

Blind Denoising Algorithms for Image Restoration

Chris Fox, Matt McBrien, Zechen Lu¹

¹School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA

Three common degradation models found in images are additive Gaussian noise, multiplicative speckle noise, and motion blur. This paper presents three solutions to these three degradation modes, in which the parameters of the model are unknown, i.e. blind denoising. The additive Gaussian restoration model uses a block frequency domain approach to estimate the original image's power spectral density and process the coefficients according the estimated noise probability distribution. The speckle noise approach builds upon a popular additive denoising method, namely the Non-Local Means Denoising method, by adding clustering. The motion blur approach uses a Wiener filter and objective measurements to estimate the blurring kernel. All three approaches prove to be successful in addressing their respective degradation modes.

Additive Noise | PSD | Multiplicative Noise | Clustering | Motion Blur | Wiener Filter

Correspondence: chrisfox@gatech.edu, mmcbrien3@gatech.edu, zlu89@gatech.edu

Introduction. The topic of this paper is blind estimation and restoration of typical image degradation models. The three models selected for analysis are Additive Noise, Speckle Noise, and Motion Blur. The degradation model is assumed known a priori, so the blind component of this paper is determining the parameters of the individual models.

Additive noise is modeled by:

$$D = I + n \quad (1)$$

Where D, I, and n represent the degraded image, the original image, and a Gaussian distribution with mean 0 and variance σ^2 , respectively. A standard approach to removing additive noise is the use of the optimal Wiener Filter for the given noise variance. This paper uses the performance of the Wiener Filter as a point of comparison. Other research into additive noise reduction propose a block-based Singular Value Decomposition approach (1). Our proposed algorithm uses a block-based (Discrete Cosine Transform) DCT approach.

Speckle noise is defined by the following equation:

$$D = I + nI \quad (2)$$

Once again D, I, and n represent the degraded image, the original image, and a Gaussian distribution with mean 0 and variance σ^2 , respectively.

Speckle noise commonly appears in different imaging technologies, such as SAR (2) and ultrasound (3). Its prevalence in these applications make it a good candidate for denoising research.

Speckle noise adds noise to an image differently than additive

noise in that it affects the image unevenly. Areas of the image that are flat and bright will generally have a large amount of noise and regions that are flat and dark will generally have little noise. Regions of the original image that are high frequency will continue to be high frequency after the noise has been added.

Most methods of speckle noise reduction have focused on either using wavelet transforms, such as (4) and (5), or using temporal-adaptive filters which process multiple images of the same scene for noise reduction, such as (6). The focus of our work is to show how breaking a single image into clusters gathered from statistics from the spatial domain can be a boon for denoising techniques.

Motion Blur is defined by the following equation:

$$g(x, y) = f(x, y) * h(x, y) \quad (3)$$

where $f(x, y)$ is the original image, $h(x, y)$ is the blurring point spread function, and $g(x, y)$ is the degraded image. Motion blur is caused by relative motion between the camera and the object being captured (7). Many images are captured everyday that have considerable amounts of motion blur. Common examples include traffic cameras that capture moving vehicles and pictures of athletes in motion. These applications make motion blur a good candidate for restoration research.

Motion blur adds blurring by convolution of the image with a blurring point spread function. The convolution creates an effect that makes the objects in the image seem to streak across the image.

In the blind case, the blurring point spread function is unknown, meaning the length and angle of the blurring are both unknown. Previous methods have used Radon transforms, Cepstral Method (7), and sparse approximation (8) to measure the motion blur kernel. The focus of our work is to identify if we can use the blurred image as a frame of reference with noise level to identify the motion blur kernel.

Methodology.

Additive Gaussian Noise: Our algorithm uses a transform domain approach in attempt to estimate and isolate the noise. The processing steps are outlined in the block diagram below:

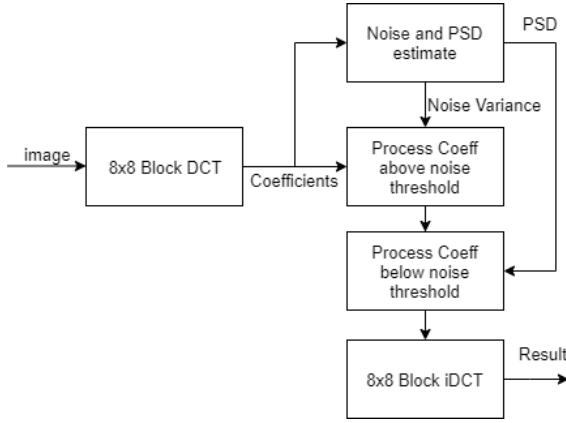


Fig. 1. Additive Noise Block Diagram

First the 8x8 block DCT coefficients are computed and ordered according to the same zigzag sequence used by the JPEG standard (9). Next, the Power Spectral Density (PSD) of the noisy image is calculated. Given the fact that the PSD of AWG Noise is flat, the noise variance can be estimated by the smallest coefficients in the noisy PSD. Further more, an estimate for the original PSD is obtained by subtracting the noise variance from the noisy PSD. Following this estimation step, a two pass processing approach is applied to the noisy DCT coefficients.

First Pass: Process the statistically significant coefficients first determined by a noise threshold. The expected value of a DCT coefficient of AWG noise is given by:

$$\mathbb{E}(\text{AWG coeff}) = \sigma_n \sqrt{2} \quad (4)$$

The threshold for significant coefficients is chosen as

$$\text{significant if } >= 2\sigma_n \quad (5)$$

These coefficients are likely to be dominated by the original image. In the first pass, all coefficients that pass the threshold are averaged with other coefficients from the local 5x5 region of blocks that are within the range of the current coefficient \pm the threshold

Second Pass: Now the coefficients close to the noise level need to be processed. It is unlikely to get a good estimate of the original since the noise is high compared the value, so the overall PSD estimate is used instead. It turns out that processing the significant coefficients first accounts for the majority of the PSD structure. Therefore in the second pass, all noisy coefficients are scaled such that the estimated overall PSD is matched. Important note: often times there is no energy left to allocate and this results in setting that specific coefficient to zero.

Final step: The final step is to compute the 8x8 block iDCT to retrieve the processed result.

Multiplicative Speckle Noise: Our speckle denoising technique builds upon a preexisting denoising technique by adding clustering. The method chosen for extension is the popular Non-Local Means Denoising, implemented by OpenCV as the fast NL means denoising method, q.v. (10). This technique assumes the noise is additive Gaussian noise. This method works by evaluating each pixel in the degraded

image and searching in a search window for similar pixels. The output pixel is a weighted average of the original pixel and the pixels in the search window. The strength of the weighting algorithm is based upon the similarity of the original pixel and the pixel in the search region and a parameter h . Specifically, the weighting algorithm is:

$$w(p, q) = e^{-\frac{\max(d^2, 0)}{h^2}} \quad (6)$$

, where w is the weighting function, p is the current pixel, q is the pixel in the search region, d is related to the similarity between the pixels at p and q , and h is set by the researcher. This leads to the following results: 1) when many similar pixels are found in the search region, the weighting function is large and 2) the weighting function is directly correlated with the h parameter. When the weighting function is large, the denoising method tends to blur a regions to a single color, and when the weighting function is small, the denoising method tends to maintain much of the information from the noisy image.

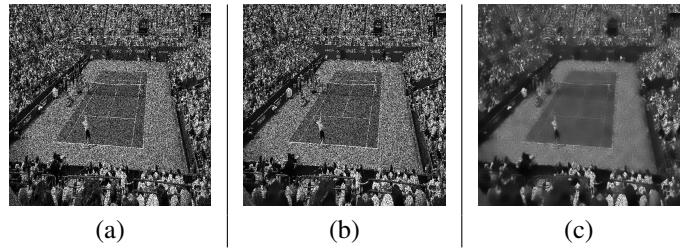


Table 1. (a) The original image degraded with speckle noise with $\sigma^2 = 0.4$. (b) The degraded image restored with the Non-Local Means Denoising method and a low h value. (c) The degraded image restored using the original Non-Local Means Denoising method and a high h value.

This method can work well when the noise is consistent across an entire image, given the h parameter is set correctly. The failure of an h value that is too small or too large can be seen in table 2. With a low h value, the tennis court retains too much high frequency noise; with a high h value, the crowd has been smoothed over.

With additive Gaussian noise, the h parameter can be set once to achieve the appropriate denoising effect. In the case of speckle noise, the noise is not consistent across the entire image. The proposed solution to this problem is to add clustering to dynamically set the value of the h parameter based on the statistics of the image region.

The two main tasks for solving this problem then become to determine the appropriate number of clusters in an image and how to set the value of the h parameter based on these clusters. In the proposed method, the number of clusters is determined by clustering the image 4 times with a different number of clusters from 2 to 5. Clustering is performed by breaking the image into 16x16 blocks and using the k-means clustering algorithm. The validity of the clustering is determined by calculating the silhouette coefficient on each clustering result. The clustering result with the best coefficient is selected to proceed with the clustering algorithm. The clustering is performed on the following statistics:

- the variance of the block

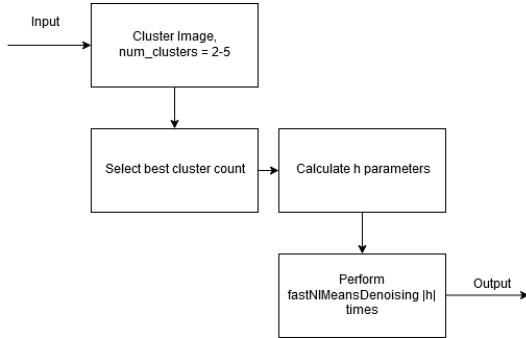


Fig. 2. A block diagram of the speckle denoising method.

- the absolute value of the difference between the mean and median of the block
- the 55th percentile of the block
- the 45th percentile of the block

After clustering, the h parameter is set for each cluster based upon the center of each cluster. The algorithm for setting the h parameter is:

$$h_i = 575v_id_i \quad (7)$$

, where h_i is the h parameter for the i th cluster, v_i is the center of the i th cluster in the variance dimension, and d_i is the difference between the i th cluster's 55th percentile and 45th percentile. Now the h parameters of each block have been set. These h parameters can be very different from block to block which commonly leads to some obvious blocking effects. This is remediated by performing a Gaussian blurring across the h parameters before performing the Non-Local Means Denoising method. Once all of the runs of the denoising method have been run across the different clusters with the associated h value, the different outputs are combined to create the final denoised image.

Motion Blur: Our motion blurring correction technique involves determining the correct motion blur kernel, and utilizing a Wiener filter with the motion blur kernel on the degraded image. If a Wiener filter with an incorrect blurring kernel is applied, the incorrect Wiener filter introduces artifacts, which causes the noise level of the image to increase. Conversely, if a correct blurring kernel Wiener filter is applied, the Wiener filter introduces minimum noise artifacts, meaning the noise of the image is at a minimum. If the noise level is higher using both an undersized and oversized blurring kernel when compared to the correct blurring kernel, the correct blurring kernel is at a local minimum in noise level.

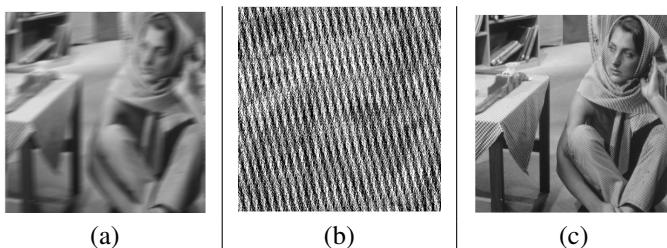


Table 2. (a) The motion blurred Barbara image. (b) The motion blurred Barbara image with an applied Wiener filter with an incorrect motion blur kernel. (c) The motion blurred Barbara image with an applied Wiener filter with the correct motion blur kernel.

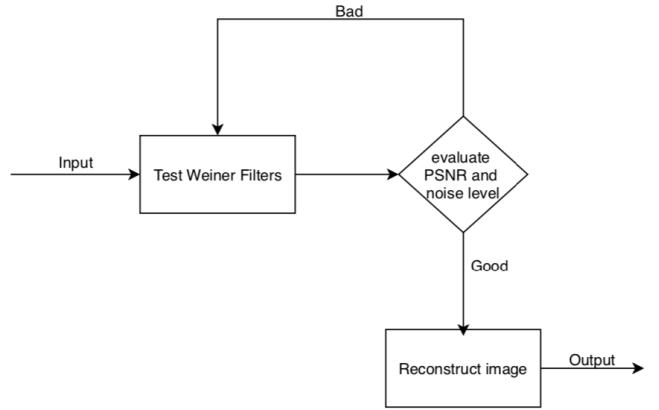


Fig. 3. A block diagram of the motion deblur.

If we assume that the Mean Squared Error of the blurred image when compared to the non-degraded image, does not exceed the peak signal value of the non-degraded image, we can assume that the the blurred image has a PSNR of at least 20dB when compared to the non-degraded image.

$$PSNR = 20\log_{10}(\text{peak value}/\text{MSE}) \quad (8)$$

In addition, if we assume that the reconstructed image using the correct motion blur kernel is close to the non-degraded image, then we can conclude that the reconstructed image using the correct motion blur kernel will have a PSNR of at least 20dB when compared to the degraded blurred image.

So, when the Wiener filter with the correct blurring kernel is applied, the reconstructed image should have a PSNR of at least 20dB when referencing the degraded image and at a local minimum in noise level. So, we can determine the correct blurring kernel by iterating through Wiener filters with various motion blurring kernels until the reconstructed image has a PSNR of at least 20dB when referencing the degraded image and is at a local minimum in noise level, and returning that blurring kernel. We utilized the blind noise estimation technique detailed in (11) to measure the noise level of the image. The method uses covariance matrices of image patches to estimate a Gaussian noise variance of the image and return a noise level based on the Gaussian noise variance.

Results. Additive Gaussian Noise: Table 3 contains an example processing result for additive noise. An example of the intermediate processing steps with figures can be found in Appendix A. The important results to note are high frequency coefficients are preserved because each coefficient is treated independently. Furthermore, the difference image has only a small amount of structure and closely resembles the underlying noise so the algorithm is successfully separating much of the noise. Additionally, more results can be found in the Analysis Section when addressing the blocking artifacts introduced by the process.

Multiplicative Speckle Noise: The results of the speckle denoising method vary from image to image. Images that have clear, blocked regions of both high frequency and low fre-

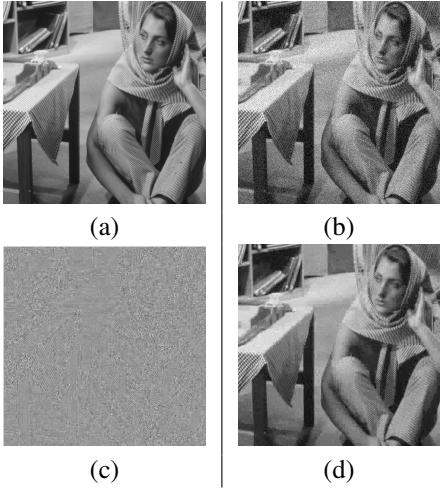


Table 3. (a) The original Barbara, undegraded image. (b) The original image degraded with additive noise with $\sigma^2 = 0.01$ (PSNR = 20.03, SSIM = 0.393). (c) The difference image between processed image and original. (d) The processed image (PSNR = 26.46, SSIM = 0.719).

quency tend to produce a high fidelity output. Images that have regions that overlap or do not align easily to blocking tend to not produce as good of results.

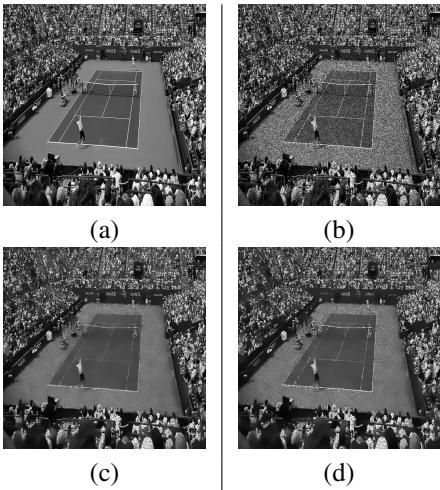


Table 4. (a) The original tennis, undegraded image. (b) The original image degraded with speckle noise with $\sigma^2 = 0.2$ (PSNR = 15.82, SSIM = 0.59). (c) The degraded image restored with the new clustering method (PSNR = 18.05, SSIM = 0.76). (d) The degraded image restored using the original Non-Local Means Denoising (PSNR = 17.67, SSIM = 0.71).

The tennis image in table 4 is an example of an image that works well with our restoration method. The crowd in the original image is a high frequency block and the tennis court is a flat region.

Notably, the new method proposed in this paper outperforms the original Non-Local Means Denoising method on both PSNR and SSIM, regardless of the level of noise.

Motion Blur: A wide range of angles and lengths for motion blur were tested. The results of the motion deblur method generally return a reconstructed image with a PSNR greater than 50dB when compared to the non-degraded image and a SSIM approximately equal to 1.

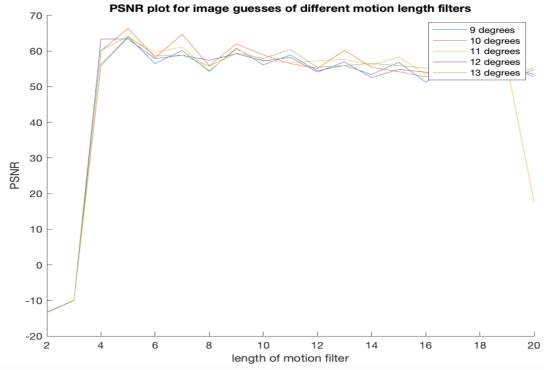


Fig. 4. A plot of PSNR plot for reconstruction of degraded images with various blurring kernels. The x-axis indicates the length of the blurring kernel that was tested. The legend indicates which angle was tested. The y-axis indicates the PSNR of the image reconstruction when compared to the non-degraded image.

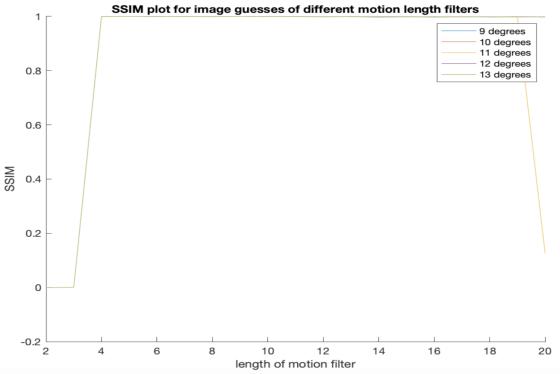


Fig. 5. SSIM plot for reconstruction of degraded images with various blurring kernels. The x-axis indicates the length of the blurring kernel that was tested. The legend indicates which angle was tested. The y-axis indicates the SSIM of the image reconstruction when compared to the non-degraded image.

Analysis. Additive Gaussian Noise: The results for the additive noise algorithm demonstrates that processing frequency domain coefficients independently is a valid approach. However the block method implemented introduces quite noticeable blocking artifacts to the denoised image. In order to mitigate these artifacts, the same algorithm detailed in the Methodology section can be run multiple times when using different definitions of 8x8 block origin locations. That is 8x8 artifacts are reduced to 4x4 artifacts when running the algorithm 4 times with blocks defined at (0,0),(4,0),(0,4),(4,4). Any blocks that extend past the region of support are not processed by that iteration. Finally, the resulting denoised images from each block definition are pixel-wise averaged together to provided an even better denoised image. Given the original size was 8x8, there are 64 unique locations to define blocks to process.

An example of the effects of processing offset blocks is shown in Table 5. Note: all offset locations were defined on a square grid extending the pattern described for the 4x case. Seeing the results from the table, the initial 4x case provides the most improvement. There is subsequent improvement as the number of offset processed increases, but the benefit diminishes while the processing time increases greatly. As the Table 5 shows, edges are preserved with this denoising approach. Averaging the offset blocks together causes the co-

efficients originally set to zero to be reintroduced in order to represent the block boundaries being averaged together. Interestingly, the resulting coefficients are quite accurate considering the given original noise level. More on this is available in Appendix A.

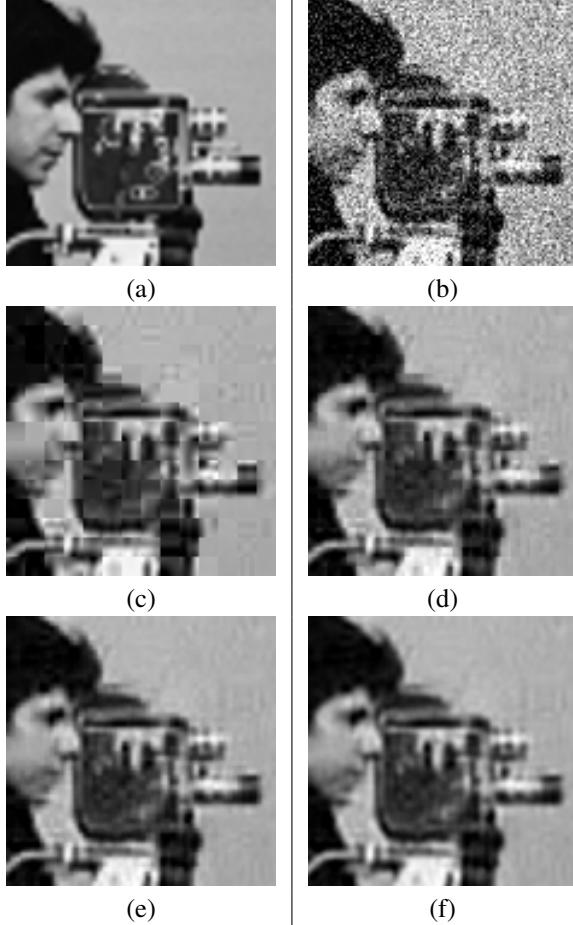


Table 5. (a) The original cameraman image, zoomed in. (b) The original image degraded with additive noise with $\sigma^2 = 0.025$ (PSNR = 16.02, SSIM = 0.145). (c) The degraded image restored with the a single block method (PSNR = 27.07, SSIM = 0.659). (d) The degraded image restored using blocking method 4x (PSNR = 28.66, SSIM = 0.735). (e) The degraded image restored using blocking method 16x (PSNR = 29.28, SSIM = 0.767). (f) The degraded image restored using blocking method 64x (PSNR = 29.58, SSIM = 0.781).

So far only exemplar cases for additive noise processing have been discussed in this paper. So considering everything that has been covered, Figure 6 consolidates the PSNR results of processing over a wide range of noise variances. In addition, the optimal 8x8 Wiener filter for that noise variance is included for comparison. Figure 7 shows the same for the SSIM results. Interestingly, the PSNR plot for the block process and Wiener filter are very close together. The SSIM plot diverges slightly as the noise gets large with the Wiener filter performing better. This is the blocking effects coming into play. Just increasing to the 4x process performs better than the Wiener filter at every variance. As seen in Table 5, increasing further to the x16 and x64 provide marginal improvements above 4x.

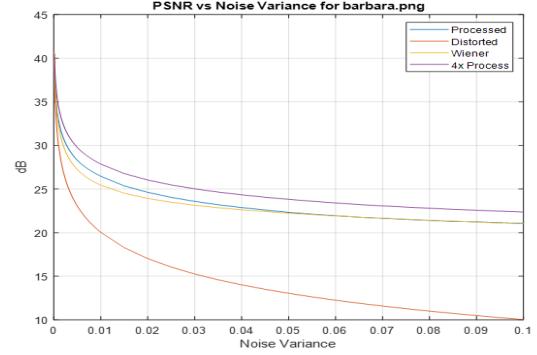


Fig. 6. A plot of PSNR against variance of additive noise for the image Barbara. The red line displays the PSNR of the degraded image. The yellow line displays the PSNR of the degraded image after being restored by optimal 8x8 Wiener Filter. The blue line displays the PSNR of the degraded image after being restored by a single block process. The purple line displays the PSNR of the degraded image after being restored by 4x block processes

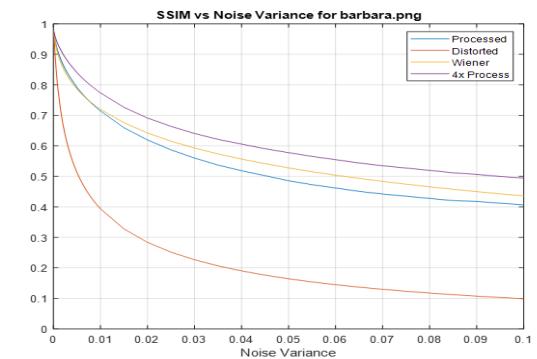


Fig. 7. A plot of SSIM against variance of additive noise for the image Barbara. The red line displays the SSIM of the degraded image. The yellow line displays the SSIM of the degraded image after being restored by optimal 8x8 Wiener Filter. The blue line displays the SSIM of the degraded image after being restored by a single block process. The purple line displays the SSIM of the degraded image after being restored by 4x block processes

Multiplicative Speckle Noise: The results of our speckle denoiser show the value of using spatial-based clustering to perform denoising. Speckle noise is similar to additive noise, but it does not uniformly affect the image. Similarly, our denoising method is like an additive noise denoising method but it uses clustering to denoise different parts of the image to different degrees.

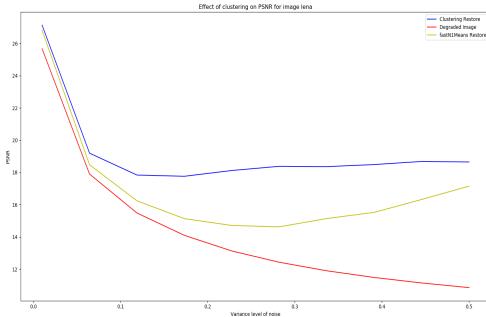


Fig. 8. A plot of PSNR against variance of speckle noise for the image Lena. The red line displays the PSNR of the degraded image. The yellow line displays the PSNR of the degraded image after being restored by the Non-Local Means Denoising method. The blue line displays the PSNR of the degraded image after being restored by the new clustering method.

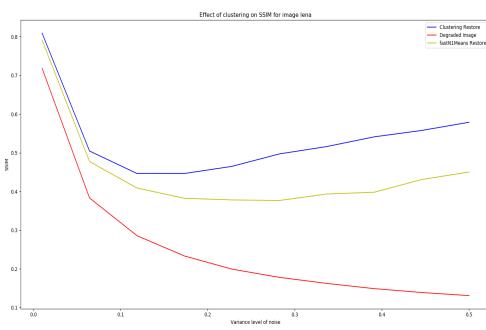


Fig. 9. A plot of SSIM against variance of speckle noise for the image Lena. The red line displays the SSIM of the degraded image. The yellow line displays the SSIM of the degraded image after being restored by the Non-Local Means Denoising method. The blue line displays the SSIM of the degraded image after being restored by the new clustering method.

Regardless of image type or level of noise, the new clustering denoising algorithm outperformed the original Non-Local Means Denoising method. In figures 8 and 9, it is shown that no matter the level of noise, the new clustering method outperforms the original method in both PSNR and SSIM measurements. This shows the validity of clustering images based upon spatial statistics and this technique could be applied to other spatial denoising techniques to achieve better results.

Computation time suffers under the new algorithm. If the original Non-Local Means Denoising method takes t_d seconds to compute, the new algorithm will take approximately $|h_i|t_d + t_c$ seconds to compute where $|h_i|$ is the number of h parameters set across the image and t_c is the time taken to determine the best number of clusters. The number of h parameters can be larger than the number of clusters after the h parameters are blurred to remove the blocking effects.

The most challenging aspects of implementing such a system are to determine what statistics are the most useful for performing the clustering and how to formulate the algorithm to set the h parameters. The methods proposed in this paper started with beliefs about how multiplicative noise would affect different regions of an image and how those effects would be visible in the statistics of a block. Tweaking the

parameters to produce consistent results was the final step. There is no reason to believe that the statistics selected for this proposed method or the final equation used for determining the h parameters are the ideal values and more work could be done to determine how far this method could be pushed by altering parameters.

It is worth noting that while this method outperforms the original in terms of SSIM and PSNR regardless of image, some images do produce poor results. These images tend to have many high frequency edges and the different regions of the image do not fall neatly into blocks. For example, the Barbara image, featured in table 6, has noticeable problems in its denoising. This method could be improved upon by investigating how block size could affect these images and tweaking both the clustering algorithm and the h parameter algorithm to produce better results.

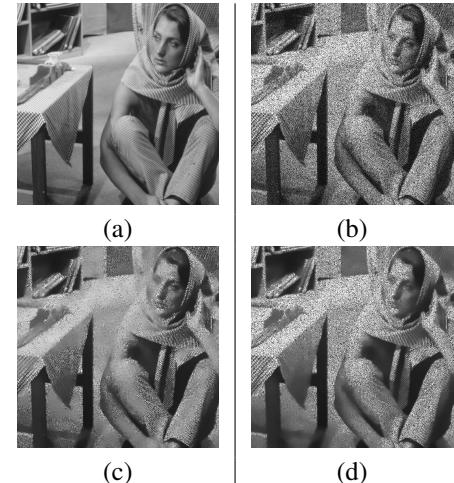


Table 6. (a) The original barbara, undegraded image. (b) The original image degraded with speckle noise with $\sigma^2 = 0.2$ (PSNR = 13.93, SSIM = 0.31). (c) The degraded image restored with the new clustering method (PSNR = 18.16, SSIM = 0.55). (d) The degraded image restored using the original Non-Local Means Denoising (PSNR = 15.60, SSIM = 0.51).

Motion Blur: The results of our deblurring technique show the value of using the blurred image as a reference image and the noise level to restore the image. As seen from the results, the restored image from our algorithm generally has a SSIM close to 1 and high PSNR for blurring kernels that had a length greater than 3. The blurring kernels with a length of 3 or less failed our method.

The blurring kernels of length 3 or less gave a very small blurring degradation to the original image. The small blurring causes the Wiener filters to overcompensate more easily, which causes the measured local minimum noise level to occur early. Similarly to a small length motion blurring kernel, a short length Wiener filter adjusts the blurred image minimally, which means the reconstructed image of a small length Wiener filter will be similar to the blurred image, causing the PSNR between the reconstructed image and the blurred image to be above 20dB. These two facts cause our method to reconstruct the image prematurely with the wrong Wiener filter utilizing an undersized blurring kernel.

For blurring kernels of length greater than 3, the blurring be-

comes more evident and thus the premature nature of the small length motion blurring kernel does not occur. In our test set of motion blurring kernels of angles between 9 and 13 degrees and lengths between 2 and 20, the motion blurring kernel of 13 degree angle and 20 length was the only other case where the image was incorrectly reconstructed. For this case, the reconstruction method reached an edge case where the PSNR comparison between the wrongly reconstructed image and the blurred image was 21dB, barely above the 20dB threshold, and the noise level had reached a local minimum noise level where the noise level was barely less than the blurring kernels tested alongside it.

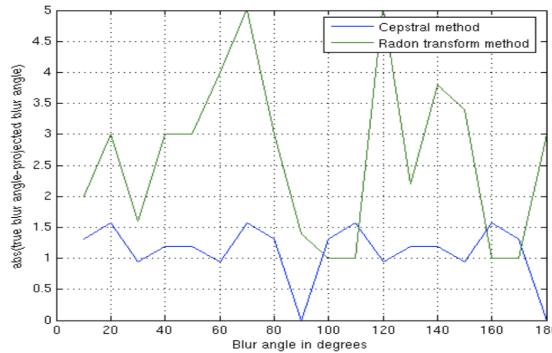


Fig. 10. A plot of the average error in angle estimation for the cepstral and Radon transform methods when the length of the blurring kernel is 10.

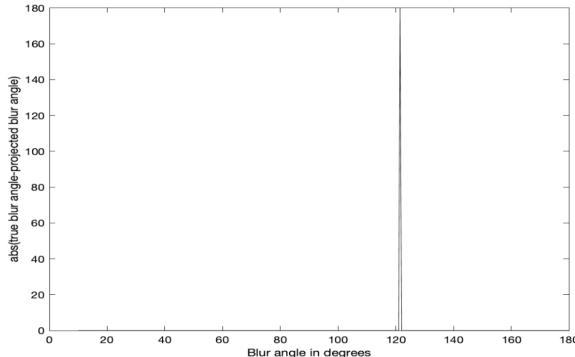


Fig. 11. A plot of the average error in angle estimation for our blurring restoration method when the length of the blurring kernel is 10.

The cepstral and Radon transform motion blur kernel prediction techniques from (7) for motion blurring kernel with length 10 usually predict a kernel within 2 degrees of the actual blurring kernel as seen in Figure 10. Our method when given an input length of length 10 is able to accurately obtain the angle for most cases as shown in figure 11.

The one case that our method failed in our image restoration was for the restoration of a degraded image with a blurring kernel of length 10 and angle of 121.5 degrees. For this case, the tested Wiener filter with a blurring kernel of length 10 and angle at 121.5 degrees failed the local noise minimum condition, as the test case before it, utilizing a blurring kernel of length 10 and angle at 121 degrees, had very minimal measured noise.

This case is an outlier of the 341 angles tested between 10

degrees and 180 degrees, but it does show that the local minimum noise condition may not always be applicable. Further conditions may need to be placed on the image restoration in regards to the noise level, such as increasing the definition of what can classify the noise level as local minimum for the image restoration.

Computation time of our process for blurring restoration is slower than the cepstral and Radon transform methods. While the cepstral and Radon transform methods are strictly a calculation, our method needs to account for the iterations of numerous Wiener filters.

If the angle and length of the blurring kernel are both not known, then our max computation time can be computed as the product of the total number of lengths tested, the total number of angles tested, and the time it takes to test one Wiener filter. The more lengths and angles are tested, the more accurate the reconstruction will be.

References

- K. Konstantinides, B. Natarajan, and G. S. Yovanof, “Noise estimation and filtering using block-based singular value decomposition,” *IEEE Transactions on Image Processing*, vol. 6, pp. 479–483, March 1997.
- L. D. Robertson, A. Davidson, H. McNairn, M. Hosseini, S. Mitchell, D. de Abelleira, S. Verón, and M. H. Cosh, “Sar speckle filtering and agriculture field size: Development of sar data processing best practices for the jecam sar inter-comparison experiment,” in *IGARSS 2018 - 2018 IEEE International Geoscience and Remote Sensing Symposium*, pp. 3828–3831, July 2018.
- H. Choi and J. Jeong, “Speckle noise reduction in ultrasound images using srad and guided filter,” in *2018 International Workshop on Advanced Image Technology (IWAIT)*, pp. 1–4, Jan 2018.
- A. Aghabalaee, Y. Amerian, H. Ebadi, and Y. Maghsoudi, “Speckle noise reduction of time series sar images based on wavelet transform and kalman filter,” in *IGARSS 2018 - 2018 IEEE International Geoscience and Remote Sensing Symposium*, pp. 625–628, July 2018.
- S. Solbo and T. Eltoft, “A stationary wavelet-domain wiener filter for correlated speckle,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 46, pp. 1219–1230, April 2008.
- T. Yassine, S. Ahmed, and N. Abdelkrim, “Speckle noise reduction in digital speckle pattern interferometry using riesz wavelets transform,” in *2017 International Conference on Advanced Technologies for Signal and Image Processing (ATSiP)*, pp. 1–4, May 2017.
- F. Krahmer, Y. Lin, B. C. McAdoo, K. A. Ott, J. Wang, D. Widemann, and B. Wohlberg, “Blind image deconvolution motion blur estimation,” 2006.

8. J. Cai, Hui Ji, Chaoqiang Liu, and Z. Shen, “Blind motion deblurring from a single image using sparse approximation,” in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 104–111, June 2009.
9. G. K. Wallace, “The jpeg still picture compression standard,” *IEEE Transactions on Consumer Electronics*, vol. 38, pp. xviii–xxxiv, Feb 1992.
10. A. Buades, B. Coll, and J.-M. Morel, “Non-local means denoising,” *Image Processing On Line*, vol. 1, pp. 208–212, 2011.
11. X. Liu, M. Tanaka, and M. Okutomi, “Single-image noise level estimation for blind denoising,” *IEEE Transactions on Image Processing*, vol. 22, pp. 5226–5237, Dec 2013.

Appendix A

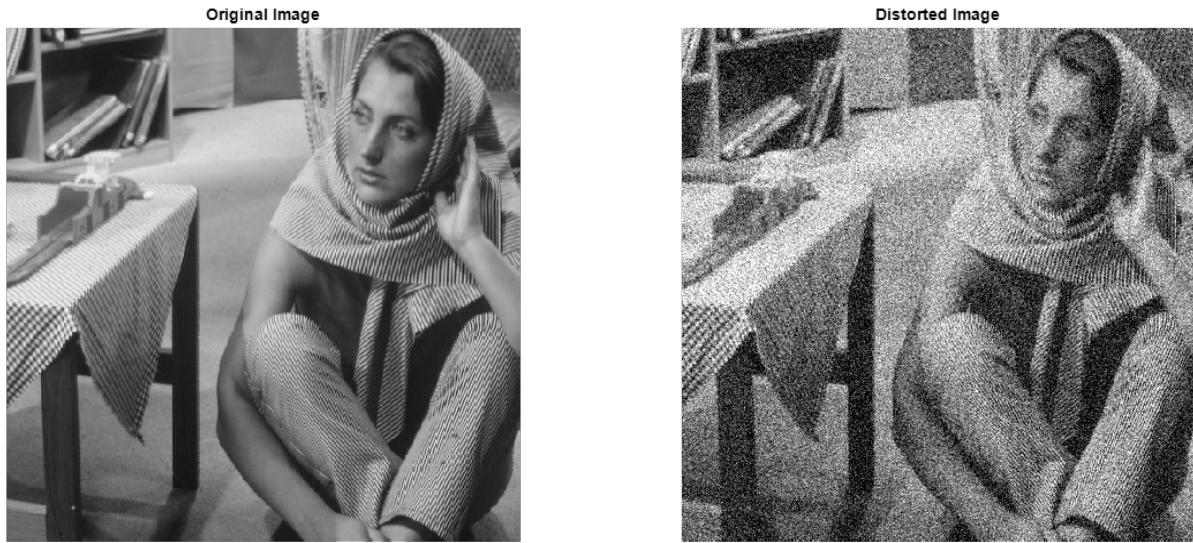


Fig. 12. Original and additive noise degraded image for barbara with $\sigma^2 = 0.025$.

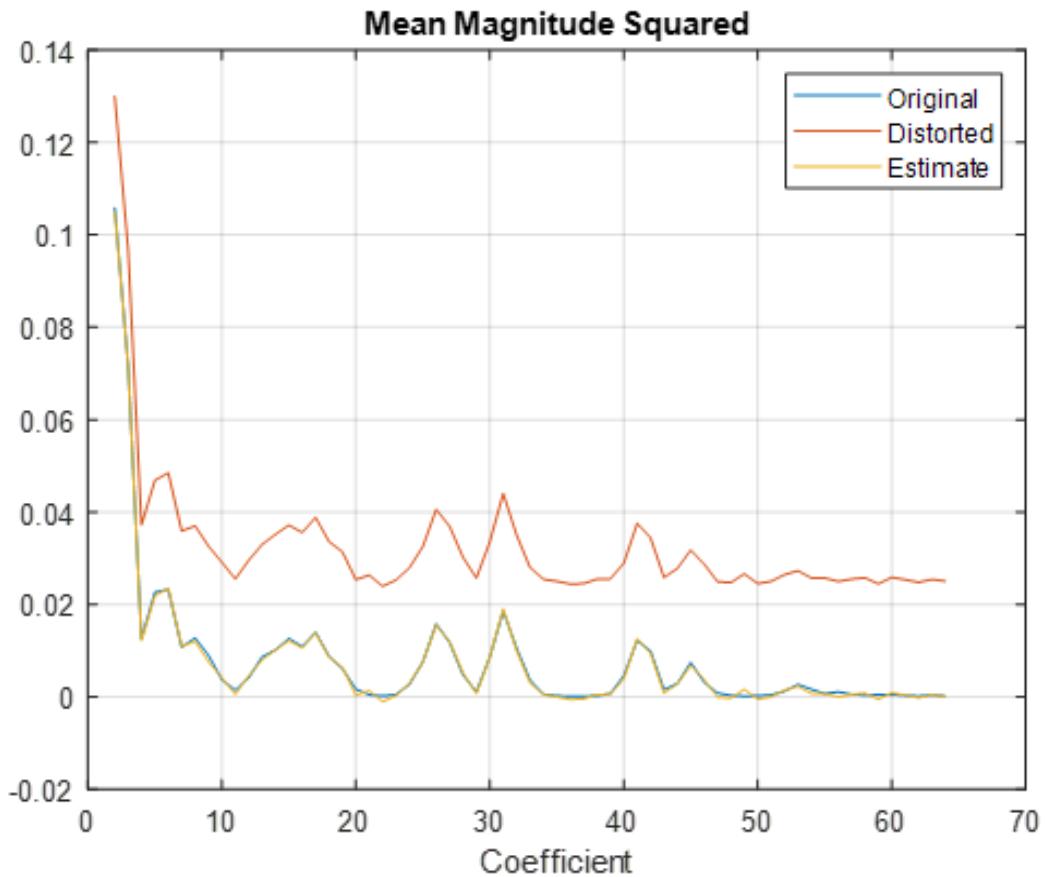


Fig. 13. PSD plots vs DCT coefficients for original and distorted image as well as estimated PSD found by subtracting noise variance estimate from distorted PSD. This results is a very faithful estimate of the original. The $\sigma^2 = 0.025$ can be seen as the minimum of the distorted PSD.

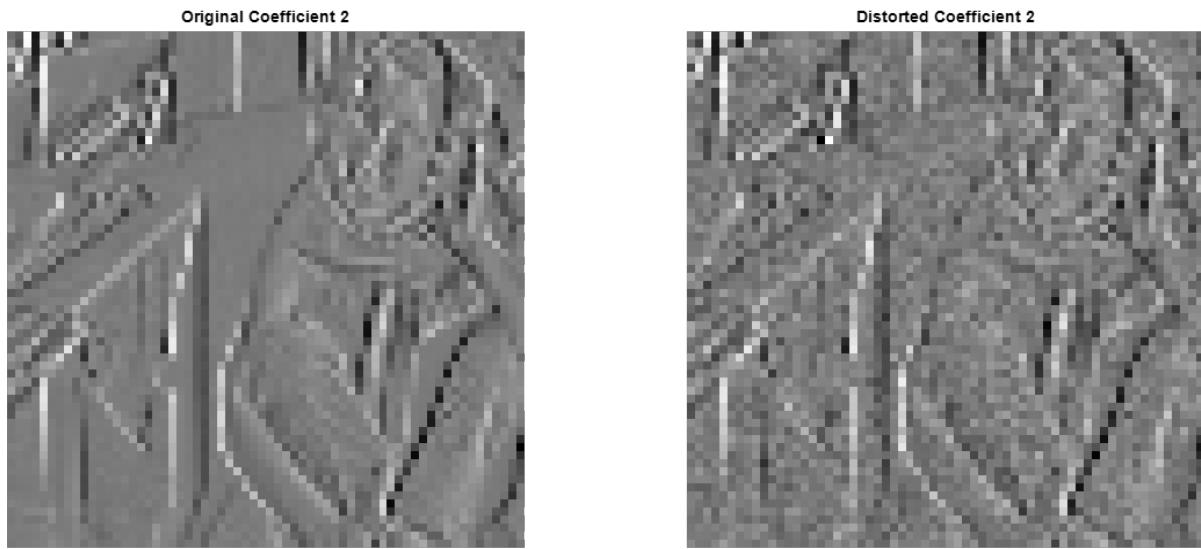


Fig. 14. Original and additive noise degraded coefficient 2. Additive noise looks the same in the frequency domain.

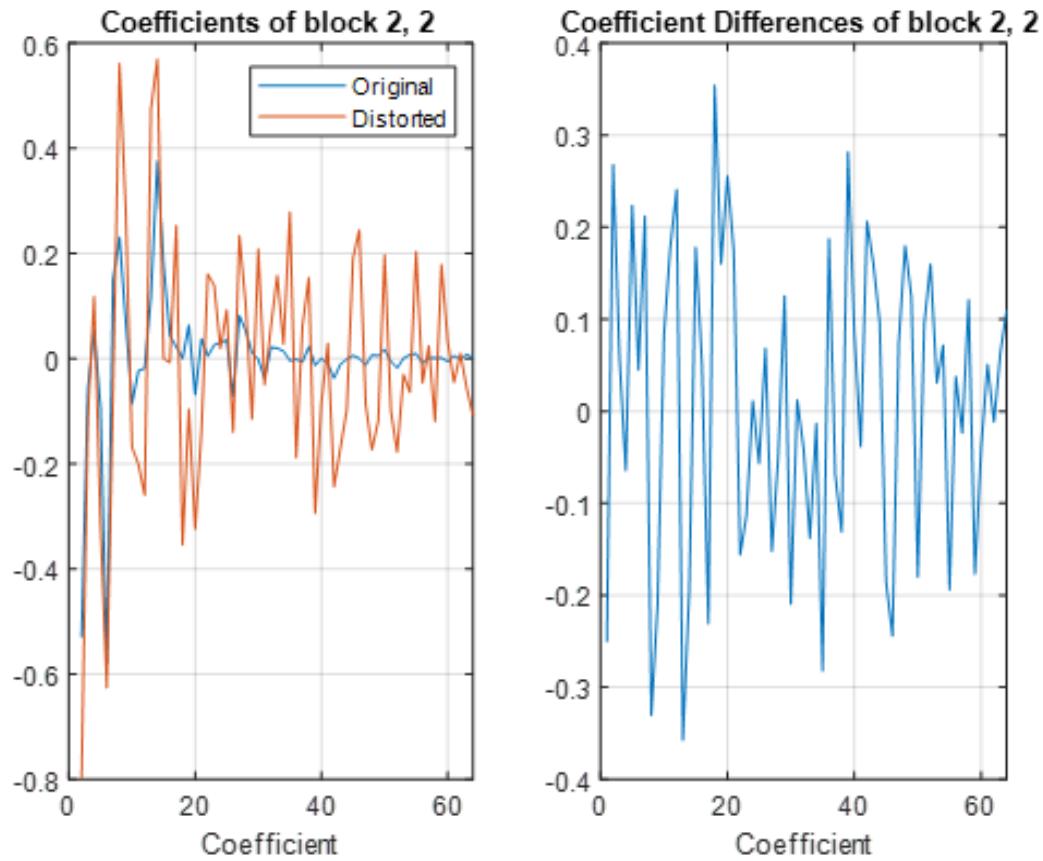


Fig. 15. Original and additive noise degraded image DCT coefficient plot for block (2,2) as well difference plot. The expected value of the noisy coefficients is approximately an order of magnitude above the high frequency coefficients.

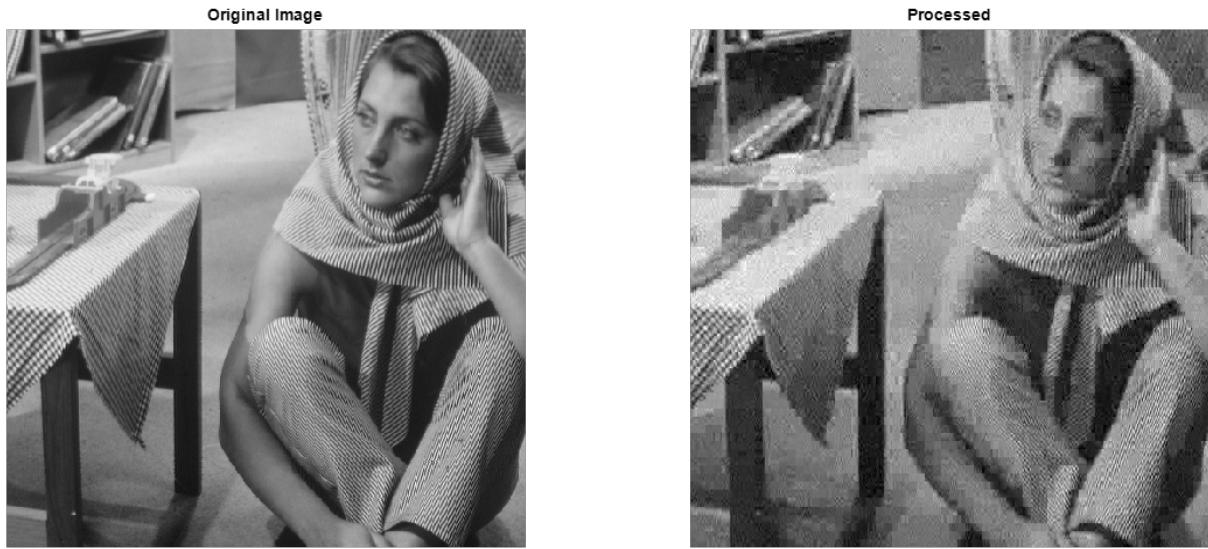


Fig. 16. Original and processed image after running first and second pass on distorted coefficients. Blocking artifacts are quite noticeable.

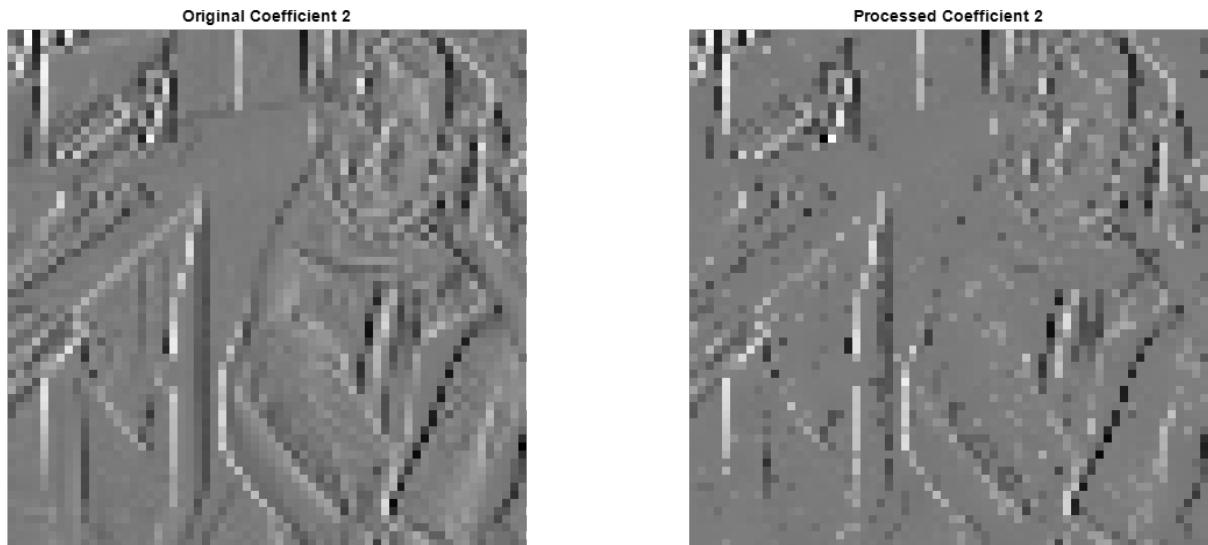


Fig. 17. Original and processed coefficient 2. Additive noise looks the same in the frequency domain. It is apparent which coefficients were recovered and which were far enough below the noise to be set to zero. Strong edges are preserved.

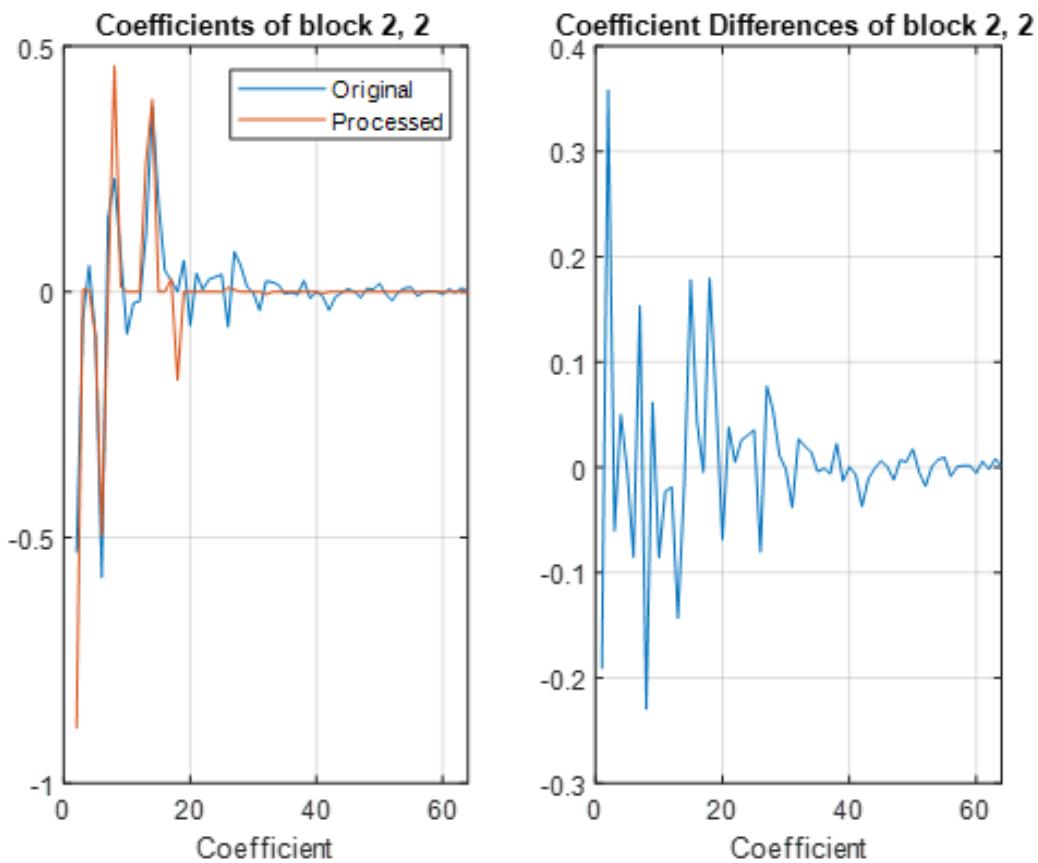


Fig. 18. Original and processed image DCT coefficient plot for block (2,2) as well difference plot. Statistically significant coefficients estimated, noisy coefficients scaled down or set to zero.

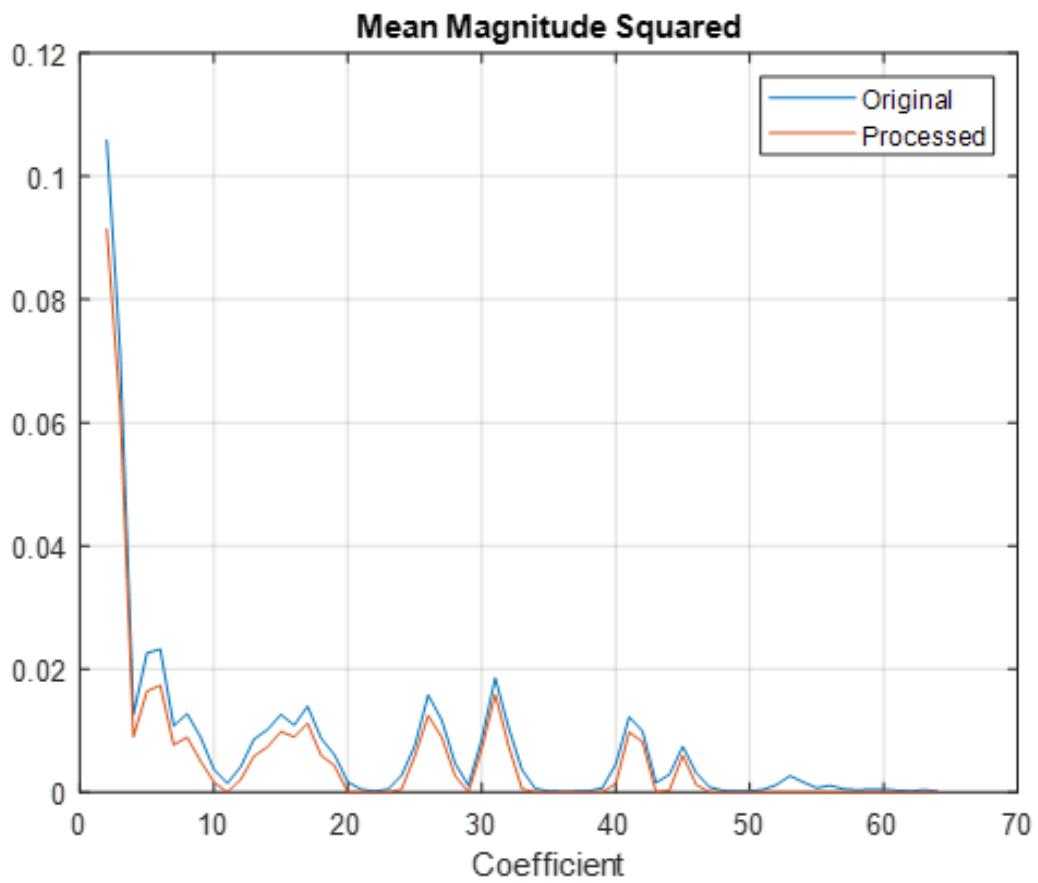


Fig. 19. PSD plots vs DCT coefficients for original and processed image. The processed result undershoots the original given because the noise variance estimate generally estimates the variance high.

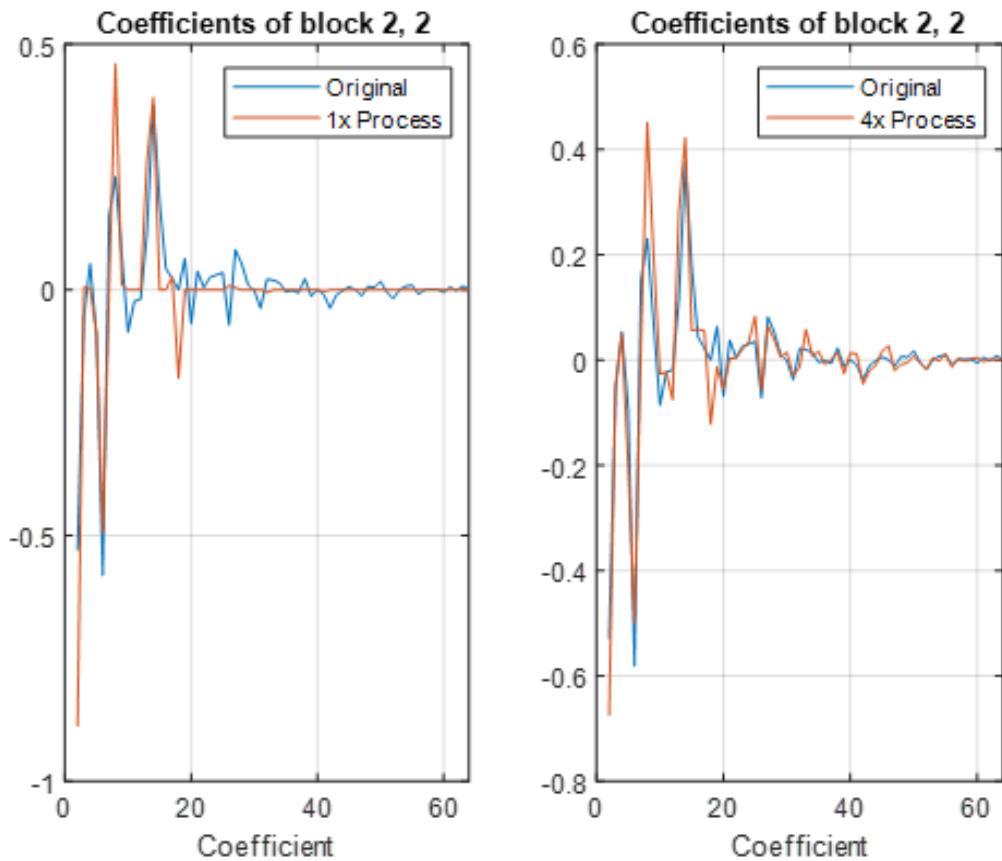


Fig. 20. Original and single processed image DCT coefficient plot for block (2,2) compared to running the algorithm 4 times with offset block indices. The small high frequency coefficients are reintroduced as the blocking artifacts are reduced.