
MIXED MULTISCALE BM4D FOR THREE-DIMENSIONAL OPTICAL COHERENCE TOMOGRAPHY DENOISING

A PREPRINT

Ashkan Abbasi¹, Amirhassan Monadjemi², Leyuan Fang³, Hossein Rabbani⁴, Bhavna Josephine Antony^{5,6}, and Hiroshi Ishikawa^{1,7,*}

¹Department of Ophthalmology, Casey Eye Institute, Oregon Health & Science University, USA.

²School of Continuing and Lifelong Education, National University of Singapore, Singapore

³College of Electrical and Information Engineering, Hunan University, China.

⁴Department of Biomedical Engineering, Medical Image and Signal Processing Research Center, School of Advanced Technologies in Medicine, Isfahan University of Medical Sciences, Iran.

⁵Electrical and Computer System Engineering, Faculty of Engineering, Monash University, Australia.

⁶Department of Infectious Diseases, Alfred Health, Australia.

⁷Department of Medical Informatics and Clinical Epidemiology, Oregon Health & Science University, USA, (email: ishikawh@ohsu.edu).

*Corresponding author.

ABSTRACT

A multiscale extension for the well-known block matching and 4D filtering (BM4D) method is proposed by analyzing and extending the wavelet subbands denoising method in such a way that the proposed method avoids directly denoising detail subbands, which considerably simplifies the computations and makes the multiscale processing feasible in 3D. To this end, we first derive the multiscale construction method in 2D and propose multiscale extensions for three 2D natural image denoising methods. Then, the derivation is extended to 3D by proposing mixed multiscale BM4D (mmBM4D) for optical coherence tomography (OCT) image denoising. We tested mmBM4D on three public OCT datasets captured by various imaging devices. The experiments revealed that mmBM4D significantly outperforms its original counterpart and performs on par with the state-of-the-art OCT denoising methods. In terms of peak-signal-to-noise-ratio (PSNR), mmBM4D surpasses the original BM4D by more than 0.68 decibels over the first dataset. In the second and third datasets, significant improvements in the mean to standard deviation ratio, contrast to noise ratio, and equivalent number

of looks were achieved. Furthermore, on the downstream task of retinal layer segmentation, the layer quality preservation of the compared OCT denoising methods is evaluated.

Keywords Wavelets · Sparse Representations · BM4D · Multiscale Denoising · Optical Coherence Tomography

1 Introduction

Optical coherence tomography (OCT) is widely used to capture cross-sectional images of the retina. Due to the coherent nature of light, OCT images are contaminated with speckle noise [1, 2] which heavily degrades the image quality and complicates downstream tasks such as retinal layers [3, 4] and cysts [5, 6] analysis. Many promising attempts have been made over the years for denoising OCT images using local neighborhood processing [7], diffusion equations [8], total variations [9, 10], Bayesian methods [11–13], Nonlocal self-similarity (NSS) in both spatial [14] and transform [11, 12] domains, and sparsity of coefficients in fixed transform domains such as wavelets [15–19], curvelets [20], and contourlets [21].

Recent OCT denoising methods have mostly relied on data-driven approaches based on either dictionary learning [22–28] or deep learning [29–41]. However, adapting dictionary atoms substantially increases the inference time, and the need for learning from high signal-to-noise-ratio (SNR) images restricts the usability of a method, because capturing high SNR OCT images for supervised training is difficult and costly. Furthermore, learning-based methods are generally not adaptable to different levels of degradation [42]. Model-based optimization methods, on the other hand, are unsupervised and flexible [8–13], but they are typically computationally expensive due to the need for optimization during inference.

The well-known block matching and 4D filtering (BM4D) method [43] features exploiting sparsity in the transform domain, correlation between frames, and NSS priors at the same time without the need for learning from high SNR images and optimization during inference. Thus, BM4D is a fast 3D denoising method that was also used as a benchmark in several recent OCT denoising papers [10, 12, 25–27, 29, 44]. Nonetheless, BM4D was easily outperformed mostly due to the noticeable artifacts in its results.

In this paper, we show that the proposed multiscale extension for BM4D considerably reduces the artifacts, and almost all of the recent OCT denoising methods mentioned can be outperformed in terms of both visual quality and feature preservation. This is because BM4D is a patch-based (or local) method. When noise is weak, local modeling can achieve plausible results. However, as noise level increases, local image structures are substantially distorted by noise and denoising through local operations becomes difficult. Therefore, enlarging the effective modeling and denoising regions through multiscale processing can be used [45–54].

In order to develop a multiscale extension for BM4D, we first start our derivation in 2D, where we empirically analyze the subband denoising method (initially proposed by multiscale K-SVD (MS K-SVD) [51] for denoising natural images). By emphasizing the shortcomings of this approach, we use a simple fusion method based on the wavelet

subbands mixing [55, 56] to combine the results of single and multiscale methods. This method greatly suppresses undesirable artifacts and maintains the low-frequency content and main structure of the image. Nevertheless, the need for directly denoising detail subbands makes it hard to use with the most popular image denoisers, like BM4D, which are not developed for denoising detail subbands. Therefore, we propose a method in which there is no need to explicitly denoise detail subbands. We have tested the proposed method for constructing multiscale extensions for three 2D natural image denoisers. Finally, we extend our discussion to 3D and propose our multiscale extension for BM4D. All of the proposed multiscale extensions have been thoroughly tested on real-world datasets. Furthermore, we performed retinal layer segmentation experiments on the results of denoising methods to demonstrate the effectiveness of our proposed method in preserving layer structure in comparison to other compared methods. Fig. 1 is based on the results reported in Section 6.2. This figure shows that our proposed mixed multiscale extensions for BM4D achieve superior results both in terms of quality and retinal layer preservation.

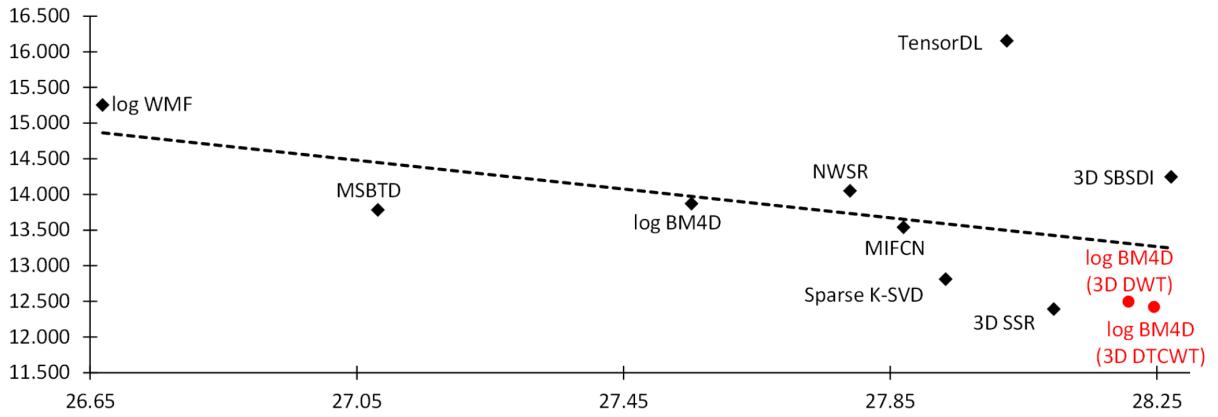


Figure 1: A graph illustrating the relationship between segmentation errors and image quality scores for the compared denoising methods. For each denoising method, retinal layer segmentation was performed on its results and the segmentation error was computed. The vertical axis represents the sum of absolute error (SAE) for segmentation, and the horizontal axis represents the mean of peak-signal-to-noise-ratio (PSNR). The dashed line shows the trend line. Our proposed methods are represented with red font and circles, and the compared methods are represented with black font and diamonds. This plot is drawn based on the results reported in Table 4 and 5 in Section 6.2.

2 Related Works

Tremendous efforts have been made over the years to build multiscale denoising methods from single-scale ones. A considerable amount of these efforts has been done with the aim of integrating multiscale analysis into data-driven sparse methods, which are intrinsically single-scale [28, 50, 51, 54, 57–60]. In [57, 58], the authors demonstrated how a wavelet basis can be adapted to the statistics of natural images and achieved slightly better coding efficiency and denoising. But these methods are fairly elaborate and computationally demanding, since they rely on sampling techniques to infer sparse codes. The work by [50] has extended K-SVD [22, 23] to simultaneously use different sizes of atoms, each size corresponding to a different scale. They have reported very promising image restoration results, though their method is extremely computationally demanding.

In a different approach, off-the-shelf single-scale denoising methods are directly applied in the multi-resolution analysis domain [28, 51, 54, 59, 60]. In this way, the denoising method acts as a complex shrinkage operator, and the frequency selectivity of subbands acts as a divide and conquer strategy, which could result in sparser representations and thus simpler denoising subproblems. In these studies, the multi-resolution analysis domain was mainly wavelets [61] or Laplacian pyramid [62]. The multiscale meta-procedure [45], multiscale EPLL (MSEPLL) [52], conservative scale recombination [46], and MSND [47] are some representative works in the Laplacian pyramid domain. The discrete cosine pyramid was also exploited in MS DCT [46, 53] with notable results.

Wavelets were also successfully used to construct multiscale methods from single-scale ones [51, 54, 59, 60]. In [59], a method based on filtering principal component coefficients of wavelet packet subbands was presented. One of the most notable wavelet based construction methods is multiscale K-SVD (MS K-SVD) [51]. In this method, the K-SVD image denoising [22] was directly used to filter all wavelet subbands (including the approximate subband). They reported that MS K-SVD was more powerful than pure K-SVD in recovering textures, edges, and main image structures. However, on average, MS K-SVD was not able to outperform its single-scale baseline. Later, Fused K-SVD [54] combined the results of single and multiscale methods using a joint sparse representation framework over a learned dictionary. Although the artifacts are greatly suppressed, the computational cost increases significantly since the joint sparse representations are obtained using concatenated input vectors and dictionaries.

In contrast to both MS K-SVD and Fused K-SVD, our proposed method (Section 4) avoids directly denoising detail subbands, saving a significant amount of computations. Moreover, it improves the construction method’s practical applicability by eliminating the need for capability of directly denoising detail subbands by the single-scale denoiser (or act as shrinkage operators [61]). As a result, it is more generalizable for proposing multiscale extensions for a wider range of single-scale denoising methods (Section 5).

3 Motivational Example

In this section, we briefly review the subband denoising method proposed by MS K-SVD [51] as a general multiscale construction method that can be used to create a multiscale method from a single-scale one. Then, we analyze its shortcomings in practice.

3.1 Constructing Multiscale Methods via Subband Denoising

The steps of the subband denoising method are listed in Algorithm 1. On the first line, it applies a J -scale wavelet transform (denoted by $\text{WT}(\cdot)$) to the input image \mathbf{Y} and decomposes it into a set of subbands $\mathcal{S} = \{\mathbf{A}^J, \mathcal{D}^J\}$ which includes the approximate subband at scale J (\mathbf{A}^J) and a set (\mathcal{D}^J) containing the detail subbands. Next, each subband is denoised via a single-scale denoising method (denoted by $\text{Dn}(\cdot)$) in line 4. Then, the denoised image ($\hat{\mathbf{X}}_M$) is reconstructed by computing the inverse wavelet transform ($\text{iWT}(\cdot)$) at line 6.

Algorithm 1 multiscale Denoiser (mDn) - The subband denoising method.

Input: \mathbf{Y} : Noisy image, σ_S : Noise standard deviation of the subband S , p : Patch size, and J : Number of scales.
Hyper-parameters: $Dn(\cdot)$: Single-scale denoiser, $WT(\cdot)$ and $iWT(\cdot)$: Wavelet transform and its inverse. The analysis and synthesis filter banks are denoted by af and sf , respectively.

```

1:  $\mathcal{S} = WT(\mathbf{Y}, J, af)$            //  $\mathcal{S}$  contains approximate and detail subbands:  $\{\mathbf{A}^J, \mathbf{D}^J\}$ 
2:  $\hat{\mathcal{S}} = \{\}$                    // An empty set is initialized.
3: for each subband ( $S \in \mathcal{S}$ ) do
4:    $\hat{S} = Dn(S, \sigma_S, p)$        // The single-scale denoising method denoises each subband.
5:    $\hat{\mathcal{S}} = \hat{\mathcal{S}} \cup \hat{S}$      // The Denoised subband will be added to the set.
6: Output:  $\hat{\mathbf{X}}_M = iWT(\hat{\mathcal{S}}, J, sf)$ 
```

Although the subband denoising method [51] or mDn (Algorithm 1) was originally realized by utilizing the K-SVD denoising method [22] and 2D discrete wavelet transform (DWT) [61], we emphasize here that mDn is general enough to be implemented with other 2D or 3D single-scale denoisers and wavelet transforms. In this way, mDn's single-scale denoiser ($Dn(\cdot)$), the type of J -scale wavelet transform ($WT(\cdot)$), and filter bank (af and sf) are all considered as hyper-parameters of the mDn method, and thus, by allowing to change these hyper-parameters, mDn has the potential to boost different single-scale denoising methods suitable for various types of images.

3.2 Subband Denoising in Practice

For practically evaluating mDn (Algorithm 1), it is required to realize it with a specific single-image denoiser ($Dn(\cdot)$) and wavelet transform ($WT(\cdot)$). In MS K-SVD [51], the K-SVD denoising method [22] and 2D DWT [61] with discrete Meyer filter bank were used in place of $Dn(\cdot)$ and $WT(\cdot)$, respectively. Here, we keep the wavelet transform and filter bank unchanged. But, we substitute the sparsifying transform learning denoising (TLD) [63] method in place of $Dn(\cdot)$ due to its simplicity and computational advantages over K-SVD, and to practically show that mDn is a general multiscale construction method. Note that the experimental results with K-SVD [51] will also be reported in Section 6.1.

By replacing TLD into mDn, we call the resultant method as mTLD. According to TLD's original papers [63–65], we set the patch size (p) equal to 11×11 pixels. And, we estimate the noise level (σ) from the noisy image using a noise estimation method [66]. The same noise level is used for denoising subbands of the image, i.e., $\sigma_S = \sigma$ (Algorithm 1, line 4). In Fig. 2, the outputs of TLD (c) and its multiscale version (mTLD) (d) for denoising a test image (b) are shown. We implemented mTLD with a 1-scale ($J=1$) DWT. The quality of denoised images is quantized using the Peak Signal-to-Noise-Ratio (PSNR) and structural similarity index (SSIM) [67].

Comparing Fig. 2 (c) and (d) clearly shows that although the main image structures (or low-frequency information) are better recovered through mTLD, the high frequency artifacts in mTLD's output negatively affect the result. TLD does not suffer from such artifacts, and it effectively removes high frequency noise, but it also removes the prominent image structures and degrades the low frequency content. Our experiments in Section 6.1 confirms that this does not happen accidentally, and implementing mDn (Algorithm 1) with other single-scale denoisers also leads to similar

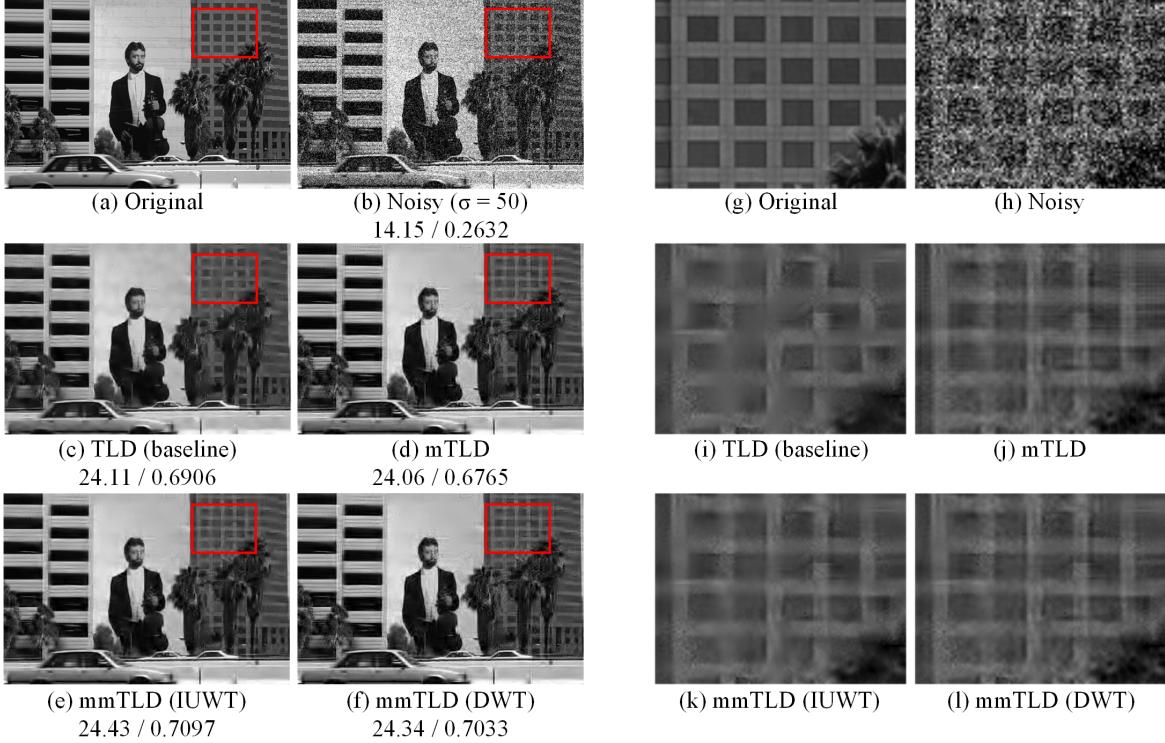


Figure 2: Visual comparison of the denoising results by a single-scale denoiser (here, TLD) and its multiscale extensions constructed with: 1. mDn (Algorithm 1) which is denoted by mTLD, and 2. our proposed mmDn which is denoted by mmTLD. Zoomed region for mTLD (j) shows substantial artifacts. These artifacts are mitigated in the results of mmTLD due to subbands mixing. Here, mmTLD is implemented using two different types of wavelet transform at its mixing stage. The transform type is mentioned in the parenthesis. For each method, the PSNR and SSIM results are reported below.

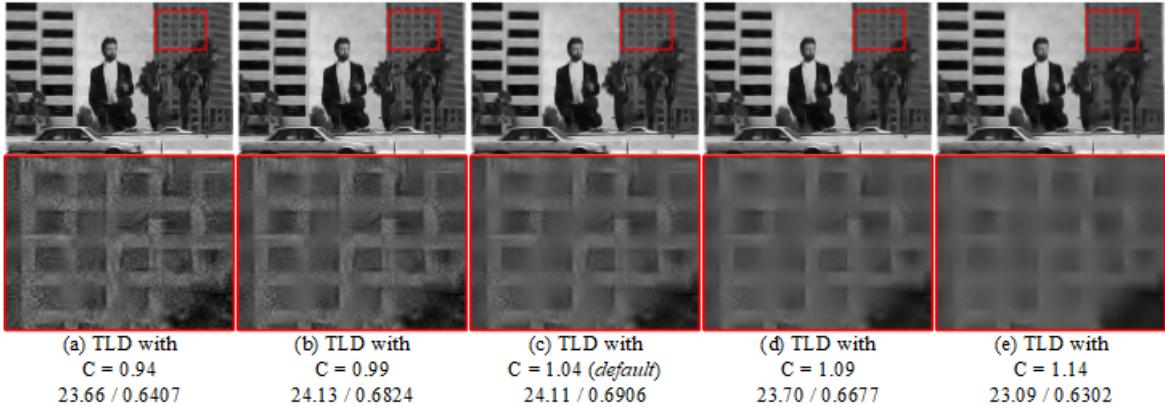


Figure 3: Effects of using different sparsity levels (or error thresholds) on the output of TLD. The C constant modifies the sparsity level in TLD [63–65]. By increasing this constant, the representations in TLD become sparser, and, therefore, noise suppression becomes stronger. For each method, the PSNR and SSIM are reported below.

results. The same observation has been also reported in [54] by practically comparing the MS K-SVD [51] pure K-SVD denoising [22] methods.

In order to restore the damaged structures in Fig. 2 (c), a reader may suggest that we can carefully control the amount of denoising in TLD. This can be done by controlling the sparsity level used for recovering patches in TLD, because it has a direct impact on the amount of denoising. In TLD [63–65], the algorithm updates the sparse representations until the error $\|Y - \hat{U}\|$ for recovering noisy patches Y by \hat{U} falls below a given threshold $nC^2\sigma^2$, where C is a fixed constant, n is the dimension of patches (here, $n = 11 \times 11 = 121$) and σ is the standard deviation of noise. Therefore, C is considered as a hyper-parameter which controls the sparsity of representations. To visually inspect the effects of different C values, we change it with steps of 0.05 in Fig. 3. Comparing Fig. 3 (b) and (c), it can be seen that although the default value of C is reasonable for this specific image, when we set C to 0.99 (Fig. 3 (b)), PSNR maximizes, and the damaged structures are better recovered. However, note that this is achieved at the expense of decreasing SSIM and tolerating more artifacts. It turns out that even with this careful hyper-parameter adjustment, TLD cannot efficiently recover the damaged parts without sacrificing quality.

A number of reasons can be thought of for explaining why mTLD generally leads to inferior results than TLD: First, denoising a subband is not an easy task due to its low signal to noise ratio [52]. This is why a lot of efforts have been made over the years to introduce more effective wavelet shrinkage methods. Second, to denoise subbands using TLD, careful parameter adjustment for each subband might be required, which is quite a cumbersome procedure. In the following section, we present our proposed methods to mitigate mDn’s shortcomings. Our final method avoids directly denoising subbands while it holds the advantages of mDn in preserving low-frequency content.

4 Proposed Method

4.1 Mixed Multiscale Denoiser (mmDn): Initial Version

Using wavelet subbands mixing technique [55, 56], we can combine the merits of both single-scale and multiscale denoisers: (i) The main image structure and low frequency content of multiscale denoiser’s output, and (ii) The high frequency content of single-scale denoiser’s output. We call the resultant method the mixed multiscale denoiser (mmDn) method.

The main steps of the initial version of our proposed mmDn method are listed in Algorithm 2. The input image is denoised first with the single-scale denoiser ($Dn(\cdot)$ at line 1) and second with the multiscale denoiser ($mDn(\cdot)$ at line 2). Their outputs are then separately decomposed using a K -scale wavelet transform (denoted by $WT_X(\cdot)$). Next, the wavelet subbands mixing is applied (in line 7) to create a new set ($\hat{\mathcal{S}}_X$) which comprises of the approximate subband ($\hat{\mathbf{A}}_M^K$) of $mDn(\cdot)$ and detail subbands ($\hat{\mathcal{D}}_S^K$) of $Dn(\cdot)$. Finally, the inverse wavelet transform is computed on ($\hat{\mathcal{S}}_X$) to reconstruct the final denoised image ($\hat{\mathbf{X}}_{MM0}$).

In order to analyze mmDn-0 (Algorithm 2) in practice, similar to mDn (Algorithm 1), we need to realize mmDn-0 by specifically determining its hyper-parameters. Here, we again replace $Dn(\cdot)$ with TLD as its single-scale denoiser. The multiscale denoiser ($mDn(\cdot)$) is realized as we have described in Section 3.2, and we use 2D DWT in place of its J -scale wavelet transform ($WT(\cdot)$). Therefore, no hyper-parameters are left in mmDn except the type of K -scale

Algorithm 2 Initial Version of the Mixed Multiscale Denoiser (mmDn-0)

Input: Noisy image (\mathbf{Y}) and its noise standard deviation (σ), Patch size (p), Number of scales (J), Number of scales (K) for the mixing stage.

Hyper-parameters: Hyper-parameters of Algorithm 1, Type of wavelet transform (WT_X) and its filter bank (af and sf).

- 1: $\hat{\mathbf{X}}_S = Dn(\mathbf{Y}, \sigma, p)$ // $Dn(\cdot)$: Single-scale denoiser
- 2: $\hat{\mathbf{X}}_M = mDn(\mathbf{Y}, \sigma, p, J)$ // $mDn(\cdot)$: Multiscale denoiser (Algorithm 1)
- 3: $\hat{\mathcal{S}}_S = WT_X(\hat{\mathbf{X}}_S, K, af)$ // $\hat{\mathcal{S}}_S$ will be $\{\hat{\mathbf{A}}_S^K, \hat{\mathcal{D}}_S^K\}$
- 4: $\hat{\mathcal{D}}_S^K =$ Get the detail subbands from $\hat{\mathcal{S}}_S$
- 5: $\hat{\mathcal{S}}_M = WT_X(\hat{\mathbf{X}}_M, K, af)$ // $\hat{\mathcal{S}}_M$ will be $\{\hat{\mathbf{A}}_M^K, \hat{\mathcal{D}}_M^K\}$
- 6: $\hat{\mathbf{A}}_M^K =$ Get the approximate subband from $\hat{\mathcal{S}}_M$
- 7: $\hat{\mathcal{S}}_X = \{\hat{\mathbf{A}}_M^K, \hat{\mathcal{D}}_S^K\}$
- 8: **Output:** $\hat{\mathbf{X}}_{MM0} = iWT_X(\hat{\mathcal{S}}_X, K, sf)$

wavelet transform ($WT_X(\cdot)$) used for subbands mixing. This transform is completely independent of the wavelet transform used in ($mDn(\cdot)$), and we are free to select the appropriate transform for each stage. To show this, we use two different types of wavelet transforms in place of $WT_X(\cdot)$. Firstly, we realize $WT_X(\cdot)$ with a 1-scale ($K=1$) isotropic undecimated wavelet transform (IUWT) with an Astro filter bank [68]. Secondly, we realize $WT_X(\cdot)$ with a 1-scale ($K=1$) 2D DWT. Since we are using mmDn to construct multiscale extensions of TLD, we call the resultant method mmTLD. The outputs of mmTLD with these transforms are shown in Fig. 2 (e) and (f), respectively. It can be seen that, contrary to the realization of mDn (Fig. 2 (d)), denoted by mTLD, our proposed mmDn based methods do not suffer from high frequency artifacts while they are able to preserve main structure of the image. The PSNR and SSIM also reflect the superiority of our proposed methods.

To demonstrate why the mixing stage of mmTLD works in practice, consider its mixing stage when a 1-scale IUWT is used as $WT_X(\cdot)$. Then, in line 3, the result of the single-scale denoiser (Fig. 2 (c)) is decomposed, and then in line 3, the result of the multiscale denoiser (Fig. 2 (d)) is decomposed. In each level, IUWT decomposes its input into one approximate and one detail subbands. Here, we use 1-scale decomposition. Therefore, the resultant subbands can be easily shown in Fig. 4. In this figure, the first two images illustrate the subbands obtained from TLD's output (denoted by $\hat{\mathcal{S}}_S$ in Algorithm 2) and the second two images show the subbands of MTLD's output ($\hat{\mathcal{S}}_M$). This figure clearly shows that the low frequency content in the approximate subband of MTLD (c) is much richer than the corresponding subband of TLD (a). Conversely, the detail subband of MTLD (d) is noisier than the corresponding subband of TLD (b). Therefore, line 7 of mmDn-0 (Algorithm 2) takes the strength of TLD and mTLD (Fig. 4 (b) and (c)) and discards their weaknesses (Fig. 4 (a) and (d)) to produce the final output (Fig. 2 (e)).

4.2 Removing the Need to Directly Denoise Detail Subbands and Increasing the Number of Scales

This section will address two major flaws in our initial proposed mmDn (denoted by mmDn-0 in Algorithm 2). First, similar to mDn, our mmDn-0 must directly denoise detail subbands. This is because mmDn-0 includes mDn as one of its main steps. Second, if more than one decomposition level is used in the mixing stage (i.e., $K>1$ in Algorithm 2), the output quality approaches that of the single-scale denoiser. This is due to the fact that mmDn-0 only employs the

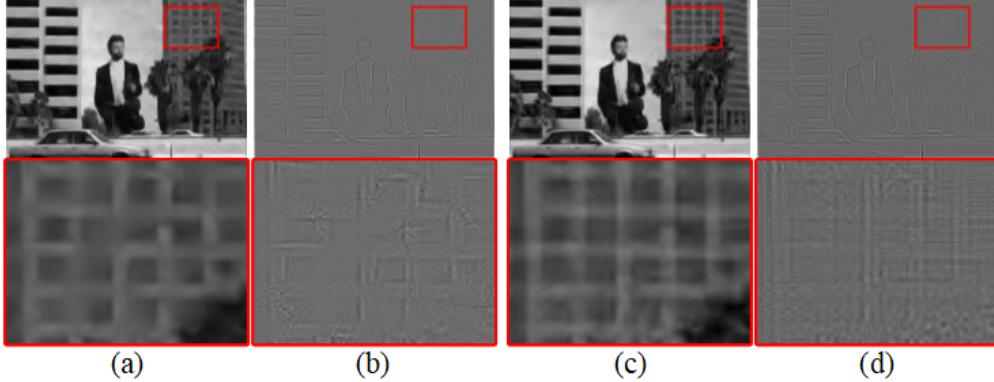


Figure 4: The subbands obtained by using 1-scale ($K=1$) IUWT decompositions on TLD and MTLD outputs are depicted. In the first two images, the TLD subbands are shown: (a) the approximate subband ($\hat{\mathbf{A}}_S^K$), and (b) the detail subband ($\hat{\mathcal{D}}_S^K$). In the last two images, the subbands of MTLD's output are shown: (c) the approximate subband ($\hat{\mathbf{A}}_M^K$), and (d) the detail subband ($\hat{\mathcal{D}}_M^K$).

last approximate subband of the multi-scale denoiser ($mDn(\cdot)$). Therefore, by increasing the number of scales, the contribution of the multiscale denoiser decreases substantially.

As mentioned in the previous section, mmDn-0 (Algorithm 2) employs two independent wavelet transforms: 1) $WT(\cdot)$ is used for denoising stage inside mDn (line 2), and 2) $WT_X(\cdot)$ is used for subbands mixing. There might be some benefits if we are free to select these transforms. For example, Fig. 2 shows that although both the realization of mmDn-0 (mmTLD with IUWT (e), and with DWT (f)) use DWT for their denoising stages, exploiting IUWT for the subbands mixing stage enhances the output quality in terms of both PSNR and SSIM. However, when these two transforms are selected to be the same, a considerable amount of computations become redundant, and, more importantly, a way to avoid directly denoising detail subbands is provided.

In mmDn-0 (Algorithm 2) subbands of mDn 's output ($\hat{\mathbf{X}}_M$) are extracted in line 5. When both wavelet transforms ($WT(\cdot)$ and $WT_X(\cdot)$) are identical, the $\hat{\mathcal{S}}$ in mDn and $\hat{\mathcal{S}}_M$ in mmDn-0 becomes equal. As a result, the inverse wavelet transform at the end of mDn is no longer necessary, and we can use $\hat{\mathcal{S}}_M = \hat{\mathcal{S}}$ instead. Furthermore, only the approximate subband from $\hat{\mathcal{S}}$ is used in mmDn-0. This implies that there is no need to denoise the set of detail subbands ($\hat{\mathcal{D}}_M^K$) inside mDn , and we can only pass the last denoised approximate subband to mmDn-0. Therefore, using the same wavelet transforms for both denoising and mixing stages, we can summarize the steps of mDn as follows: 1. Denoise the noisy input image \mathbf{Y} , and get its set of detail subbands ($\hat{\mathcal{D}}_S^J$) at scale J , 2. Get the J -th approximate subband ($\hat{\mathbf{A}}^J$) of the noisy image, and denoise it ($\hat{\mathbf{A}}^J$), and 3. Reconstruct the image by computing the inverse wavelet transform on $\hat{\mathcal{S}}_X = \{\hat{\mathbf{A}}^J, \hat{\mathcal{D}}_S^J\}$. In this way, the forward and inverse transforms for subbands mixing are not required anymore.

The second issue with mmDn-0 (Algorithm 2) happens when the number of scales increases. Naturally, we expect that by increasing the number of scales (to some extent), more details can be recovered. However, in practice, the opposite happens for mmDn-0. In Fig. 5, it can be seen that using 4-scale transforms to implement mmTLD-0 significantly reduces the effectiveness of this method in comparison with mTLD which realizes mDn (Algorithm 1).

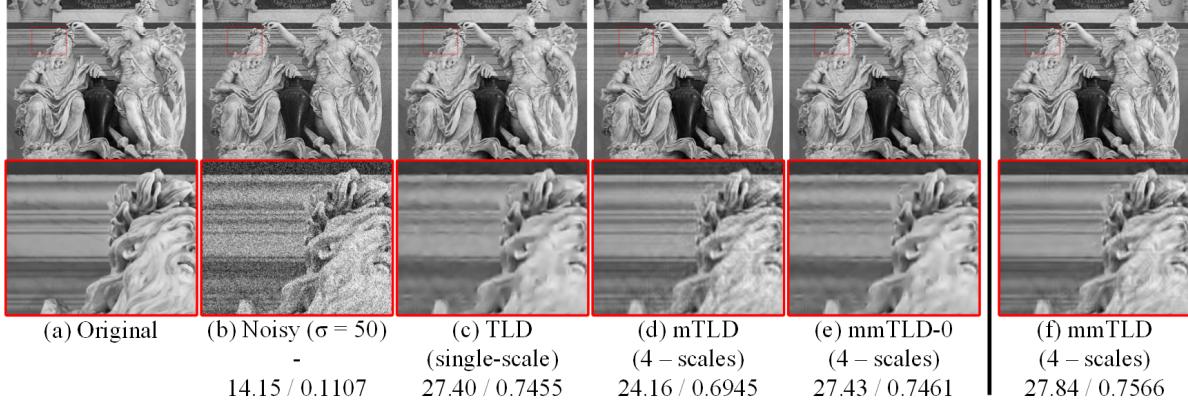


Figure 5: Effect of increasing the number of scales on the performance of multiscale extensions. A noisy image, and the results of a single-scale denoiser (TLD), and its multiscale extensions based on mDn (mTLD), mmDn-0 (mmTLD-0), and mmDn (mmTLD) are shown. TLD (c) substantially damages the low frequency content of the image. mTLD (d) preserves low frequency content, but it suffers from artifacts. We expect that mmTLD-0 (e) will behave like mTLD because it essentially combines the high frequency content of TLD and low frequency content of mTLD. However, since it borrows the last approximate subband from mTLD, the borrowed content from TLD dominates the final result when the number of decomposition scales increases in the mixing stage. This weakness is addressed in our final version of the mmDn method, which is implemented here as mmTLD (f). The original image (a) is a high-resolution image (with spatial size of 2040×1740 pixels) from the validation set of DIV2K dataset [69, 70]. The image (a) is corrupted in (b) by adding a white Gaussian noise with $\sigma = 50$. For each method, the PSNR and SSIM results are reported below.

This is because mDn effectively uses the information of all decomposition stages. But, mmDn-0 only uses the last approximate subband. Therefore, increasing the number of scales borrows more information from the single-scale denoiser, decreasing the performance and shifting the overall quality of our proposed mmDn-0 method toward the single-scale denoiser. This argument can be validated by comparing the quality metric values for TLD and mmTLD-0 results in Fig. 5 (c) and (e).

To fix this issue for mmDn-0, the detail subbands should be processed separately, similar to mDn. However, in contrast to mDn, we want to indirectly denoise detail subbands. This problem can be solved effectively by storing all approximate subbands across all levels of the decomposition scales. Then, we can extract the set of detail subbands at each scale by performing a 1-scale wavelet transform at that scale. This allows for simultaneous denoising and subband mixing across all decomposition levels. We list the steps of our final version of the proposed mmDn method in Algorithm 3. Firstly, it stores the noisy image (the finest scale) and all of its approximate subbands in an array (denoted by AP in Algorithm 3). Secondly, the method starts with the last two approximate subbands ($AP[s - 1]$ and $AP[s]$) and denoises them (in lines 7 to 8). Thirdly, the method performs the subbands mixing (line 13). It reconstructs the denoised approximate subband at the finer scale ($AP[s - 1]$) by changing its approximate subband with $AP[s]$ (coarser scale). Finally, these steps are repeated across all scales to reconstruct the final image. It is easy to see that when the number of scales is set to 1 ($J = 1$), mmDn (Algorithm 3) is exactly equivalent to mmDn-0 (Algorithm 2). However, when the number of scales increases, mmDn better preserves the advantages of the multiscale denoising. Fig. 5 (e) and (f) compares the denoising qualities of these two methods for their realizations when $Dn(\cdot) = TLD$ and $WT(\cdot) = DWT$ with 4 decomposition scales.

Algorithm 3 Final Version of the Mixed Multiscale Denoiser (mmDn)

Input: Noisy image (\mathbf{Y}) and its noise standard deviation (σ), Patch size (p), Number of scales (J).

Hyper-parameters: Hyper-parameters of Algorithm 1.

```

1: AP[0] = { $\mathbf{Y}$ }                                // AP is an array to store the approximate subbands.
2: for s = 1 to  $J$  do
3:    $\mathcal{S} = \text{WT}(\text{AP}[s - 1], 1, af)$            // 1-scale wavelet transform
4:    $\mathbf{A} = \text{Get the approximate subband from } \mathcal{S}$ 
5:   AP[s] = { $\mathbf{A}$ }
6: for s =  $J$  to 1 do
7:   AP[s - 1] =  $\text{Dn}(\text{AP}[s - 1], \sigma, p)$       // Denoising finer scale
8:   AP[s] =  $\text{Dn}(\text{AP}[s], \sigma, p)$                 // Denoising coarser scale
9:    $\mathcal{S}_F = \text{WT}(\text{AP}[s - 1], 1, af)$ 
10:   $\mathcal{D}_F = \text{Get the detail subbands from } \mathcal{S}_F$ 
11:   $\mathcal{S}_C = \text{WT}(\text{AP}[s], 1, af)$ 
12:   $\mathbf{A}_C = \text{Get the approximate subband from } \mathcal{S}_C$ 
13:   $S_X = \{\mathbf{A}_C, \mathcal{D}_F\}$ 
14:  AP[s - 1] =  $\text{iWT}(\mathcal{S}_X, 1, sf)$ 
15: Output:  $\hat{\mathbf{X}}_{MM} = \text{AP}[0]$ 
```

5 Generalization

Until now, we have discussed two methods for constructing multiscale extensions of a given single-scale denoiser:

1. mDn (Algorithm 1) [51], and 2. our proposed mmDn (Algorithm 3). In the preceding sections, we limited the single-scale denoiser ($\text{Dn}()$) to the TLD [63] and mostly used the 2D DWT [61]. However, as we have mentioned earlier, the mDn (Algorithm 1) and mmDn (Algorithm 3) methods are quite general and we can realize them with other single-scale denoisers and wavelet transforms. For example, mDn was previously realized by K-SVD denoising method [22], and they called it MS K-SVD [51]. In this section, we further generalize mDn and our proposed mmDn to introduce multiscale extensions for two other 2D image denoising methods. 1. K-SVD denoising [22], and 2. Spatially adaptive iterative singular-value thresholding (SAIST) [71]. Moreover, we generalize our discussion from 2D to 3D and introduce an efficient multiscale extension for BM4D in Section 5.1. The summary of all multiscale denoising methods are reported in Table 1. The thorough evaluation and experimental results are presented in Section 6.

5.1 Mixed Multiscale BM4D (mmBM4D)

In this section, we generalize our discussion from 2D to 3D, and apply our proposed mmDn (Algorithm 3) method to develop a multiscale extension for the well-known BM4D method in order to use it for 3D OCT image denoising. Note that a multiscale extension based on mDn (Algorithm 1, which was initially proposed by [51]) cannot be developed easily for BM4D, since it is not easy to tune BM4D for directly denoising detail subbands (which contain only high frequency content). Unlike mDn, our proposed mmDn avoids directly denoising detail subbands. Therefore, BM4D can be naturally used in mmDn to denoise the approximate subbands at different decomposition levels.

We will use mmDn (Algorithm 3) for constructing a multiscale extension for BM4D and we call the resultant method as mmBM4D. It is worth mentioning that BM4D was previously used multiple times to denoise 3D OCT images [10, 12, 25–27, 29, 44]. In these works, researchers have used this method in the spatial domain. However, we show

Table 1: Summary of all multiscale methods developed for the 2D or 3D single-scale denoising methods used in this study. In the last column, the components to realize each multiscale method are briefly mentioned, which includes the multiscale construction method (either mDn (Algorithm 1 [51]) or our proposed mmDn (Algorithm 3)) along with assigned single-scale denoiser ($Dn(\cdot)$) and wavelet transform (WT).

Single-scale Method	Multiscale Method	Multiscale Method Components
TLD [63]	mTLD	mDn is realized with $Dn(\cdot) = TLD$ and $WT = 2D$ DWT.
TLD [63]	mmTLD	mmDn is realized with $Dn(\cdot) = TLD$ and $WT = 2D$ DWT.
K-SVD denoising [22]	mK-SVD	mDn is realized with $Dn(\cdot) =$ the K-SVD denoising method and $WT = 2D$ DWT. Note that this method is called MS K-SVD in [51].
K-SVD denoising [22]	mmK-SVD	mmDn is realized with $Dn(\cdot) =$ the K-SVD denoising method and $WT = 2D$ DWT.
SAIST [71]	mSAIST	mDn is realized with $Dn(\cdot) = SAIST$ and $WT = 2D$ DWT.
SAIST [71]	mmSAIST	mmDn is realized with $Dn(\cdot) = SAIST$ and $WT = 2D$ DWT.
BM4D [43]	mmBM4D	mmDn is realized with $Dn(\cdot) = BM4D$ and $WT = 3D$ wavelet transform (e.g., 3D DWT). Detailed description of this method is presented in Section 5.1.

here that the performance of BM4D may greatly be enhanced for OCT denoising if we use it in the logarithm domain. Exploiting the logarithm domain to convert the multiplicative to additive noise is a common practice for OCT images, especially for methods based on fixed transform domains such as wavelets [17]. Therefore, we here present our results both in the spatial and logarithm domains.

In Fig. 6 (a) and (b), one slice of a noisy 3D OCT image is shown along with its corresponding registered and averaged image. The averaged image was obtained by a dedicated and costly imaging procedure. This image can be considered as the ground-truth [25, 26, 28]. The rest of images in the left column ((c), (e), and (g)) show the results of applying BM4D and its multiscale realizations in the spatial domain. Note that, here, we realize mmBM4D with two types of 3D wavelet transforms: 1) 3D DWT [61], and 2) 3D dualtree discrete complex wavelet transform (3D DTCWT) [72]. The first one is denoted by mmBM4D (3D DWT) and the second one is denoted by mmBM4D (3D DTCWT). The right column ((d), (f), and (h)) shows the results of each method in the logarithm domain. As we have mentioned earlier, in contrast to the results reported in [10, 12, 25–27, 29, 44], it can be seen that exploiting the logarithm domain can enhance the denoising results substantially. We will report more results and discussions in the upcoming section.

6 Experimental Results

In this section, we present the qualitative and quantitative results of the methods discussed in the previous section. The compared methods, datasets, and parameters are thoroughly presented in the following subsections. The source code for reproducing OCT denoising experiments in MATLAB® is publicly available at <https://github.com/ashkanabbasi66/OCT-denoising-package>. All experiments were conducted on a desktop PC with an Intel® i5-7400 CPU and 16 GB of RAM.

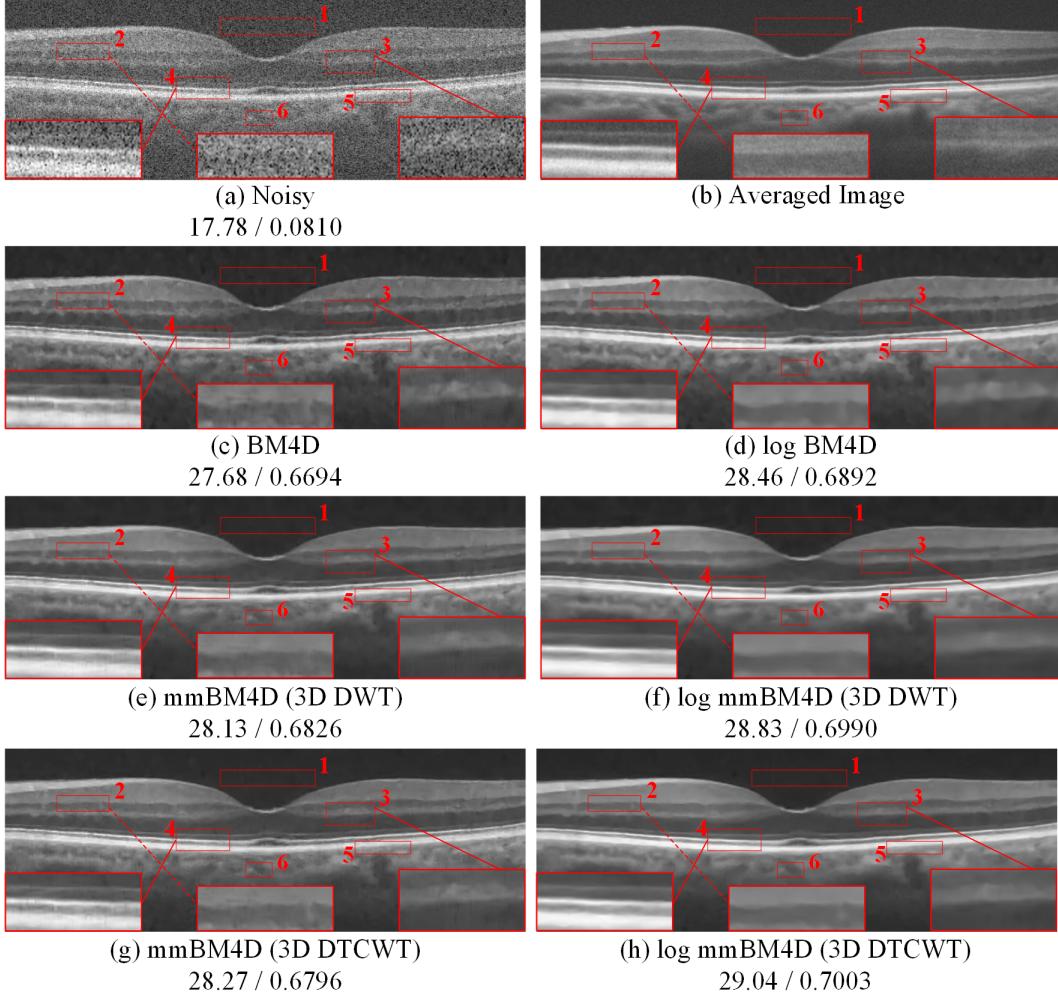


Figure 6: A visual comparison of the denoising results by BM4D and its two multiscale extensions implemented with 3D DWT and 3D DTCWT. The transform type is mentioned in the parenthesis. The prefix "log" shows that the corresponding method is applied in the logarithm domain. For each method, the PSNR and SSIM results are reported below.

6.1 2D Natural Image Denoising

6.1.1 Datasets

To evaluate the performance of our proposed multiscale methods with their single-scale counterparts in removing synthetic Gaussian noise from gray-scale images, two datasets are used here: 1. Classic test images (9 images) [52]: includes *Barbara*, *Boat*, *Cameraman*, *Fingerprint*, *Hill*, *Lena*, *Couple*, *Pentagon*, and *Man*. All images have 512×512 pixels, except for *Pentagon* which has 1024×1024 pixels, and 2. CSR test set (8 images) [46]: a set of relatively high-resolution test images (e.g., 1423×949 pixels). Note that both of these datasets were previously employed for the assessment of multiscale denoising methods [46, 52]. We corrupt all of the images from these datasets by adding white Gaussian noise with three different standard deviations (15, 25, and 50) from relatively weak to strong noise levels.

6.1.2 Results

We show visual comparisons in Fig. 7 between the results of three single-scale denoisers and their corresponding multiscale extensions, which are developed based on mDn (Algorithm 1) and our proposed mmDn (Algorithm 3) methods. This figure shows that, as expected, the single-scale denoising methods cannot recover the building facades. To recover the zoomed region, it seems that non-local information might be beneficial because it is a dominant pattern in this figure. However, even SAIST which is able to use this information, fails to recover most of it. Multiscale processing is helpful in this case. It can be seen that our proposed multiscale extensions (whose names begin with "mm") significantly improve the results in comparison with both their corresponding single-scale and mDn based (whose names are started with "m") methods. Note that in [51], mK-SVD was referred to as MS K-SVD.

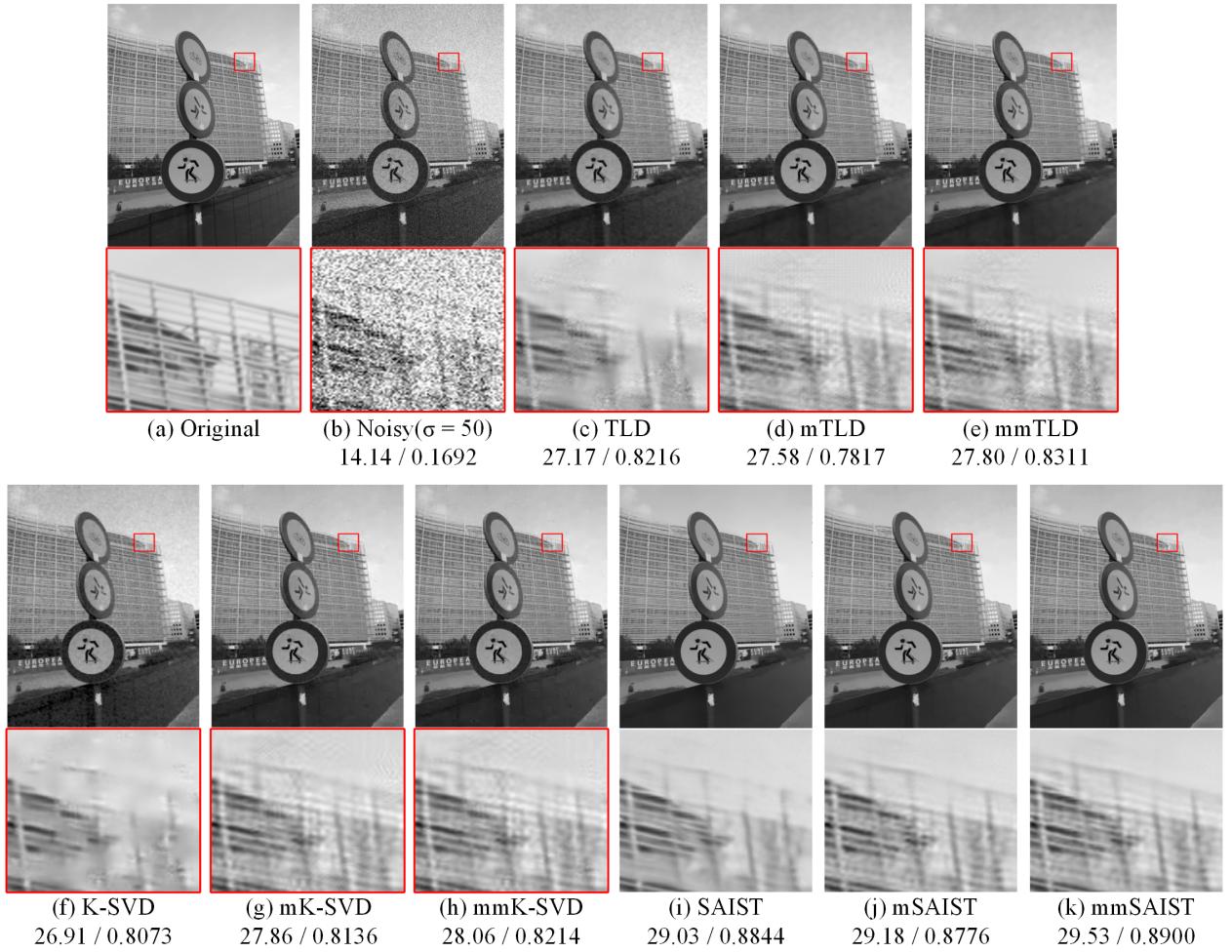


Figure 7: Visual comparison of the results obtained by three groups of single-scale and multiscale denoising methods. In each group, there is one single-scale (TLD, K-SVD, and SAIST) and two multiscale denoising methods whose names begin with "m" or "mm". For each method, the PSNR and SSIM results are reported below.

The observations and explanations can be validated by the average quantitative results reported in Table 2 and 3. The results obtained on the classic test images are reported in Table 2. It can be seen that in almost all cases our proposed subbands mixing based multiscale methods improve significantly over their corresponding baselines. For each

multiscale method, the gain over its single-scale counterpart for all three noise levels are averaged and then reported in the last column of the table. Similar to the other studies such as [46, 51, 52, 54, 60], it can be seen that the amount of gains obtained by our proposed mmDn based methods depend on the noise level, baseline methods, and datasets. Generally, no gain from multiscale processing is expected when noise standard deviation (σ) is below 20 [48]. Table 2 and 3 show that when noise is relatively weak ($\sigma = 15$), both mDn and mmDn based methods fail to have significant gains except for mmSAIST. But, when noise level increases, the other methods also show considerable gains.

These tables show that for both datasets, mSAIST leads to unsatisfactory results (on average, it shows either decreased performance or no improvement). This is because SAIST was not originally designed to directly denoise high frequency content. Instead, our mmDn based version of it (mmSAIST), bypasses this difficulty by extracting the denoised detail subbands from the denoised approximate subbands. Therefore, mmSAIST consistently improves the results on both datasets. The best result with mDn based methods achieved by mK-SVD (or MS K-SVD [51]). This method was highly engineered to denoise detail subbands. In contrast, our corresponding mmDn based method (mmK-SVD) was just uses the default parameters of the K-SVD denoising [22] to denoise the approximate subbands. Since the nature of the approximate subbands are more similar to the input image itself, almost no tuning is needed for developing mmK-SVD, while it has better performance.

Table 2: Mean of PSNR (dB) of each method for Gaussian image denoising of classic test images. For each group of compared methods, the best PSNR result is shown in **bold**. Also, averaged PSNR gains for each multiscale method is reported in the last column.

Methods	15	25	50	Gain
Noisy Images	24.61	20.17	14.15	-
TLD	32.03	29.45	26.10	-
mTLD	31.87	29.46	26.30	0.02
mmTLD	32.05	29.59	26.43	0.16
K-SVD	32.13	29.58	26.00	-
mK-SVD	31.95	29.65	26.68	0.19
mmK-SVD	32.11	29.81	26.69	0.30
SAIST	32.16	30.03	27.03	-
mSAIST	31.68	29.60	26.98	-0.32
mmSAIST	32.30	30.16	27.20	0.15

Table 3: Mean of PSNR (dB) of each method for Gaussian image denoising of gray-scale high-resolution images of CSR test set. For each group of compared methods, the best PSNR result is shown in **bold**. Also, averaged PSNR gains for each multiscale method is reported in the last column.

Methods	15	25	50	Gain
Noisy Images	24.61	20.17	14.16	-
TLD	33.36	30.49	26.90	-
mTLD	33.31	30.61	27.15	0.11
mmTLD	33.47	30.74	27.33	0.26
K-SVD	33.59	30.57	26.83	-
mK-SVD	33.27	30.63	27.41	0.11
mmK-SVD	33.48	30.81	27.47	0.26
SAIST	33.40	31.28	27.89	-
mSAIST	33.38	31.10	28.09	0.00
mmSAIST	33.80	31.48	28.20	0.30

6.2 3D OCT Image Denoising

Experiments with natural images (Section 6.1.2) show that when noise is not weak, the proposed mmDn-based methods outperform single-scale methods significantly. Noise in OCT images is usually not weak. Therefore, we expect that harnessing the idea of multiscale denoising for OCT images enhances the result. In this section, we demonstrate our OCT denoising results. As we have introduced in Section 5.1, here, the baseline single-scale denoiser is BM4D. Although BM4D is not considered a dedicated OCT denoising method, we show that its extension based on mmDn can boost its performance in such a way that it can compete well with various dedicated OCT denoising methods. We also enriched our experiments with OCT layer segmentation results to show that multiscale BM4D is able to preserve layer structures, which are among the most important cues for clinicians.

6.2.1 Datasets

In order to evaluate the 3D OCT denoising methods, we conduct denoising on three publicly available datasets. The first dataset contains 18 spectral domain OCT (SDOCT) images captured by a commercial Bioptigen SDOCT imaging device [25, 26, 28]. Due to providing high signal-to-noise-ratio (SNR) images, this dataset is very popular for evaluating OCT image reconstruction methods [10, 12]. An example of a high SNR image is illustrated in Fig. 6 (b), which was the result of an averaging and registration procedure [25]. Each subject was either diagnosed as normal or age-related macular degeneration (AMD), and there are four noisy nearby OCT images for each test image, and thus, 3D OCT processing is possible. Specifically, each input volume has a size of $450 \times 900 \times 5$ voxels.

The second dataset contains 13 SDOCT volumes captured from the macular area in the retina by a Topcon 3D OCT-1000 imaging device. The subjects were without any anomalies, and the size of each volume is $650 \times 512 \times 128$ voxels [24]. The third dataset contains SDOCT volumes of size $512 \times 1000 \times 100$ voxels captured by a Biotigen imaging device from 269 AMD and 115 normal subjects. Due to the computational burdens of denoising these high-resolution volumes, we randomly chose one AMD and one normal SDOCT volumes for our upcoming experiments from this dataset. In contrast to the first dataset, there are no high SNR images in the second and third datasets. Therefore, the denoising results are only assessed either visually or through non-reference image quality assessment metrics.

6.2.2 Metrics

We exploit different image reconstruction metrics to assess the quality of denoising. The PSNR, SSIM, mean-to-standard-deviation ratio (MSR) [73], contrast-to-noise-ratio (CNR) [74], and equivalent number of looks (ENL) [1] are used here. In contrast to PSNR and SSIM, the other metrics do not need reference images and are computed locally. Therefore, it is required to select a few regions of interest (ROIs) in the images. The contrast between foreground regions (e.g., red box 2-6 in Fig. 6) and background noise is measured through the CNR metric. The background noise is considered using a region from the background area (e.g., red box 1 in Fig. 6). MSR is a sign of good feature recovery without considering the background regions. CNR considers both background and foreground regions, and it is big

when ROIs contain prominent features. ENL is a measure of smoothness in the background. Large ENL values indicate stronger noise suppression and smoother background areas.

6.2.3 Denoising Results on the First OCT Image Dataset

In Fig. 8, a visual comparison of the denoising results obtained using 11 denoising methods is presented. All parameters involved in the compared methods were optimally assigned or chosen as described in their reference papers. We use a 5-scale 3D DWT and 3-scale 3D DTCWT to implement our proposed mixed multiscale BM4D extensions (mmBM4D). Note that the number of decomposition levels are fixed during all experiments on this dataset. Increasing the number of decomposition levels hardly improves the results while negatively contributing to the runtime. The compared methods in Fig. 8 are as follows: Sparse K-SVD [75], tensor dictionary learning (Tensor DL) [76], wavelet based denoising of multiple frame data (WMF) [16], multi-scale sparsity based tomographic denoising (MSBTD) [28], nonlocal weighted sparse representation (NWSR) [27], 3D sparsity based simultaneous denoising and interpolation (3D SBSDI) [25], 3D segmentation based sparse reconstruction (3D SSR) [26], and multi-input fully-convolutional network (MIFCN) [29]. Among these methods, only MSBTD, NWSR are single-frame, and the others are 3D denoising methods.

As can be seen in Fig. 8, all 3D denoising methods (except log WMF) outperform 2D denoising counterparts (MSBTD and NWSR). Sparse K-SVD and Tensor DL suppress noise while introducing blurring and artifacts. 3D SBSDI further reduces the artifacts, but it results in a slightly noisy reconstruction. 3D SSR reduces the noise greatly, however, its results are not natural-looking, due to exploiting segmentation during denoising. MIFCN preserves the layers, however, its result's boundaries are somewhat blurry. Overall, the proposed log mmBM4D methods can effectively reduce noise while preserving many structures, compared to the average image. Comparing different realizations of log mmBM4D through 3D DWT and 3D DTCWT, it can be seen that 3D DTCWT leads to a sharper denoising quality. This is consistent with the results reported in [19], where they found that 2D denoising through CWT better preserves edges in OCT images. Furthermore, the superiority of proposed methods is reflected in both PSNR and SSIM scores.

The observations and explanations can be validated by the average quantitative results reported in Table 4. This table provides a comprehensive evaluation of the compared methods on the first dataset (Section 6.2.1). In addition to the compared methods in Fig. 8, we also report the average PSNR and SSIM results of two other recent OCT denoising methods: Laplacian scale mixture-generalized extreme value (LSM-GEV) [12] and DnCNN [31]. It can be seen that, in terms of average PSNR, MSR, and CNR, the best result is achieved by 3D SBSDI. However, note that 3D SBSDI is a supervised denoising method which is specifically designed for OCT images. In contrast, our log mmBM4D methods are unsupervised, i.e., log mmBM4D does not need high SNR images for training its parameters. Considering the hard and costly procedure of obtaining these images, mmBM4D has wider applicability in comparison with supervised methods. Nevertheless, the proposed log mmBM4D's realizations outperform the original BM4D by more than 0.68 dB, and they achieve the best SSIM results, which indicates better edge recovery. Edges in OCT images are of the utmost

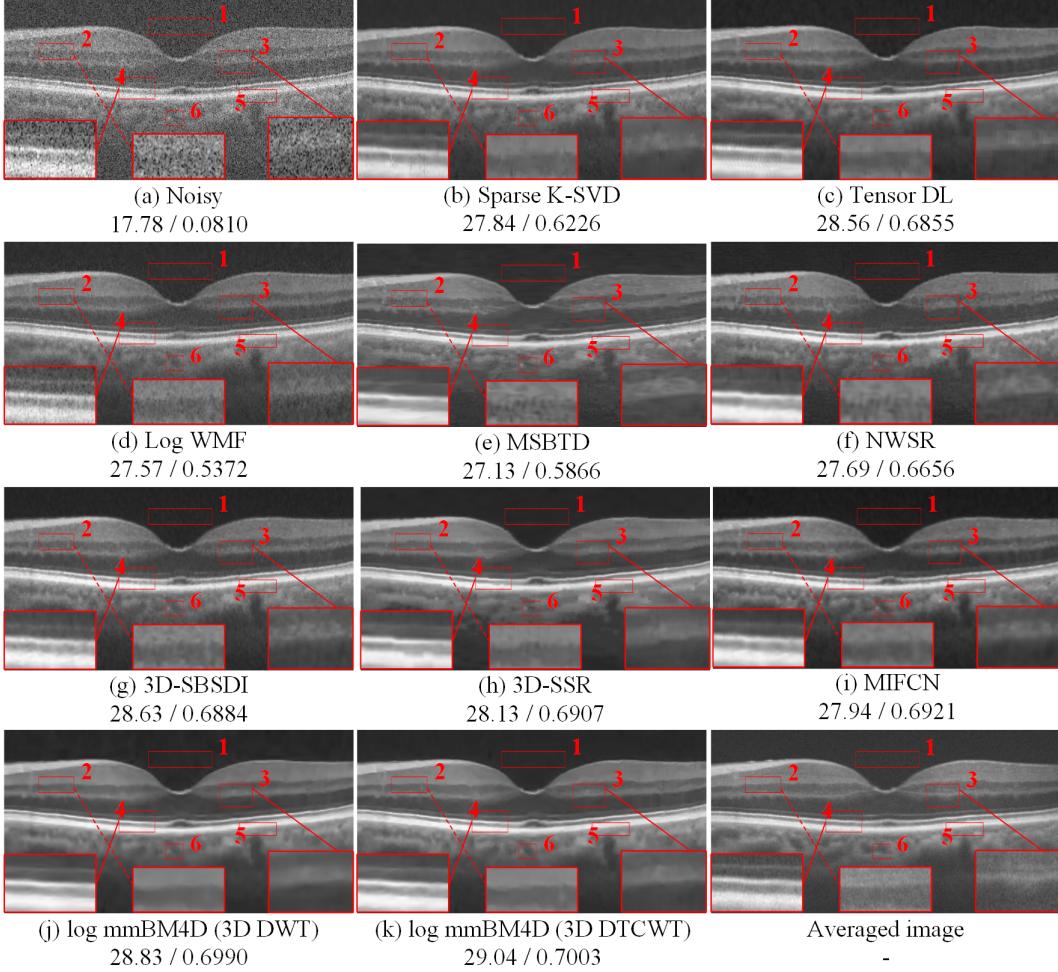


Figure 8: Visual comparison of the 3D OCT denoising results by the compared methods and our proposed multiscale BM4D extensions. For each method, the PSNR and SSIM results are reported below.

importance since they play a prominent role in layer segmentation, which is a common analysis over OCT images. We will examine the edge preservation quality of the compared methods in Section 6.2.4.

Table 4 also shows that, in terms of runtime, the fastest method is MIFCN. This method is essentially a neural network which exploits the power of GPU computations. The third fastest method is BM4D. Our proposed log mmBM4D (3D DTCWT) has a reasonable overhead. It adds only 2.5 seconds per frame to incorporate the ability of 3D processing into the original BM4D. However, when 3D DWT is used instead of 3D DTCWT, the running time increases. This difference between the two realizations of mmBM4D is due to the difference in the number of decomposition levels, filter banks, and wavelet transforms. For log mmBM4D (3D DWT), we have tested different filter banks including Daubechies wavelets, Symlets, etc. [61]. We empirically found that Daubechies 7 (db7) filter bank is a good choice for this method.

The filter length for db7 is $2 \times 7 = 14$, and the input depth is 5 (Section 6.2.1). Because we have used log mmBM4D (3D DWT) with 5 scales, according to the MATLAB®'s documentation for "dwt3" function, the approximate subbands

Table 4: Mean of PSNR (dB), SSIM, MSR, CNR, ENL of each method for denoising 18 OCT images. The best quality metric results obtained by the compared methods are shown in **bold** and the second one is underlined. Additionally, the average processing time per frame (TPF) for denoising OCT volumes with size $450 \times 900 \times 5$ voxels is reported. Furthermore, the last column indicates whether each method is unsupervised or not. The star "*" shows that the output/source code of a method is not publicly available, and the values are copied from the original paper.

Methods	PSNR	SSIM	MSR	CNR	ENL	TPF	Unsupervised?
Noisy Images	17.74	0.0867	3.18	1.61	6.97	-	-
Sparse K-SVD	27.35	0.6145	7.99	3.46	3027.38	7.9	✓
Tensor DL	28.03	0.6734	8.06	3.63	1116.08	4.75	✓
log WMF	26.67	0.5347	7.29	3.52	232.88	0.80	✓
MSBTD	27.08	0.5902	7.43	3.34	1643.06	780.0	✗
NWSR	27.79	0.6686	<u>8.67</u>	3.45	1514.89	66.5	✗
3D SBSDI	28.27	0.6853	8.72	3.75	2572.91	9.5	✗
3D SSR	28.10	0.6905	8.04	3.57	<u>5225.34</u>	-	✗
MIFCN	27.87	0.6879	8.34	<u>3.73</u>	2340.38	0.2	✗
DnCNN*	28.08	0.6910	n\aa	n\aa	n\aa	n\aa	✗
LSM-GEV*	28.25	0.6879	n\aa	n\aa	n\aa	150	✓
BM4D (baseline)	27.52	0.6597	7.14	3.32	1507.68	4.02	✓
log BM4D	27.89	0.6822	7.47	3.43	1664.02	4.02	✓
log mmBM4D (3D DWT)	28.21	0.6925	8.07	3.58	3720.29	11.2	✓
log mmBM4D (3D DTCWT)	<u>28.25</u>	<u>0.6924</u>	8.12	3.65	7530.18	6.5	✓

have the following depths: 9, 11, 12, 12, 12. This is because when the filter length is a and the input dimension is b , performing 1-scale decomposition results in a depth equal to $\lfloor (a+b)/2 \rfloor$. However, the computation for 3D DTCWT is different. 3D DTCWT requires its input dimensions to be a power of two, then it halves the input dimensions on each scale. To perform 3 scales of 3D DTCWT on the input with depth equal to 5, we need to pad it to become a power of two (here, it becomes 8). Then, the approximate subbands have the following depths: 4, 2, 2. Note that for computing 3D DTCWT, the input dimension must be even and greater than or equal to 4 [77, 78]. Therefore, on the second scale, we need to again pad the volume to become 4.

6.2.4 Segmenting Layers of Noisy and Denoised OCT Images

Even though improving the visual quality of OCT images is desirable, in practical applications it is more important to increase the quality of high-level analysis such as layer segmentation and image classification. Here, this question is triggered: Do the proposed methods for restoring OCT images improve the quality of retinal layer segmentation? To answer this question, we have exploited two open source programs: CASEREL [4, 79] and LiveLayer [80]. First, we used CASEREL to automatically segment retinal layers on noisy and denoised OCT images. Then, the results were compared with the segmentation maps created for all test images (Section 6.2.1) by an expert through the manual layer segmentation tab of LiveLayer [80]. The segmented borders are internal limiting membrane (ILM), nerve fiber layer-ganglion cell layer (NFL-GCL), inner plexiform layer-inner nuclear layer (IPL-INL), outer plexiform layer-outer nuclear layer (OPL-ONL), inner segments of photoreceptors-outer segments of photoreceptors (IS-OS), Bruch's membrane-Choroid (BM-Choroid) from top to bottom, respectively. For each image, we compute the unsigned border

positioning error for each segmented border by calculating the absolute difference of vertical values (for each pixel) of the segmented and ground-truth borders [10]. Finally, the average errors are summed up and indicated by the sum of absolute error (SAE) for each method.

We have summarized the retinal layer segmentation errors of the compared methods evaluated on the first dataset (Section 6.2.1) in Table 5. The average error made by each method for segmenting each layer is reported separately, and their sum is reported in the last column (SAE). These results reveal that identifying borders in the noisy images leads to the highest error, and all denoising methods can help reduce the segmentation errors to some extent. In terms of SAE, the best method for preserving layer structures is 3D SSR. This is expected because 3D SSR [26] is not only a supervised denoising method, but it also internally exploits an advanced segmentation method through its reconstruction process. Therefore, its denoising results are expected to preserve layer structures better than the other compared methods. Nevertheless, our proposed log mmBM4D methods can achieve very similar SAEs in comparison with 3D SSR. Also, the results show that incorporating multiscale information into BM4D [43] through the proposed approach can greatly enhance its results in such a way that our multiscale extensions for BM4D are well comparable (both in terms of denoising scores and segmentation errors) with sophisticated methods such as 3D SSR and other dedicated OCT denoising methods.

Table 5: Average border position errors obtained by segmenting retinal layers from OCT images denoised by the compared methods. The sum of errors for each method is reported in the last column (SAE).

Methods	ILM	NFL-GCL	IPL-INL	OPL-ONL	ONL-IS/OS	BM-Choroid	SAE
Noisy	2.090	7.018	5.357	5.233	2.136	2.165	24.000
log WMF	1.706	2.415	3.256	4.466	1.649	1.759	15.251
MSBTD	1.332	2.159	3.118	4.280	1.219	1.673	13.781
NWSR	1.355	2.178	3.250	4.241	1.225	1.800	14.049
3D SBSDI	1.316	3.091	3.064	3.863	1.306	1.603	14.244
3D SSR	1.211	1.914	2.817	3.661	1.034	1.754	12.391
MIFCN	1.300	2.493	2.994	3.809	1.138	1.803	13.536
log BM4D	1.396	2.400	2.826	3.961	1.226	2.060	13.869
log mmBM4D (3D DWT)	1.381	2.023	2.635	3.536	1.167	1.753	12.496
log mmBM4D (3D DTCWT)	1.432	2.025	2.594	3.567	1.147	1.658	<u>12.422</u>

Here, because we have reported both denoising scores and segmentation errors for all compared methods, it is possible to use the reported results in Table 4 and 5 to draw the plot of PSNR versus SAE in Fig. 1. This figure shows that the proposed multiscale extensions for BM4D can achieve a good denoising quality score while it also preserves layer structures. Furthermore, it reveals that while generally there is a correlation between quality scores and segmentation errors, there might be some contradictions in practice. For example, 3D SBSDI achieves the best result in terms of PSNR. However, probably due to remaining noise and artifacts in its results (e.g., Fig. 8 (g)), its corresponding segmentation error is higher than what is expected.

6.2.5 Denoising Results on the Second and Third OCT Image Datasets

In this section, we further evaluate the applicability and performance of our proposed mixed multiscale methods on two other datasets (Section 6.2.1).

As we have mentioned earlier, the second dataset contains 13 volumes, each one has 128 slices. To quantitatively and qualitatively assess the denoising quality of the compared 3D denoising methods, we randomly selected three slices from each volume. Then, using the compared method, each slice is denoised by utilizing seven slices around it (i.e., the input volume size is $650 \times 512 \times 8$ voxels). The denoising results of the original BM4D and our proposed multiscale extensions are shown in Fig. 9. Furthermore, the average MSR, CNR, and ENL metric values (computed over $13 \times 3 = 39$ images) are presented in Table 6. It is evident that our proposed mixed multiscale BM4D extensions greatly improve over the original BM4D method, for which the output is dominated by artifacts.

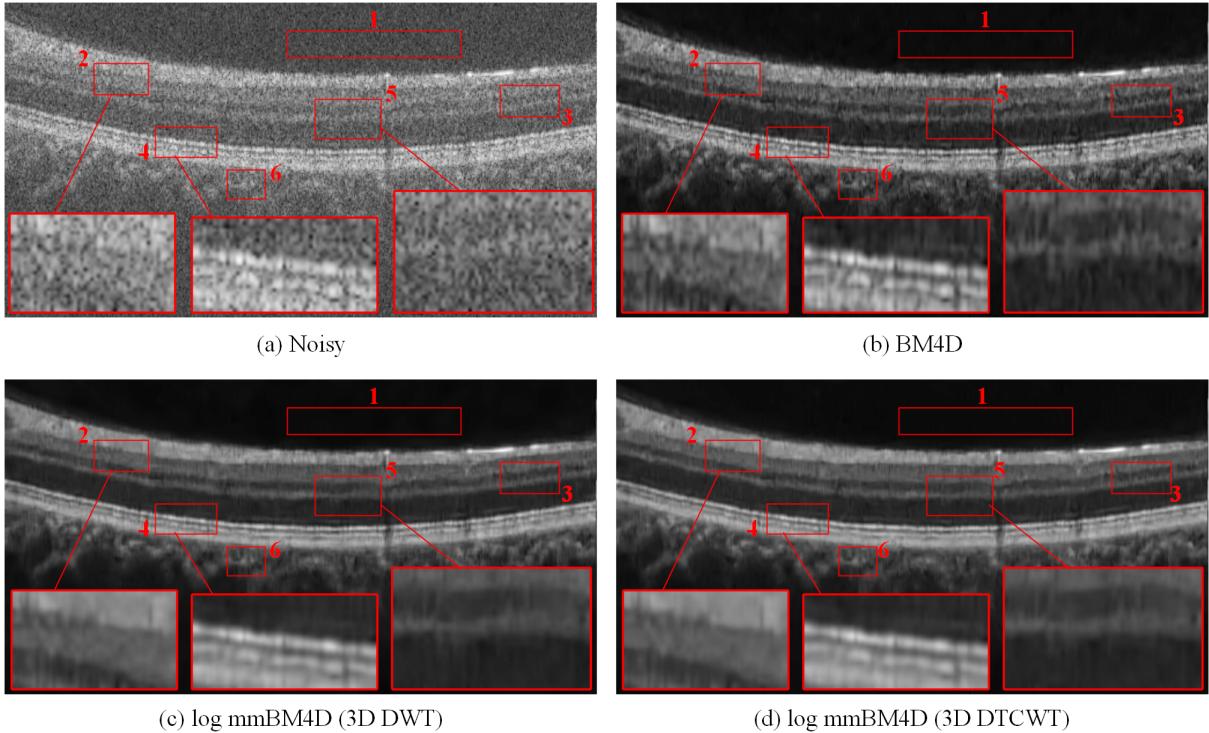


Figure 9: Visual comparison of the 3D OCT denoising results obtained by denoising a test image from the second dataset through BM4D and the proposed multiscale BM4D extensions.

In the third dataset, the volumes are relatively high-resolution ($512 \times 1000 \times 100$ voxels). In contrast to the experiments on the first and second datasets, here, we fed the whole volume to the denoising method. Due to the computational costs or intrinsic method limitations (e.g., MIFCN only accepts 5 slices per volume, supervised methods are not trained on this dataset), the application of most compared denoising methods reported in Section 6.2.3 to denoise a whole volume is not straightforward. However, due to the flexibility and computational advantages of BM4D and subsequently the proposed multiscale extensions, we can feed a whole volume to BM4D or log mmBM4D at a reasonable runtime. A visual comparison of the denoising results for one test slice from this dataset is shown in Fig. 10. We also report the

Table 6: Mean of MSR, CNR, ENL of each method for 3D denoising 39 TOPCON OCT images. The best quality metric results obtained by the compared methods are shown in **bold** and the second one is underlined. Additionally, the average processing time per frame (TPF) for denoising OCT volumes with size $650 \times 512 \times 8$ voxels is reported in the last column.

Methods	MSR	CNR	ENL	TPF
Noisy Images	4.24	1.86	29.82	-
BM4D (baseline)	6.83	2.90	3335.03	4.8
log BM4D	7.87	3.17	4282.53	4.8
log mmBM4D (3D DWT)	<u>8.01</u>	<u>3.20</u>	<u>6067.87</u>	5.6
log mmBM4D (3D DTCWT)	8.15	3.25	8994.30	5.0

average metric values for denoising 2 normal and AMD OCT volumes at Table 7. These metric values were computed based on five random slices per volume.

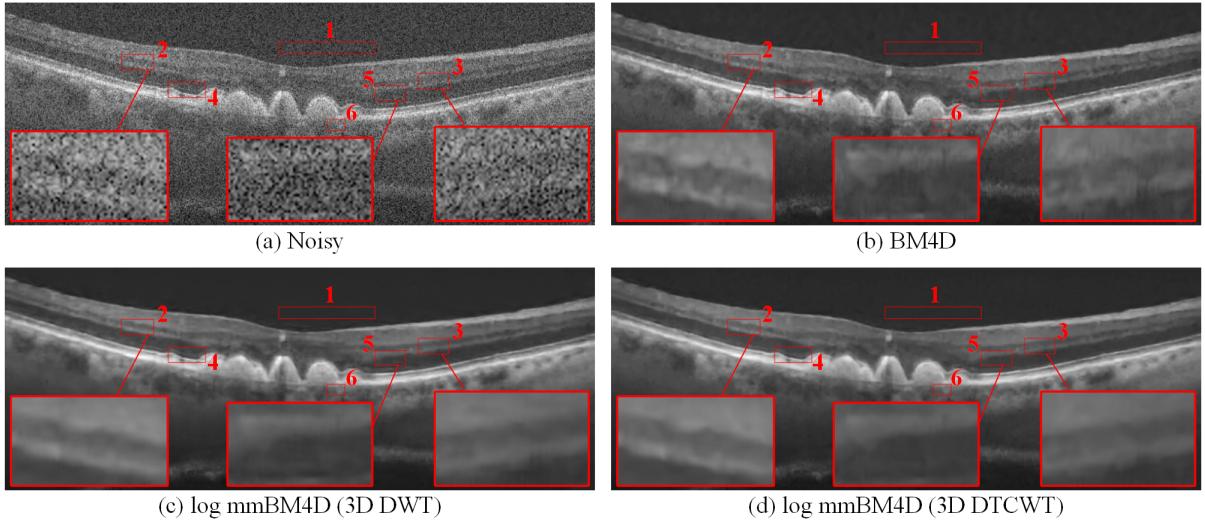


Figure 10: Visual comparison of the 3D OCT denoising results obtained by denoising a test image from the third dataset through BM4D and the proposed multiscale BM4D extensions.

Table 7: Mean of MSR, CNR, ENL of each method for denoising the third OCT image dataset. The best quality metric results obtained by the compared methods are shown in **bold** and the second one is underlined. Additionally, the average processing time per frame (TPF) for denoising OCT volumes with size $512 \times 1000 \times 100$ voxels is reported in the last column.

Methods	MSR	CNR	ENL	TPF
Noisy Images	3.05	1.31	6.73	-
BM4D (baseline)	7.77	3.13	2335.46	17.95
log BM4D	8.40	3.30	2375.00	17.95
log mmBM4D (3D DWT)	<u>8.25</u>	<u>3.30</u>	9643.98	20.51
log mmBM4D (3D DTCWT)	8.68	3.41	<u>7395.83</u>	20.86

7 Conclusion

This paper presents a method for constructing multiscale denoising methods from single-scale ones. We start with the subbands denoising approach which was initially proposed by multiscale K-SVD [51]. Then, by emphasizing its shortcomings, we propose the mixed multiscale method. Our proposed method avoids directly denoising detail subbands, and instead, extracts them from the approximate subbands at finer level. Therefore, it is possible to combine the advantages of single-scale and multiscale methods in an efficient way. We have exploited the proposed method to construct multiscale extensions for multiple 2D and 3D denoising methods. The experiments show that our multiscale extensions can significantly boost their single-scale counterparts. In particular, we propose a multiscale extension for the well-known BM4D benchmark denoising method, which can outperform several dedicated recent OCT image denoising methods. We do not restrict our evaluations to the usual quantitative and qualitative image quality assessment experiments, and further, we evaluate the retinal layer preservation quality of the proposed OCT denoising method by conducting comprehensive segmentation experiments.

Acknowledgment

This work was supported in part by grant number R01EY030929 from the National Institute of Health.

References

- [1] A. Pizurica, L. Jovanov, B. Huysmans, V. Zlokolica, P. De Keyser, F. Dhaenens, and W. Philips, “Multiresolution denoising for optical coherence tomography: a review and evaluation,” *Current Medical Imaging*, vol. 4, no. 4, pp. 270–284, 2008.
- [2] M. D. Abràmoff, M. K. Garvin, and M. Sonka, “Retinal imaging and image analysis,” *IEEE Reviews in Biomedical Engineering*, vol. 3, pp. 169–208, 2010.
- [3] A. Lang, A. Carass, M. Hauser, E. S. Sotirchos, P. A. Calabresi, H. S. Ying, and J. L. Prince, “Retinal layer segmentation of macular OCT images using boundary classification,” *Biomedical Optics Express*, vol. 4, no. 7, pp. 1133–1152, 2013.
- [4] S. J. Chiu, X. T. Li, P. Nicholas, C. A. Toth, J. A. Izatt, and S. Farsiu, “Automatic segmentation of seven retinal layers in SD-OCT images congruent with expert manual segmentation,” *Optics Express*, vol. 18, no. 18, pp. 19413–19428, 2010.
- [5] L. de Sisternes, J. Hong, T. Leng, and D. L. Rubin, “A machine learning approach for device-independent automated segmentation of retinal cysts in spectral domain optical coherence tomography images,” *Proceeding Optima Challenge-MICCAI*, 2015.
- [6] M. Esmaeili, A. M. Dehnavi, H. Rabbani, and F. Hajizadeh, “3d segmentation of retinal cysts from SD-OCT images by the use of three dimensional curvelet based K-SVD,” *J Med Signals Sens*, vol. 6, pp. 166–71, 2015.

- [7] A. Ozcan, A. Bilenca, A. E. Desjardins, B. E. Bouma, and G. J. Tearney, “Speckle reduction in optical coherence tomography images using digital filtering,” *Journal of the Optical Society of America A*, vol. 24, no. 7, pp. 1901–1910, 2007.
- [8] P. Rodrigues and R. Bernardes, “3-d adaptive nonlinear complex-diffusion despeckling filter,” *IEEE Transactions on Medical Imaging*, vol. 31, no. 12, pp. 2205–2212, 2012.
- [9] J. Duan, W. Lu, C. Tench, I. Gottlob, F. Proudlock, N. N. Samani, and L. Bai, “Denoising optical coherence tomography using second order total generalized variation decomposition,” *Biomedical Signal Processing and Control*, vol. 24, pp. 120–127, 2016.
- [10] P. G. Daneshmand, A. Mehridehnavi, and H. Rabbani, “Reconstruction of optical coherence tomography images using mixed low rank approximation and second order tensor based total variation method,” *IEEE Transactions on Medical Imaging*, 2020.
- [11] S. Jorjandi, H. Rabbani, Z. Amini, and R. Kafieh, “OCT image denoising based on asymmetric normal laplace mixture model,” in *International Conference of the IEEE Engineering in Medicine and Biology Society*, pp. 2679–2682, IEEE, 2019.
- [12] P. G. Daneshmand, H. Rabbani, and A. Mehridehnavi, “Super-resolution of optical coherence tomography images by scale mixture models,” *IEEE Transactions on Image Processing*, vol. 29, pp. 5662–5676, 2020.
- [13] M. Samieinasab, Z. Amini, and H. Rabbani, “Multivariate statistical modeling of retinal optical coherence tomography,” *IEEE Transactions on Medical Imaging*, vol. 39, no. 11, pp. 3475–3487, 2020.
- [14] C. Cuartas-Vélez, R. Restrepo, B. E. Bouma, and N. Uribe-Patarroyo, “Volumetric non-local-means based speckle reduction for optical coherence tomography,” *Biomedical Optics Express*, vol. 9, no. 7, pp. 3354–3372, 2018.
- [15] S. Chitchian, M. A. Mayer, A. Boretsky, F. J. Van Kuijk, and M. Motamedi, “Retinal optical coherence tomography image enhancement via shrinkage denoising using double-density dual-tree complex wavelet transform,” *Journal of Biomedical Optics*, vol. 17, no. 11, p. 116009, 2012.
- [16] M. A. Mayer, A. Borsdorf, M. Wagner, J. Hornegger, C. Y. Mardin, and R. P. Tornow, “Wavelet denoising of multiframe optical coherence tomography data,” *Biomedical Optics Express*, vol. 3, no. 3, pp. 572–589, 2012.
- [17] H. Rabbani, M. Sonka, and M. D. Abramoff, “Optical coherence tomography noise reduction using anisotropic local bivariate gaussian mixture prior in 3d complex wavelet domain,” *International Journal of Biomedical Imaging*, vol. 2013, 2013.
- [18] F. Zaki, Y. Wang, H. Su, X. Yuan, and X. Liu, “Noise adaptive wavelet thresholding for speckle noise removal in optical coherence tomography,” *Biomedical Optics Express*, vol. 8, no. 5, pp. 2720–2731, 2017.
- [19] Z. Khodabandeh, H. Rabbani, and A. Mehri, “Geometrical x-lets for image denoising,” in *Proceedings of The 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pp. 2691–2694, IEEE, 2019.

- [20] M. Esmaeili, A. M. Dehnavi, F. Hajizadeh, and H. Rabbani, “Three-dimensional curvelet-based dictionary learning for speckle noise removal of optical coherence tomography,” *Biomedical Optics Express*, vol. 11, no. 2, pp. 586–608, 2020.
- [21] J. Xu, H. Ou, E. Y. Lam, P. Chui, and K. K. Wong, “Speckle reduction of retinal optical coherence tomography based on contourlet shrinkage,” *Optics Letters*, vol. 38, no. 15, pp. 2900–2903, 2013.
- [22] M. Elad and M. Aharon, “Image denoising via sparse and redundant representations over learned dictionaries,” *IEEE Transactions on Image Processing*, vol. 15, no. 12, pp. 3736–3745, 2006.
- [23] M. Aharon, M. Elad, and A. Bruckstein, “K-svd: An algorithm for designing overcomplete dictionaries for sparse representation,” *IEEE Transactions on signal processing*, vol. 54, no. 11, pp. 4311–4322, 2006.
- [24] R. Kafieh, H. Rabbani, and I. Selesnick, “Three dimensional data-driven multi scale atomic representation of optical coherence tomography,” *IEEE Transactions on Medical Imaging*, vol. 34, no. 5, pp. 1042–1062, 2014.
- [25] L. Fang, S. Li, R. P. McNabb, Q. Nie, A. N. Kuo, C. A. Toth, J. A. Izatt, and S. Farsiu, “Fast acquisition and reconstruction of optical coherence tomography images via sparse representation,” *IEEE Transactions on Medical Imaging*, vol. 32, no. 11, pp. 2034–2049, 2013.
- [26] L. Fang, S. Li, D. Cunefare, and S. Farsiu, “Segmentation based sparse reconstruction of optical coherence tomography images,” *IEEE Transactions on Medical Imaging*, vol. 36, no. 2, pp. 407–421, 2016.
- [27] A. Abbasi, A. Monadjemi, L. Fang, and H. Rabbani, “Optical coherence tomography retinal image reconstruction via nonlocal weighted sparse representation,” *Journal of Biomedical Optics*, vol. 23, no. 3, p. 036011, 2018.
- [28] L. Fang, S. Li, Q. Nie, J. A. Izatt, C. A. Toth, and S. Farsiu, “Sparsity based denoising of spectral domain optical coherence tomography images,” *Biomedical Optics Express*, vol. 3, no. 5, pp. 927–942, 2012.
- [29] A. Abbasi, A. Monadjemi, L. Fang, H. Rabbani, and Y. Zhang, “Three-dimensional optical coherence tomography image denoising through multi-input fully-convolutional networks,” *Computers in Biology and Medicine*, vol. 108, pp. 1–8, 2019.
- [30] Y. Ma, X. Chen, W. Zhu, X. Cheng, D. Xiang, and F. Shi, “Speckle noise reduction in optical coherence tomography images based on edge-sensitive cGAN,” *Biomedical Optics Express*, vol. 9, no. 11, pp. 5129–5146, 2018.
- [31] N. Gour and P. Khanna, “Speckle denoising in optical coherence tomography images using residual deep convolutional neural network,” *Multimedia Tools and Applications*, pp. 1–17, 2019.
- [32] Y. Huang, Z. Lu, Z. Shao, M. Ran, J. Zhou, L. Fang, and Y. Zhang, “Simultaneous denoising and super-resolution of optical coherence tomography images based on generative adversarial network,” *Optics Express*, vol. 27, no. 9, pp. 12289–12307, 2019.
- [33] Y. Huang, W. Xia, Z. Lu, Y. Liu, H. Chen, J. Zhou, L. Fang, and Y. Zhang, “Noise-powered disentangled representation for unsupervised speckle reduction of optical coherence tomography images,” *IEEE Transactions on Medical Imaging*, 2020.

- [34] S. Apostolopoulos, J. Salas, J. L. Ordóñez, S. S. Tan, C. Ciller, A. Ebneter, M. Zinkernagel, R. Sznitman, S. Wolf, S. De Zanet, and M. R. Munk, “Automatically enhanced OCT scans of the retina: A proof of concept study,” *Scientific reports*, vol. 10, no. 1, pp. 1–8, 2020.
- [35] A. Guo, L. Fang, M. Qi, and S. Li, “Unsupervised denoising of optical coherence tomography images with nonlocal-generative adversarial network,” *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1–12, 2020.
- [36] V. Das, S. Dandapat, and P. K. Bora, “Unsupervised super-resolution of OCT images using generative adversarial network for improved age-related macular degeneration diagnosis,” *IEEE Sensors Journal*, vol. 20, no. 15, pp. 8746–8756, 2020.
- [37] Z. Chen, Z. Zeng, H. Shen, X. Zheng, P. Dai, and P. Ouyang, “DN-GAN: Denoising generative adversarial networks for speckle noise reduction in optical coherence tomography images,” *Biomedical Signal Processing and Control*, vol. 55, p. 101632, 2020.
- [38] B. Qiu, Z. Huang, X. Liu, X. Meng, Y. You, G. Liu, K. Yang, A. Maier, Q. Ren, and Y. Lu, “Noise reduction in optical coherence tomography images using a deep neural network with perceptually-sensitive loss function,” *Biomedical Optics Express*, vol. 11, no. 2, pp. 817–830, 2020.
- [39] N. Imanpour, A. R. Naghsh-Nilchi, A. Monadjemi, H. Karshenas, K. Nasrollahi, and T. B. Moeslund, “Memory- and time-efficient dense network for single-image super-resolution,” *IET Signal Processing*, vol. 15, no. 2, pp. 141–152, 2021.
- [40] B. Qiu, Y. You, Z. Huang, X. Meng, Z. Jiang, C. Zhou, G. Liu, K. Yang, Q. Ren, and Y. Lu, “N2NSR-OCT: Simultaneous denoising and super-resolution in optical coherence tomography images using semisupervised deep learning,” *Journal of Biophotonics*, vol. 14, no. 1, p. e202000282, 2021.
- [41] H. Cheong, S. K. Devalla, T. Chuangsawanich, T. A. Tun, X. Wang, T. Aung, L. Schmetterer, M. L. Buist, C. Boote, A. H. Thiéry, and M. J. A. Girard, “OCT-GAN: single step shadow and noise removal from optical coherence tomography images of the human optic nerve head,” *Biomedical Optics Express*, vol. 12, no. 3, pp. 1482–1498, 2021.
- [42] K. Zhang, W. Zuo, S. Gu, and L. Zhang, “Learning deep cnn denoiser prior for image restoration,” in *IEEE Conference on Computer Vision and Pattern Recognition*, July 2017.
- [43] M. Maggioni, V. Katkovnik, K. Egiazarian, and A. Foi, “Nonlocal transform-domain filter for volumetric data denoising and reconstruction,” *IEEE Transactions on Image Processing*, vol. 22, no. 1, pp. 119–133, 2012.
- [44] J. Cheng, D. Tao, Y. Quan, D. W. K. Wong, G. C. M. Cheung, M. Akiba, and J. Liu, “Speckle reduction in 3d optical coherence tomography of retina by a-scan reconstruction,” *IEEE Transactions on Medical Imaging*, vol. 35, no. 10, pp. 2270–2279, 2016.
- [45] H. C. Burger and S. Harmeling, “Improving denoising algorithms via a multi-scale meta-procedure,” in *Joint Pattern Recognition Symposium*, pp. 206–215, Springer, 2011.

- [46] G. Facciolo, N. Pierazzo, and J.-M. Morel, “Conservative scale recombination for multiscale denoising (the devil is in the high frequency detail),” *SIAM Journal on Imaging Sciences*, vol. 10, no. 3, pp. 1603–1626, 2017.
- [47] W. Feng, P. Qiao, X. Xi, and Y. Chen, “Image denoising via multi-scale nonlinear diffusion models,” *CoRR*, vol. abs/1609.0, Sep 2016.
- [48] M. Lebrun, M. Colom, A. Buades, and J. M. Morel, “Secrets of image denoising cuisine,” *Acta Numerica*, vol. 21, p. 475–576, May 2012.
- [49] M. Lebrun, M. Colom, and J. Morel, “Multiscale image blind denoising,” *IEEE Transactions on Image Processing*, vol. 24, p. 3149–3161, Oct 2015.
- [50] J. Mairal, G. Sapiro, and M. Elad, “Learning multiscale sparse representations for image and video restoration,” *Multiscale Modeling & Simulation*, vol. 7, p. 214–241, Jan 2008.
- [51] B. Ophir, M. Lustig, and M. Elad, “Multi-scale dictionary learning using wavelets,” *IEEE Journal of Selected Topics in Signal Processing*, vol. 5, p. 1014–1024, Sep 2011.
- [52] V. Palyan and M. Elad, “Multi-scale patch-based image restoration,” *IEEE Transactions on Image Processing*, vol. 25, no. 1, pp. 249–261, 2015.
- [53] N. Pierazzo, J.-M. Morel, and G. Facciolo, “Multi-scale dct denoising,” *Image Processing On Line*, vol. 7, pp. 288–308, 2017.
- [54] J. Sulam, B. Ophir, and M. Elad, “Image denoising through multi-scale learnt dictionaries,” in *International Conference on Image Processing*, p. 808–812, IEEE, Oct 2014.
- [55] P. Coupé, P. Hellier, S. Prima, C. Kervrann, and C. Barillot, “3d wavelet subbands mixing for image denoising,” *International Journal of Biomedical Imaging*, vol. 2008, 2008.
- [56] A. Lukin, “A multiresolution approach for improving quality of image denoising algorithms,” in *IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 2, pp. II–857–II–860, IEEE, 2006.
- [57] B. A. Olshausen, P. Sallee, and M. S. Lewicki, “Learning sparse image codes using a wavelet pyramid architecture,” *Advances in neural information processing systems*, pp. 887–893, 2001.
- [58] P. Sallee and B. Olshausen, “Learning sparse multiscale image representations,” *Advances in neural information processing systems*, vol. 15, pp. 1351–1358, 2002.
- [59] S. Bacchelli and S. Papi, “Image denoising using principal component analysis in the wavelet domain,” *Journal of Computational and Applied Mathematics*, vol. 189, no. 1-2, pp. 606–621, 2006.
- [60] R. Yan, L. Shao, and Y. Liu, “Nonlocal hierarchical dictionary learning using wavelets for image denoising,” *IEEE Transactions on Image Processing*, vol. 22, no. 12, pp. 4689–4698, 2013.
- [61] S. Mallat, *A wavelet tour of signal processing*. Elsevier, 1999.
- [62] P. Burt and E. Adelson, “The laplacian pyramid as a compact image code,” *IEEE Transactions on Communications*, vol. 31, p. 532–540, Apr 1983.

- [63] S. Ravishankar and Y. Bresler, “Learning doubly sparse transforms for images,” *IEEE Transactions on Image Processing*, vol. 22, no. 12, pp. 4598–4612, 2013.
- [64] S. Ravishankar and Y. Bresler, “Learning sparsifying transforms,” *IEEE Transactions on Signal Processing*, vol. 61, no. 5, pp. 1072–1086, 2012.
- [65] S. Ravishankar and Y. Bresler, “Sparsifying transform learning with efficient optimal updates and convergence guarantees,” *IEEE Transactions on Signal Processing*, vol. 63, no. 9, pp. 2389–2404, 2015.
- [66] A. Foi, “Noise estimation and removal in mr imaging: The variance-stabilization approach,” in *International symposium on biomedical imaging: from nano to macro*, pp. 1809–1814, IEEE, 2011.
- [67] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, “Image quality assessment: from error visibility to structural similarity,” *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, 2004.
- [68] J.-L. Starck, J. Fadili, and F. Murtagh, “The undecimated wavelet decomposition and its reconstruction,” *IEEE Transactions on Image Processing*, vol. 16, no. 2, pp. 297–309, 2007.
- [69] E. Agustsson and R. Timofte, “Ntire 2017 challenge on single image super-resolution: Dataset and study,” in *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 126–135, 2017.
- [70] A. Ignatov, R. Timofte, T. Van Vu, T. Minh Luu, T. X Pham, C. Van Nguyen, Y. Kim, J.-S. Choi, M. Kim, J. Huang, J. Ran, C. Xing, X. Zhou, P. Zhu, M. Geng, Y. Li, E. Agustsson, S. Gu, L. V. Gool, E. de Stoutz, N. Kobyshev, K. Nie, Y. Zhao, G. Li, T. Tong, Q. Gao, L. Hanwen, P. Navarrete Michelini, Z. Dan, H. Fengshuo, Z. Hui, X. Wang, L. Deng, R. Meng, J. Qin, Y. Shi, W. Wen, L. Lin, R. Feng, S. Wu, C. Dong, Y. Qiao, S. Vasu, N. Thekke Madam, R. A. N. Kandula, Praveen, J. Liu, and C. Jung, “PIRM challenge on perceptual image enhancement on smartphones: Report,” in *IEEE European Conference on Computer Vision Workshops*, pp. 1–18, 2018.
- [71] W. Dong, G. Shi, and X. Li, “Nonlocal image restoration with bilateral variance estimation: a low-rank approach,” *IEEE Transactions on Image Processing*, vol. 22, no. 2, pp. 700–711, 2012.
- [72] I. W. Selesnick, R. G. Baraniuk, and N. C. Kingsbury, “The dual-tree complex wavelet transform,” *IEEE signal processing magazine*, vol. 22, no. 6, pp. 123–151, 2005.
- [73] G. Cincotti, G. Loi, and M. Pappalardo, “Frequency decomposition and compounding of ultrasound medical images with wavelet packets,” *IEEE Transactions on Medical Imaging*, vol. 20, no. 8, pp. 764–771, 2001.
- [74] P. Bao and L. Zhang, “Noise reduction for magnetic resonance images via adaptive multiscale products thresholding,” *IEEE Transactions on Medical Imaging*, vol. 22, no. 9, pp. 1089–1099, 2003.
- [75] R. Rubinstein, M. Zibulevsky, and M. Elad, “Double sparsity: Learning sparse dictionaries for sparse signal approximation,” *IEEE Transactions on signal processing*, vol. 58, no. 3, pp. 1553–1564, 2009.

- [76] Y. Peng, D. Meng, Z. Xu, C. Gao, Y. Yang, and B. Zhang, “Decomposable nonlocal tensor dictionary learning for multispectral image denoising,” in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2949–2956, 2014.
- [77] N. Kingsbury, “Complex wavelets for shift invariant analysis and filtering of signals,” *Applied and computational harmonic analysis*, vol. 10, no. 3, pp. 234–253, 2001.
- [78] H. Chen and N. Kingsbury, “Efficient registration of nonrigid 3-d bodies,” *IEEE Transactions on Image Processing*, vol. 21, no. 1, pp. 262–272, 2011.
- [79] P.-y. Teng, “Caserel–An open source software for computer-aided segmentation of retinal layers in optical coherence tomography images,” *Zenodo, DOI*, vol. 10, 2013.
- [80] M. Montazerin, Z. Sajjadifar, E. K. Pour, H. Riazi-Esfahani, T. Mahmoudi, H. Rabbani, H. Movahedian, A. Dehghani, M. Akhlaghi, and R. Kafieh, “Livelayer: A semi-automatic software program for segmentation of layers and diabetic macular edema in optical coherence tomography images,” *arXiv preprint arXiv:2003.05916*, 2020.