D) is one of the crucial aspects in the development of modern image processing technology. One technique that has significantly improved performance in 3D image processing is the BM3D algorithm. BM3D is an innovative approach to addressing image processing problems,

particularly in the contexts of medical imaging, video processing, and image compression. By employing collaborative approaches and block processing, BM3D effectively reduces noise or disturbances present in 3D images, enabling clearer and more accurate image restoration. In the performance of BM3D, the utilization of 3D blocks within the image is crucial for the BM3D algorithm as it facilitates the identification of recurring patterns, thereby enhancing its ability to effectively remove noise. Additionally, BM3D can model the image structure more accurately, allowing the algorithm to distinguish between the original signal and noise, resulting in better denoising. This algorithm relies on two forms of correlation (intra-patch and inter-patch). To utilize both correlations, BM3D calculates the distance between each pair of patches nearby to find similar patches. Next, BM3D gathers a series of similar patches using a formula. The algorithm then combines all these patches with the currently processed patch, forming a 3D data structure called a group. With the formation of the 3D data structure, separate (2D + 1D) transformation selection is suitable for BM3D, allowing the selected transformation to leverage both intra-patch and inter-patch correlations.

B. Proposed Method

However, the algorithm's effectiveness heavily relies on the availability of a sufficient number of analogous patches. If BM3D cannot identify an ample number of similar patches, its performance will be constrained. For instance, if BM3D cannot locate eight or more akin patches for a given processed patch, the exploitation of inter-patch correlation will be suboptimal. In such scenarios, applying 3D transformation is unlikely to yield accurate estimations. Therefore, we suggest that if BM3D faces challenges in identifying eight or more analogous patches for a processed patch, it is preferable to employ 2D transformation rather than 3D. More precisely, Discrete Wavelet Transform 2D can be applied to the processed patch to achieve better results. In this study, several steps will be used to manage patches in the denoising process. First, the distance *d* from each reference patch to the patch being processed will be measured using Eq. (4). Then, a set of patches considered similar to each patch being processed will be formed. A reference patch will be deemed analogous to the patch being processed if "distance *d*" is lower than a predetermined threshold. Furthermore, the count of patches regarded as analogous to each patch being processed is also established. If the number of patches considered similar is <8, then a 3D data structure will not be created for those patches. Instead, 2D transformations will be applied to these patches. On the other hand, if there are eight or more patches considered similar, a 3D data structure will be created by arranging the similar patches with the patch being processed, and then a 3D transformation will be performed following the BM3D algorithm. With this approach, the denoising results are expected to be significantly improved in cases where patches do not have sufficient similarity. In this context, Z_x will linked as a patch, τ_{2D}^{ht} as 2D transformation2D, and $\Upsilon_x^{ht}$is the estimates obtained after the 2D transformation, so this procedure will use the following Eq. (13).

$$\Upsilon S_{xR}^{ht} = \tau_{2D}^{ht}(Z_x) \tag{13}$$

Next, a threshold will be applied to adjusted coefficients for 2D and 3D patches, and inverse transformation will be performed to obtain the values of the estimates. All estimates will be combined and calculated using Eq. (8). Upon completion of the initial step, a denoised image will be generated. Subsequently, the final image designated for experimentation will undergo the second phase of BM3D processing, culminating in the acquisition of the image for the ultimate denoising iteration. The steps performed for the experiments in this study are shown in Fig. 2. First, the noisy image will be taken, and then the distance *d* will be measured to determine the number of akin patches within each picture. If the number of patches in the image is ≤8, then a 2D transformation will be applied to the patch being processed, followed by thresholding on the modified coefficients and performing the inverse 2D transformation. However, if the count of analogous patches is >8, then transformation will be applied to the group, thresholding will be applied to the coefficients and inverse 3D transformation will be performed. After the transformation step, all estimates will be combined, including the calculation of the basic estimate and the estimate for each pixel. Finally, the second step of BM3D will be performed to obtain the final denoised image.

The proposed method integrates adaptive 2D and 3D transformations within the BM3D algorithm. The detailed steps are outlined in the following pseudocode in Algorithm 1:

Algorithm 1: BM3D with Adaptive 2D and 3D Transformations

Input: Noisy Image I, Threshold T

1. Initialize denoised image I' as a copy of I
2. For each block B in image I:
   a. Find similar blocks S using block-matching
   b. If |S| < T:
      i. Apply 2D transformation to B
      ii. Perform hard thresholding on 2D coefficients
      iii. Inverse 2D transformation to obtain denoised block B'
   c. Else
      i. Group B with similar blocks to form a 3D array
      ii. Apply 3D transformation to the 3D array
      iii. Perform hard thresholding on 3D coefficients
      iv. Inverse 3D transformation to obtain denoised block B'
   d. Aggregate B' into denoised image I'
3. Perform a second pass of BM3D processing on I':
   a. Use the denoised image I' from the first pass as the input
   b. Repeat steps 2a to 2d using Wiener filtering instead of hard thresholding
4. Return the denoised image I'

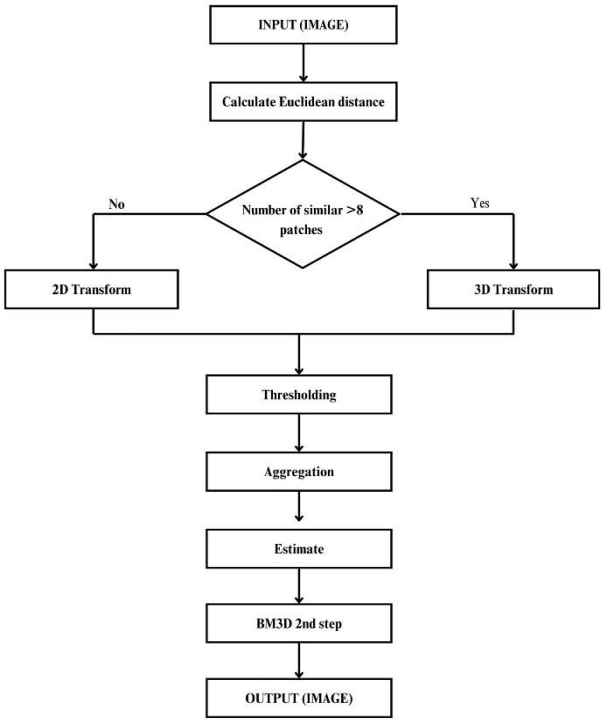

Fig. 2. The proposed model.

C. Dataset

The dataset used in this study is the Standard Test Image dataset from Kaggle *(Standard Test Image)* [33]. BM3D performs better when the input images are textured, as the algorithm can find many similar patches, facilitating collaborative filtering, for example, IMAGE-2 from the dataset used in this study (see Fig. 3). Conversely, when the images lack adequately similar patterns, BM3D's effectiveness diminishes. Hence, we've opted for this image dataset, encompassing both textured and non-textured images.

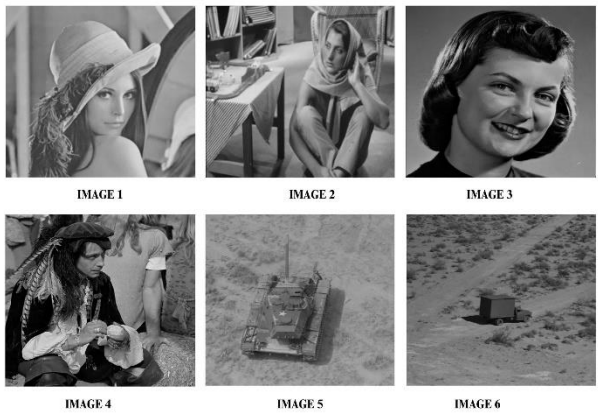

Fig. 3. Dataset of images selected [33] and used for experiments in C1 and C2.

D. Evaluation Metrics Performance

Assessing the effectiveness of the employed methodology for experiments compared to BM3D's performance, this research considers using PSNR as the performance metric. Additionally, the results were objectively evaluated and contrasted with outcomes obtained through the BM3D approach. PSNR is represented as follows:

$$PSNR = 10 \times \log 10 \left(\frac{MAX_1^2}{MSE} \right)$$

where $MSE = \frac{1}{M \times N} \sum_{i=1}^{M} \sum_{j=1}^{N} (y(i,j) Y(i,j))^2$ (14)

IV. Experiment Result

The experiments in this study are divided into two parts, namely C1 and C2. In C1, the experiment is conducted when the count of analogous patches detected is ≤2 for one patch, so only 2D transformations will be performed. Conversely, in C2, if the count of analogous patches detected is ≤4 for one patch, 2D transformation will be carried out, and vice versa. However, if more than 4 similar patches are found, 3D processing will be performed.

To further validate the effectiveness of the proposed method, additional experiments were conducted using various types of noise. The types of noise considered were Gaussian noise, Salt-and-Pepper noise, and Speckle noise.

A. Comparative Analysis

1) Gaussian noise

Gaussian noise is one of the most common types of noise found in digital images, often caused by camera sensors. This noise distribution is normal, affecting each pixel independently. Testing denoising algorithms against Gaussian noise is important because this type of noise frequently appears in many practical applications such as photography and medical imaging. Based on the findings presented in Tables I and II, it is evident that the achieved PSNR levels provide superior values at all noise levels. Notably, BM3D shows a significant decline in noise removal effectiveness for noise levels equal to or exceeding 40. This decline can be attributed to BM3D's difficulty in identifying analogous patches under high noise conditions, as confirmed by previous studies. Consequently, BM3D's inability to leverage intra-patch correlation becomes apparent in such scenarios.

Based on the findings presented in Tables I and II, it is evident that the attainment of PSNR levels yields superior values across all noise levels. Notably, BM3D demonstrates a significant decrease in its noise removal efficacy [19] for noise levels equal to or exceeding 40. This decline can be attributed to BM3D's difficulty in identifying analogous patches under high noise conditions, as corroborated by prior research [19]. Consequently, BM3D's inability to exploit intra-patch correlation becomes apparent in such scenarios [19]. For denoising these patches, it does not rely on similar patches due to their potentially significant dissimilarity. Therefore, noise removal is based solely on inter-patch correlation, resulting in superior denoising performance compared to the original BM3D method. Although there is a significant improvement at all noise levels, it is found that there is a slight increase in the performance of denoising in conditions of elevated noise levels.

TABLE I. EVALUATION OF PSNR BETWEEN BM3D AND EXPERIMENTAL METHOD IN C1

σ	Items	IMAGE 1	IMAGE 2	IMAGE 3	IMAGE 4	IMAGE 5	IMAGE 6	Average
10	BM3D	35.16	33.34	34.80	32.29	33.32	30.35	33.21
	Exp	35.46	33.88	35.18	32.79	34.00	31.04	33.73
	Imp	0.39	0.77	0.59	0.63	0.47	0.78	0.61
20	BM3D	31.17	29.44	30.84	28.81	29.61	26.60	29.41
	Exp	31.66	29.99	31.41	29.26	30.13	27.21	29.94
	Imp	0.58	0.64	0.66	0.54	0.61	0.70	0.62
30	BM3D	29.04	27.03	28.68	26.89	27.43	25.02	27.35
	Exp	29.70	27.97	29.16	27.38	28.11	25.32	27.94
	Imp	0.75	1.03	0.57	0.58	0.77	0.39	0.68
40	BM3D	27.54	25.46	26.92	25.64	25.93	24.01	25.92
	Exp	28.05	26.35	27.57	26.06	26.57	24.36	26.49
	Imp	0.60	0.98	0.74	0.51	0.73	0.44	0.67
50	BM3D	26.37	24.07	25.69	24.61	24.80	23.43	24.83
	Exp	26.93	25.09	26.13	25.00	25.55	23.68	25.40
	Imp	0.65	1.11	0.53	0.48	0.84	0.34	0.66
60	BM3D	25.18	23.09	24.59	23.79	23.90	22.96	23.92
	Exp	26.00	24.19	25.33	24.38	24.63	23.12	24.61
	Imp	0.91	1.19	0.83	0.68	0.82	0.25	0.78
70	BM3D	24.56	22.24	23.66	23.23	23.35	22.56	23.27
	Exp	25.17	23.27	24.59	23.91	23.92	22.74	23.93
	Imp	0.70	1.12	1.02	0.77	0.66	0.27	0.76
80	BM3D	24.10	21.74	23.20	22.63	22.71	22.15	22.76
	Exp	24.73	22.45	23.91	23.31	23.43	22.47	23.38
	Imp	0.72	0.80	0.80	0.77	0.81	0.41	0.72
90	BM3D	23.20	21.14	22.82	22.19	22.39	21.82	22.26
	Exp	24.02	21.97	23.51	22.60	23.01	22.20	22.89
	Imp	0.91	0.92	0.78	0.50	0.71	0.47	0.72
100	BM3D	22.69	20.69	21.84	21.61	21.69	21.55	21.68
	Exp	23.59	21.38	22.70	22.47	22.54	21.99	22.45
	Imp	0.99	0.78	0.95	0.95	0.94	0.53	0.86

Note: Exp: PSNR of the processed images generated by the experimental approach; Imp: Variation in PSNR between the processed images from the experimental approach and those from the BM3D application, measured in decibels (dB).

TABLE II. EVALUATION OF PSNR BETWEEN BM3D AND EXPERIMENTAL METHOD IN C2

σ	Items	IMAGE 1	IMAGE 5	IMAGE 4	IMAGE 2	IMAGE 3	IMAGE 6	Average
10	BM3D	35.16	33.32	33.29	33.34	34.80	30.35	33.38
	Exp	35.47	34.03	32.81	33.84	35.09	31.03	33.71
	Imp	0.40	0.80	0.61	0.59	0.38	0.77	0.59
20	BM3D	31.17	29.61	28.81	29.44	30.84	26.60	29.41
	Exp	31.78	30.12	29.27	30.07	31.48	27.07	29.97
	Imp	0.70	0.60	0.55	0.72	0.73	0.56	0.64
30	BM3D	29.04	27.43	26.89	27.03	28.68	25.02	27.35
	Exp	29.53	28.04	27.44	28.05	29.21	25.33	27.93
	Imp	0.58	0.70	0.64	1.11	0.62	0.40	0.68
40	BM3D	27.54	25.93	25.64	25.46	26.92	24.01	25.92
	Exp	27.99	26.70	26.02	26.43	27.51	24.36	26.50
	Imp	0.54	0.86	0.47	1.06	0.68	0.44	0.68
50	BM3D	26.37	24.80	24.61	24.07	25.69	23.43	24.83
	Exp	27.03	25.67	25.21	24.93	26.21	23.59	25.44
	Imp	0.75	0.96	0.69	0.95	0.61	0.25	0.70
60	BM3D	25.18	23.90	23.79	23.09	24.59	22.96	23.92
	Exp	25.95	24.57	24.40	24.09	25.38	23.17	24.59
	Imp	0.86	0.82	0.70	1.09	0.88	0.30	0.78
70	BM3D	24.56	23.35	23.23	22.24	23.66	22.56	23.27
	Exp	25.26	23.91	23.79	23.25	24.53	22.79	23.92
	Imp	0.79	0.65	0.65	1.10	0.96	0.32	0.75
80	BM3D	24.10	22.71	22.63	21.74	23.20	22.15	22.76
	Exp	24.55	23.31	23.23	22.61	23.87	22.47	23.34
	Imp	0.54	0.69	0.69	0.96	0.76	0.41	0.68
90	BM3D	23.20	22.39	22.19	21.14	22.82	21.82	22.26
	Exp	24.18	22.99	22.86	21.91	23.24	22.13	22.89
	Imp	1.07	0.69	0.76	0.86	0.51	0.40	0.72
100	BM3D	22.69	21.69	21.61	20.69	21.84	21.55	21.68
	Exp	23.54	22.53	22.20	21.48	22.58	21.92	22.38
	Imp	0.94	0.93	0.68	0.88	0.83	0.46	0.79

Note: Exp: PSNR of the processed images generated by the experimental approach; Imp: Variation in PSNR between the processed images from the experimental approach and those from the BM3D application, measured in decibels (dB).

2) Salt-and-pepper noise

Salt-and-pepper noise appears as black-and-white specks on an image and is usually caused by interference in data transmission or sensor errors. Testing algorithms against Salt-and-Pepper noise is important to ensure effectiveness in removing sharp impulsive noise without sacrificing important details. Experimental results show that the proposed method outperforms the original BM3D algorithm, especially at high noise levels. Tables III and IV summarize the PSNR values for Salt-and-Pepper noise. The proposed method achieves better denoising results by dynamically applying 2D transformation when not enough similar patches are found.

From Tables IV and V, it can be observed that the tested method yields shorter execution times compared to BM3D. On average, the method tested in C1 results in a reduction in execution time by 0.99%, and in C2, it results in a reduction in execution time by 1.08% compared to the original BM3D. Although the reduction is not significant, it indicates that the method tested in C1 achieves better noise removal performance than C2, especially at high noise levels. In the method tested in C2, if patches with ≤4 similar patches are found, 2D processing will be performed on those patches. This approach is anticipated to deliver superior noise elimination results compared to C1. Nevertheless, in scenarios where four analogous patches are identified, it may be advantageous to employ 3D processing instead of 2D processing, particularly at lower noise levels.

TABLE III. COMPARISON OF PSNR-DB BETWEEN BM3D AND THE METHOD UNDER SECURITY IN C1

σ	BM3D Avg. PSNR	Exp. Avg. PSNR in C1	Imp. in C1	Exp. Avg. PSNR in C2	Imp. in C2
10	33.04	33.58	0.63	33.59	0.64
20	29.3	29.83	0.62	29.86	0.65
30	27.26	27.84	0.67	27.85	0.68
40	25.87	26.42	0.64	26.44	0.66
50	24.81	25.36	0.64	25.38	0.66
60	23.94	24.59	0.74	24.59	0.74
70	23.28	23.92	0.73	23.91	0.72
80	22.77	23.4	0.72	23.37	0.69
90	22.28	22.9	0.71	22.91	0.72
100	21.71	22.48	0.86	22.43	0.81

TABLE IV. THE DIFFERENCE IN EXECUTION TIME BETWEEN BM3D AND THE METHOD TESTED IN SECOND AT C1

σ	Items	IMAGE 1	IMAGE 5	IMAGE 4	IMAGE 2	IMAGE 3	IMAGE 6	Average
10	BM3D	44.71	51.96	57.97	48.42	57.80	56.18	52.84
	Imp	0.39	0.89	0.72	0.53	0.69	0.68	0.70
	Exp	44.41	51,16	57.34	47.98	57.20	55.59	52.28
20	BM3D	44.64	59,41	47,01	48,20	56,33	51,63	51,20
	Exp	44.28	59.33	46.85	47.84	55.81	51.25	50.89
	Imp	0.45	0.17	0.25	0.45	0.61	0.47	0.40
30	BM3D	45.26	61.24	48.66	47.85	62.40	58.39	53.97
	Exp	44.61	60.86	48.19	46.69	62.54	58.21	53.52
	Imp	0.74	0.47	0.56	1.25	0.23	0.27	0.59
40	BM3D	56.35	61.27	55.57	62.28	62.01	58.88	59.39
	Exp	55.83	60.83	55.11	61.27	60.83	58.76	58.77
	Imp	0.61	0.53	0.55	1.10	1.27	0.21	0.71
50	BM3D	57.07	60.65	53.67	59.59	58.83	56.38	57.70
	Exp	57.84	59.55	53.58	58.40	58.14	56.16	57.28
	Imp	0.86	1.19	0.18	1.28	0.78	0.31	0.77
60	BM3D	53.79	56.95	55.82	54.60	52.22	58.59	55.33
	Exp	53.04	56.26	55.48	53.24	51.44	58.51	54.66
	Imp	0.84	0.78	0.43	1.45	0.87	0.17	0.76
70	BM3D	65.07	59.20	52.25	54.38	51.93	58.64	56.91
	Exp	64.17	58.75	51.78	53.34	50.87	58.65	56.26
	Imp	0.99	0.54	0.56	1.13	1.15	0.10	0.75
80	BM3D	65.20	56.88	50.81	57.25	53.38	54.41	56.32
	Exp	64.76	55.20	50.41	56.77	52.52	54.14	55.63
	Imp	0.53	1.77	0.49	0.57	0.95	0.36	0.78
90	BM3D	52.02	57.32	51.08	53.33	57.19	50.79	53.62
	Exp	51.17	56.87	50.56	52.65	56.58	50.51	53.06
	Imp	0.94	0.54	0.61	0.77	0.70	0.37	0.66
100	BM3D	50.26	57.09	48.81	62.51	51.46	58.27	54.73
	Exp	49.64	56.36	48.18	61.74	50.86	57.73	54.09
	Imp	0.71	0.82	0.72	0.86	0.69	0.63	0.74

Note: Exp: Processing time (seconds) of cleaned images produced by the experimental method in C2; Imp: Difference in processing time (seconds) between cleansed images yielded by the experimental approach and the BM3D algorithm.

3) Speckle noise

Speckle noise often appears in images produced by coherent imaging systems such as medical ultrasound and radar, caused by the random interference of reflected waves. Testing denoising algorithms against speckle noise is important for applications in the medical field and remote sensing, where high-quality images are crucial for accurate analysis and diagnosis.

Similar trends were observed with speckle noise, as shown in Tables V and IV. The adaptive approach improves denoisi1ng performance by considering the availability of similar blocks in the image. The proposed method shows an increase in PSNR values compared to the original BM3D algorithm.

TABLE V. THE DIFFERENCE IN EXECUTION TIME BETWEEN BM3D AND THE METHOD TESFTED IN SECOND AT C2

σ	Items	IMAGE 1	IMAGE 2	IMAGE 3	IMAGE 4	IMAGE 5	IMAGE 6	Average
10	BM3D	44.71	48.42	57.8	57.97	56.18	51.96	52.84
	Exp	44.3	47.97	57.2	57.31	55.57	51.14	52.25
	Imp	0.5	0.54	0.69	0.75	0.7	0.91	0.68
20	BM3D	44.64	48.2	56.33	47.01	51.63	59.41	51.20
	Exp	44.19	47.84	55.77	46.79	51.09	59.28	50.83
	Imp	0.54	0.45	0.65	0.31	0.63	0.22	0.47
30	BM3D	45.26	47.85	62.4	48.66	58.39	61.24	53.97
	Exp	44.6	46.64	62.36	48.18	58.17	60.86	53.47
	Imp	0.75	1.3	0.13	0.57	0.31	0.47	0.59
40	BM3D	56.35	62.28	62.01	55.57	58.88	61.27	59.39
	Exp	55.64	61.26	60.78	55.1	58.66	60.83	58.71
	Imp	0.8	1.11	1.32	0.56	0.31	0.53	0.77
50	BM3D	57.07	59.59	58.83	53.67	56.38	60.65	57.70
	Exp	57.83	58.39	58.13	53.5	56.08	59.55	57.25
	Imp	0.85	1.29	0.79	0.26	0.39	1.19	0.80
60	BM3D	53.79	54.6	52.22	55.82	58.59	56.95	55.33
	Exp	53	53.19	51.43	55.38	58.52	56.23	54.63
	Imp	0.88	1.5	0.88	0.53	0.16	0.81	0.79
70	BM3D	65.07	54.38	51.93	52.25	58.64	59.2	56.91
	Exp	64.17	53.27	50.79	51.71	58.55	58.72	56.20
	Imp	0.99	1.2	1.23	0.63	0.18	0.57	0.80
80	BM3D	65.2	57,25	53.38	50.81	54.41	56.88	56.32
	Exp	64.73	56,75	52.51	50.31	54.07	55.13	55.58
	Imp	0.56	0,59	0.96	0.59	0.43	1.84	0.83
90	BM3D	52.02	53.33	57.19	51.08	50.79	57.32	53.62
	Exp	51.12	52.57	56.51	50.5	50.5	56.81	53.00
	Imp	0.99	0.85	0.77	0.67	0.38	0.6	0.71
100	BM3D	50.26	62.51	51.46	48.81	58.27	57.09	54.73
	Exp	49.59	61.68	50.8	48.17	57.69	56,34	54.05
	Imp	0.76	0.92	0.75	0.73	0.67	0.84	0.78

Note: Exp: Processing time (seconds) of cleaned images produced by the experimental method in C2; Imp: Difference in processing time (seconds) between cleansed images yielded by the experimental approach and the BM3D algorithm.

TABLE VI. PERCENTAGE OF PATCHES EXHIBITING A RESEMBLANCE WITH 2 SIMILAR PATCHES

σ	IMAGE 1	IMAGE 2	IMAGE 3	IMAGE 4	IMAGE 5	IMAGE 6	Average
10	20.04	37.04	25.21	37.42	55.83	31.25	34.47
20	17.5	36.39	24.02	31.26	41.15	26.26	29.43
30	17.81	37.4	25.77	29	28.79	25.92	27.45
40	16.18	33.94	24.08	22.62	15.2	26.61	23.11
50	19.39	44.16	30.05	25.56	19.14	27.79	27.68
60	23.76	53.34	38.03	36.33	29.53	34.87	35.98
70	35.93	63.56	46.05	47.37	47.43	45.14	47.58
80	52.2	73.58	56.16	63.26	60.96	54.51	60.11
90	66.84	78.13	73.26	63.94	69.01	73.57	70.79
100	75.06	87.43	77.04	79.35	78.25	79.71	79.47

TABLE VII. PERCENTAGE OF PATCHES THAT HAVE 2 OR 4 SIMILAR PATCHES

σ	IMAGE 1	IMAGE 2	IMAGE 3	IMAGE 4	IMAGE 5	IMAGE 6	Average
10	20.74	37.25	25.33	37.93	55.84	31.45	34.76
20	17.64	36.75	24.21	31.31	41.5	26.48	29.65
30	18.27	37.42	25.97	29.04	29	26.34	27.67
40	16.25	34.25	24.32	22.89	15.24	26.87	23.30
50	19.88	44.71	30.5	26.26	19.51	28.27	28.19
60	24.26	53.74	38.43	36.92	29.72	35.44	36.42
70	36.15	63.92	46.41	47.77	47.79	45.42	47.91
80	52.48	73.74	56.48	63.51	61.38	54.96	60.43
90	67.01	78.48	73.69	64.16	67.24	73.71	70.72
100	75.4	87.85	77.4	79.83	78.35	80.06	79.82

The percentage of patches that have 2 similar patches, can be seen in Table VI, and conversely, the percentage of patches that have 2 or 4 similar patches can be seen in Table VII. From both tables, it can be concluded that the number of patches for higher noise levels also has a higher percentage. This is evidenced by the increasing Euclidean distance at elevated noise levels, rendering the identification of analogous patches challenging.

B. Visual Evaluation Analysis

Fig. 4 illustrates a visual contrast between the noise filtering efficacy of experimental method C1 and the

filtering performance of the conventional BM3D technique. Compared to Table VI, the cleaned images by BM3D and the cleaned images by experimental method C1 exhibit a significant difference, where the proposed experimental method achieves better visual performance compared to the original BM3D. In Fig. 4, several parts indicate that the proposed experimental method outperforms the original BM3D visually, as shown in the zoomed-in images (Fig. 4). The identical inference holds for Figs. 5 and 6 when juxtaposing the resultant images generated via the conventional BM3D approach and the novel method proposed in C2.

The findings from this study indicate that the proposed adaptive approach to the BM3D image denoising algorithm can significantly enhance denoising performance, especially at high noise levels. Experimental results show that applying 2D transformations to blocks that do not have enough similar blocks can yield better denoising results compared to universally applying 3D transformations, as in conventional BM3D approaches. This aligns with findings from the study [29], which stated that modifications to the BM3D algorithm can improve denoising efficiency. The study also shows that lower computation times can be achieved with the proposed approach, which adaptively combines 2D and 3D transformations. This contributes further to the development of more efficient image-denoising techniques. These results support the research by Yahya e*t al.* [27], which developed an adaptive filter-based BM3D algorithm to enhance denoising quality with higher computational efficiency.

In the context of high noise, conventional BM3D faces difficulties in finding similar blocks, making the exploitation of intra-patch correlation less optimal. This was also observed by Dabov *et al.* [19], who showed that BM3D performance decreases at high noise levels due to limitations in identifying similar blocks. This study reinforces those findings and demonstrates that applying 2D transformations to blocks that are not sufficiently similar can overcome these limitations and provide better denoising results. Additionally, visual analysis results show that the proposed experimental approach yields better visual quality compared to conventional BM3D. These findings are relevant to the research by Ma [29], which also found improved visual quality with a modified BM3D approach.

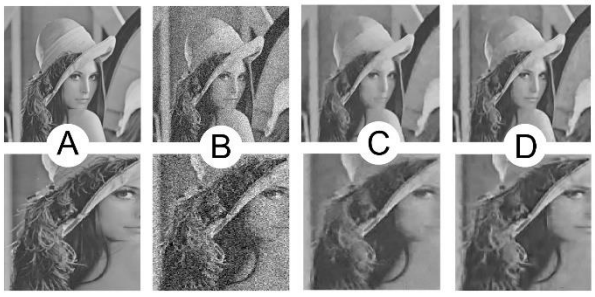

Fig. 4. Visual results obtained from experiments for the IMAGE-1 dataset (the top row shows the original experimental results while the bottom row is a zoom-in of the top row images). Image A (Original dataset used), B (corrupted by Gaussian noise) lv. 30, C (Cleaned result) using BM3D, and D (Cleaned result) using experimental method C2.

Fig. 5. Visual results obtained from experiments for the IMAGE-4 dataset (the top row shows the original experimental results while the bottom row is a zoom-in of the top row images). Image A (Original dataset used), B (corrupted by Gaussian noise) lv. 10, C (Cleaned result) using BM3D, and D (Cleaned result) using experimental method C1.

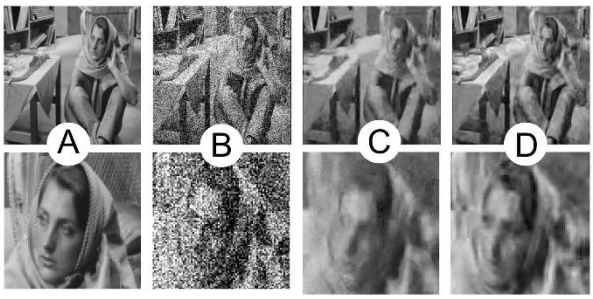

Fig. 6. Visual results obtained from experiments for the IMAGE-2 dataset (the top row shows the original experimental results while the bottom row is a zoom-in of the top row images). Image A (Original dataset used), B (corrupted by Gaussian noise) lv. 60, C (Cleaned result using BM3D), and D (Cleaned result using experimental method C1.

V. Conclusion

The results of this research indicate that the adaptively proposed approach in the BM3D image denoising algorithm can significantly improve denoising performance. Experimental results show that performing 2D transformations on blocks that do not have sufficiently similar blocks, instead of 3D transformations, can produce better denoising results, especially at high levels of noise. This suggests that considering the availability of similar blocks in the image when selecting the type of transformation can provide more optimal results in the denoising process. Additionally, this research also successfully demonstrates that the proposed method can achieve lower computational time compared to conventional approaches that use 3D transformations for all image blocks. This indicates that the adaptive use of 2D and 3D transformations not only improves the quality of denoising results but also the efficiency in image processing. Thus, this research contributes to developing more effective and efficient image-denoising techniques. The results of this research reaffirm the importance of considering the characteristics of image blocks in the denoising process to achieve optimal results. With the proposed approach, which is to perform 2D and 3D transformations adaptively based on the availability of similar blocks, the BM3D algorithm can improve its performance and produce better image denoising. This research provides new insights into the development of

image-denoising techniques and can serve as a foundation for further research in this field.

Several research recommendations can be considered and conducted to obtain further findings in optimizing the use of 2D and 3D transformations in image denoising processes, including exploring the use of hybrid transformations, evaluating algorithm performance on diverse datasets, and integrating machine learning techniques. Additionally, it is important to test algorithm performance on other types of noise such as salt-and-pepper noise, speckle noise, or other noise types, and to develop more complex adaptive algorithms that can dynamically adjust the type of transformation based on the context of the image blocks. Another recommendation is to implement the proposed algorithms in real-world applications such as medical image processing or image restoration so that further research can continue to develop more effective, efficient, and applicable image-denoising techniques in various image-processing contexts.

CONFLICT OF INTEREST

The authors declare no conflict of interest.

AUTHOR CONTRIBUTIONS

Conceptualization, AHB and MCW; Methodology, AHB and SW Software, BR; Formal Analysis: JST and SW; Investigation, SW; Resources, MCW and SW, Data Curation, AHB and SW; Writing—Original Draft Preparation, AHB and MCW, Writing—Review & Editing, AHBN, MCW and SW, Supervision, AHBN and SW, Project Administration, AHBN and MCW. All authors had approved the final version.

ACKNOWLEDGMENT

We would like to express our heartfelt gratitude to our colleagues, Dr. Mars Caroline Wibowo, and Dr. Sarwo Nugroho for their unwavering support and guidance throughout the research process. Their expertise, insights, and encouragement were invaluable in helping us to complete this work. We also want to thank our colleagues at STEKOM University Semarang Indonesia for their helpful feedback and support. Finally, we would like to thank our family and friends for their love and support throughout the research process. Without their encouragement and support, we would not have been able to complete this research.

REFERENCES

[1] S. V. Mohd Sagheer and S. N. George, "A review on medical image denoising algorithms," *Biomed Signal Process Control*, vol. 61, 102036, Aug. 2020. doi: 10.1016/j.bspc.2020.102036

[2] D. Verma, V. P. Vishwakarma, and S. Dalal, "A Hybrid Self-constrained Genetic Algorithm (HSGA) for digital image denoising based on PSNR improvement," in *Advances in Bioinformatics, Multimedia, and Electronics Circuits and Signals*, 2020, pp. 135–153. doi: 10.1007/978-981-15-0339-9_12

[3] K. Dutta, R. Lenka, and M. S. Sarowar, "Improvement of denoising in images using Generic Image Denoising Network (GID Net)," in *Proc. IEEE 2nd International Conference on Applied Electromagnetics, Signal Processing, and Communication, AESPC 2021*, 2021. doi: 10.1109/AESPC52704.2021.9708513

[4] K. Osvaldová, L. Gajdošech, V. Kocur, and M. Madaras, "Enhancement of 3D camera synthetic training data with noise models," in *Proc. the 27th Computer Vision Winter Workshop (CVWW)*, Feb. 2024. doi: 10.5281/zenodo.10694437

[5] H. Patel, D. Singh Rajput, G. Thippa Reddy, C. Iwendi, A. Kashif Bashir, and O. Jo, "A review on the classification of imbalanced data for wireless sensor networks," *Int. J. Distrib. Sens. Netw.*, vol. 16, no. 4, 155014772091640, Apr. 2020. doi: 10.1177/1550147720916404

[6] G. Ramesh, J. Logeshwaran, J. Gowri, and A. Mathew, "The management and reduction of digital noise in video image processing by using transmission based noise elimination scheme," *CTACT Journal on Image and Video Processing*, vol. 13, no. 1, pp. 2797–2801, 2022.

[7] T. Dhar, N. Dey, S. Borra, and R. S. Sherratt, "Challenges of deep learning in medical image analysis—Improving explainability and trust," *IEEE Transactions on Technology and Society*, vol. 4, no. 1, pp. 68–75, Mar. 2023. doi: 10.1109/TTS.2023.3234203

[8] L. Ju *et al.*, "Improving medical images classification with label noise using dual-uncertainty estimation," *IEEE Trans Med Imaging*, vol. 41, no. 6, pp. 1533–1546, Jun. 2022. doi: 10.1109/TMI.2022.3141425

[9] N. Pearl, T. Treibitz, and S. Korman, "NAN: Noise-aware NeRFs for burst-denoising," in *Proc. 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, Jun. 2022, pp. 12662–12671. doi: 10.1109/CVPR52688.2022.01234

[10] S. R. Karanam, Y. Srinivas, and S. Chakravarty, "A statistical model approach based on the gaussian mixture model for the diagnosis and classification of bone fractures," *Int. J. Healthc. Manag*, Jan. 2023. doi: 10.1080/20479700.2022.2161146

[11] W. Kong, J. Chen, Y. Song, Z. Fang, X. Yang, and H. Zhang, "Sobel edge detection algorithm with adaptive threshold based on improved genetic algorithm for image processing," *International Journal of Advanced Computer Science and Applications*, vol. 14, no. 2, 2023.

[12] J. Wang *et al.*, "Domain-adaptive denoising network for low-dose CT via noise estimation and transfer learning," *Med. Phys.*, vol. 50, no. 1, pp. 74–88, Jan. 2023. doi: 10.1002/MP.15952

[13] L. Wang, P. A. Fayolle, and A. G. Belyaev, "Reverse image filtering with clean and noisy filters," *Signal Image Video Process*, vol. 17, no. 2, pp. 333–341, Mar. 2023. doi: 10.1007/S11760-022-02236-W/FIGURES/8

[14] H. Shen, Z.-Q. Zhao, and W. Zhang, "Adaptive dynamic filtering network for image denoising," in *Proc. the AAAI Conference on Artificial Intelligence*, 2023, vol. 37, no. 2, pp. 2227–2235. doi: https://doi.org/10.1609/aaai.v37i2.25317

[15] L. Fan, F. Zhang, H. Fan, and C. Zhang, "Brief review of image denoising techniques," *Vis. Comput. Ind. Biomed. Art.*, vol. 2, no. 1, pp. 1–12, Dec. 2019. doi: 10.1186/S42492-019-0016-7/FIGURES/4

[16] M. Mafi, H. Martin, M. Cabrerizo, J. Andrian, A. Barreto, and M. Adjouadi, "A comprehensive survey on impulse and Gaussian denoising filters for digital images," *Signal Processing*, vol. 157, pp. 236–260, Apr. 2019. doi: 10.1016/j.sigpro.2018.12.006

[17] M. S. Priyadharsini and J. G. R. Sathiaseelan, "The new robust adaptive median filter for denoising cancer images using image processing techniques," *Indian J. Sci. Technol.*, vol. 16, no. 35, pp. 2813–2821, Sep. 2023. doi: 10.17485/IJST/v16i35.1024

[18] W. Huangfu *et al.*, "Automated extraction of mining-induced ground fissures using deep learning and object-based image classification," *Earth Surf. Process Landf.*, vol. 49, no. 7, pp. 2189–2204, Jun. 2024. doi: 10.1002/ESP.5824

[19] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3-D transform-domain collaborative filtering," *IEEE Transactions on Image Processing*, vol. 16, no. 8, pp. 2080–2095, Aug. 2007, doi: 10.1109/TIP.2007.901238

[20] E. Richard and R. C. G. Woods, *Digital Image Processing*, 3rd Ed., Pearson, Upper Saddle River, NJ 07458, 2021.

[21] A. AbdAlRahman, W. I. Al-Atabany, A. Soltan, and A. G. Radwan, "High-performance fractional anisotropic diffusion filter for portable applications," *J. Real Time Image Process*, vol. 20, no. 5, 98, Oct. 2023. doi: 10.1007/s11554-023-01339-y

[22] G. T. Vasu and P. Palanisamy, "CT and MRI multi-modal medical image fusion using weight-optimized anisotropic diffusion filtering," *Soft Comput.*, vol. 27, no. 13, pp. 9105–9117, Jul. 2023. doi: 10.1007/s00500-023-08419-y

[23] J. Weickert *et al.*, *Anisotropic Diffusion in Image Processing*, 1996.
[24] H. Kong, W. Gao, X. Du, and Y. Di, "An improved non-local means algorithm for CT image denoising," *Multimed. Syst.*, vol. 30, no. 2, 79, Apr. 2024. doi: 10.1007/s00530-024-01283-2
[25] Y. Liu, S. Duan, Z. Shen, Z. He, and L. Li, "Grasp and inspection of mechanical parts based on visual image recognition technology," *Journal of Theory and Practice of Engineering Science*, vol. 3, no. 12, pp. 22–28, Dec. 2023. doi: 10.53469/jtpes.2023.03(12).04
[26] Y. J. Sun, Y. W. Wang, F. Tang, Y. Geng, Y. Y. Xu, and M. Bu, "A two-stage synthetic Non-local Mean (NLM) filtering method for efficient denoising in Quantitative Phase Imaging (QPI)," *J. Mod. Opt.*, vol. 70, no. 7, pp. 406–418, Apr. 2023. doi: 10.1080/09500340.2023.2266057
[27] A. A. Yahya *et al.*, "BM3D image denoising algorithm based on an adaptive filtering," *Multimed. Tools Appl*, vol. 79, no. 27–28, pp. 20391–20427, Jul. 2020. doi: 10.1007/s11042-020-08815-8
[28] X. Yan, M. Xiao, W. Wang, Y. Li, and F. Zhang, "A self-guided deep learning technique for MRI image noise reduction," *Journal of Theory and Practice of Engineering Science*, vol. 17, no. 4, pp. 4970–4989, 2024.
[29] H. Ma, "BM3D denoising-based multi-target detection method for complex background radar images," *Applied Mathematics and Nonlinear Sciences*, vol. 9, no. 1, Jan. 2024. doi: 10.2478/amns.2023.1.00255
[30] G.-I. Ri, S.-J. Kim, and M.-S. Kim, "Improved BM3D method with modified block-matching and multi-scaled images," *Multimed. Tools Appl.*, vol. 81, no. 9, pp. 12661–12679, Apr. 2022. doi: 10.1007/s11042-022-12270-y
[31] Y. Zhang and J. Sun, "An improved BM3D algorithm based on anisotropic diffusion equation," *Mathematical Biosciences and Engineering*, vol. 17, no. 5, pp. 4970–4989, 2020. doi: 10.3934/mbe.2020269
[32] J. Cao, Z. Qiang, H. Lin, L. He, and F. Dai, "An improved BM3D algorithm based on image depth feature map and structural similarity block-matching," *Sensors*, vol. 23, no. 16, 7265, Aug. 2023. doi: 10.3390/s23167265
[33] S. Kamjoo. Standard Test Images. [Online]. Available: https://www.kaggle.com/datasets/saeedehkamjoo/standard-test-images