

# Optical Coherence Tomography: Mitigation of Scattering Using DSP Techniques

Jared Mann, V00187636 and Mike Vlanich, V00862757  
*University of Victoria, Faculty of Electrical Engineering*  
 (Dated: April 21, 2018)

Optical Coherence Tomography (OCT) is a non-invasive method of producing 2-Dimensional images of biological tissues. The practice came into prominence in 1991 from the developments of Huang et al. Since its introduction, advancements in laser diode technology and imagery has allowed for micrometer resolution and is widely used in Ophthalmology.[4] Although OCT produces great resolution there are drawbacks. OCT is notorious for producing a phenomenon called speckle noise. This review will utilize previous work done using imagery filters to reduce speckle noise and as well create our own simulations to confer.

## CONTENTS

List of Figures	1
List of Tables	1
List of Code	1
I. Introduction	1
II. OCT Identified Problem and Suggested Solution	2
A. The Problem of Speckle Noise	2
B. Possible Solutions for Speckle Noise	2
III. OCT Image Aquisition and Preparation	2
A. Aquiring and Visualizing OCT Images	2
B. Slicing and Preparing a 2D Image for Processing	2
IV. Quality Metrics: Signal to Noise Level, and Estimated Noise Level	3
A. Contrast to Noise (CNR) Estimation	3
B. Estimated Noise Level	3
C. Naturalness Image Quality Evaluator (NIQE) No-Reference Image Quality Score	3
D. Omission of Mean Squared Error (MSE) for OCT Images	3
V. Scope of DSP Methods Used	3
A. Wiener Filter	4
B. Lee Filter	4
C. Kuwahara Filter	4
D. Symmetric Nearest Neighbor (SNN) Filter	5
E. Hybrid Median Filter	5
VI. Conclusion	5
References	9

## LIST OF FIGURES

1	On the left is an image put through a speckle filter in Matlab. On the right is the original image.....	2
2	3D Volume Visualization Capture .....	2
3	Unaltered 2D Slice of 3D Volume .....	3
4	OCT Captures with Increasing Numbers of Composite Images Used, Adapted From [14] .....	4
5	The Kuwahara Operator [5] .....	5
6	Symmetric Nearest Neighbor Pixel Selection [6] .....	5
7	Unfiltered and Wiener Filtered Images ....	6
8	Lee and Kuwahara Filtered Images .....	7
9	SNN and Hybrid Filtered Images .....	7

## LIST OF TABLES

I	Comparison of Filter Quality Metrics .....	6
---	--	---

## LIST OF CODE

1	Take Slice of 3D Volume .....	2
2	Estimate Contrast to Noise Ratio .....	3
3	Script to Generate Figure of Filtered Images and Print Quality Metrics .....	8

## I. INTRODUCTION

Optical Coherence Tomography (OCT) is a non-invasive method of producing 2-Dimensional images of internal tissues in a method comparable to that of ultrasound imaging.[8][18] The practice came into prominence following the developments of Huang et al. at the Massachusetts Institute of Technology in 1991. [8] Improvements have been made in the technology to provide greater depth and scanning resolution, but certain limitations and problems remain, such as the introduction of speckle noise during scanning.

## II. OCT IDENTIFIED PROBLEM AND SUGGESTED SOLUTION

### A. The Problem of Speckle Noise

A problem with using an OCT interferometer is that speckle noise exists, which ultimately reduces the quality of the image. Speckle is a phenomenon that is caused by interference of the reflected wave at the photo-detector. Speckle noise is evident when trying to image non-fluid structures (such as hard tissue). As the wave propagates it becomes distorted due to low-angle multiple forward scattering and diffuse multiple backscattering of coherent photons. [16]

In OCT interferometry imaging, photons are detected after one backscattering event for most relevant information, but when a wave is propagated through a dense biological sample it experiences numerous scattering events (in OCT interferometry). These scattering events increase the likelihood for photons to change their travel distance comparative to their airborne path which results in speckle noise reducing the image quality. [16]

To display the effects of speckling on an image refer to Figure 1. Using `imnoise(I,'speckle',v)` in Matlab to create a filter that would add multiplicative noise to an image. As seen on the left the image resolution is decreased significantly after adding speckling noise especially in the area of high light reflectivity off of the table.



FIG. 1: On the left is an image put through a speckle filter in Matlab. On the right is the original image.

### B. Possible Solutions for Speckle Noise

As technology progress in imaging processing, techniques are being developed to help reduces the effects of speckle noise when using OCT. Some of these methods include decreasing spatial and temporal coherence of the laser used. While more trivial techniques such as phase-domain processing and zero-adjustments are used to improve image quality due to speckle noise. [16]

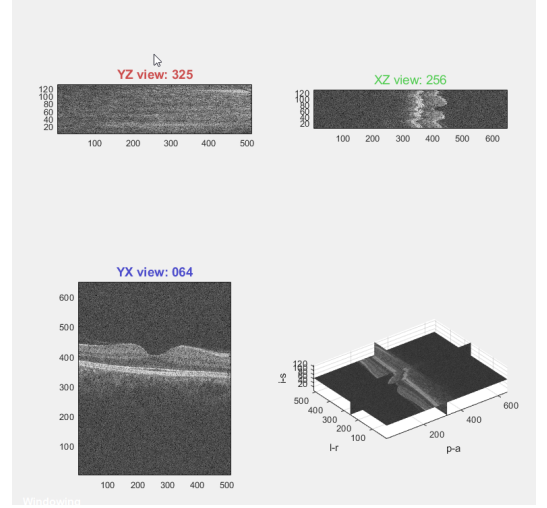


FIG. 2: 3D Volume Visualization Capture

## III. OCT IMAGE AQUISITION AND PREPARATION

### A. Aquiring and Visualizing OCT Images

While many published reports contain printed images of OCT captures, getting raw data was necessary to make realistic observations on the effects of using digital filtering. 3D OCT captures of healthy individuals were obtained from a published proceeding comparing OCT scans of healthy individuals [20]. The scans were provided in .mat file format. The size of each volume provided by these scans was  $650 \times 512 \times 128$  voxels. Each voxel had a resolution of  $11.72 \times 46.88 \times 3.54 \mu\text{m}$ . We were able to view an interactive 3D volume from the provided .mat files using a script published on MathWorks file exchange titled *3D Volume Visualization* [19]. A capture of the visualization can be seen in Figure 2.

### B. Slicing and Preparing a 2D Image for Processing

Upon importing the .mat files into Matlab, they took the form of a  $650 \times 512 \times 128$  double data type named `d3`. The Matlab code shown in Listing 1 shows the process of taking the middle XY slice and creating a gray-scale image.

```

1 %Take Slice of 3D Volume
2 SliceL = d3(:,:,64);
3 ImgSliceL = mat2gray(SliceL, [0 255]);
4 imshow(ImgSliceL);
5

```

Listing 1: Take Slice of 3D Volume

An example unaltered aquired image is shown in Figure 3.

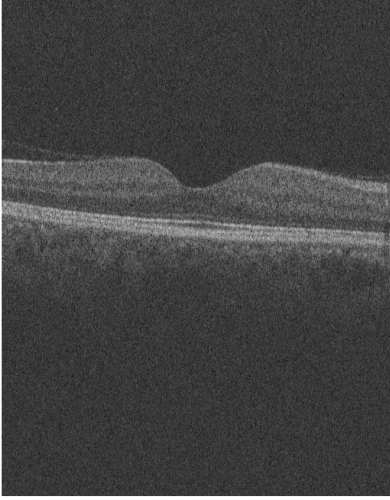


FIG. 3: Unaltered 2D Slice of 3D Volume

#### IV. QUALITY METRICS: SIGNAL TO NOISE LEVEL, AND ESTIMATED NOISE LEVEL

The results of quality measurements for each applied filter method are listed in Table I. The methods for obtaining the quality metrics are described in the following paragraphs.

##### A. Contrast to Noise (CNR) Estimation

The code shown in Listing 2 outlines the process for acquiring the contrast to noise estimation. The process starts with taking the 2D image matrix and converting it into a 1D array.

The maximum and minimum values are extracted from the 1D matrix, and then the standard deviation is calculated. The final CNR is obtained equivalent to the formula:

$$CNR = \left( \frac{Max - Min}{Standard\ Deviation} \right)$$

```

1 % Measure contrast-to-noise ratio
2 img=ImgSliceL;
3 img=double(img(:));
4 ima=max(img(:));
5 imi=min(img(:));
6 mse=std(img(:));
7 cnr=((ima-im)./mse)
8

```

Listing 2: Estimate Contrast to Noise Ratio

In terms of value dictating quality, the higher the value the better.

##### B. Estimated Noise Level

The Estimated Noise Level (ENL) is calculated using a method implemented in a Matlab function by Ashish Meshram [11]. This implementation is based off of the paper titled *Single-Image Noise Level Estimation for Blind Denoising* by Xinhao Liu, and team [9].

The paper described a patch-based noise level estimation algorithm. A patch of the image is selected based on having a lack of high frequency content. This and other selected patches are then used to provide an estimation of the noise level.

##### C. Naturalness Image Quality Evaluator (NIQE) No-Reference Image Quality Score

Due to the fact that all obtained images need to be evaluated as is, with no reference image to compare to, it is advantageous to use metrics that do not require a reference image. Matlab has a built-in function called *nique* that returns a value (lower is better, indicating higher perceptual quality) using the Naturalness Image Quality Evaluator based off the work of Anish Mittal and team [12].

The creators of this algorithm claim that the results are comparable to other No Reference Image Quality Assessment (NR IQA) models that require opinions from humans about the quality of images from large data-sets to train them.

##### D. Omission of Mean Squared Error (MSE) for OCT Images

In the analysis of filter quality, a metric called Mean Squared Error (MSE) is often used to compare an image without any distortion to an image that has been distorted and then filtered. In our analysis we do not have the ability to do this, because we have an image that has distortion inherent in its acquisition.

In the paper titled *Speckle reduction in optical coherence tomography images using digital filtering* [14] the reference image used for calculating the MSE of filters was a composite image made from using multiple scans see Figure 4 for the reference image used in the paper (where N=64). In these images the N value represents the number of scans used to make the composite.

Since we do not have access to a composite image, or an image without inherent distortion, an MSE evaluation of filtering will be omitted from our report.

#### V. SCOPE OF DSP METHODS USED

This report will focus on 2D image filtering methods which can be applied using Matlab, and due to our re-

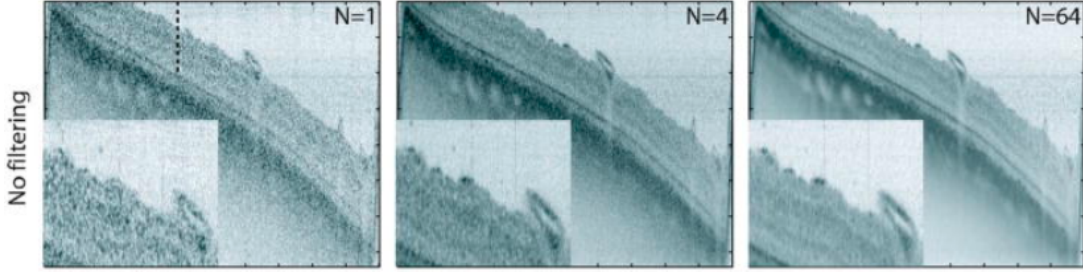


FIG. 4: OCT Captures with Increasing Numbers of Composite Images Used, Adapted From [14]

sources we will not be able to make composite captures from multiple scans.

### A. Wiener Filter

The 2D Wiener filter is a processing technique which is capable of not only improving resolution but as well signal-to-noise (SNR) ratio of particular image. This low-pass filter is an optimal trade-off between inverse filtering and noise smoothing. This filter is also referred to as a local linear minimum mean square error which means [LLMMSE] it is the optimal estimator in the sense of mean squared error (MSE). Where LLMMSE and MSE is determined by:

$$\begin{bmatrix} R_{xx}[0] & R_{xx}[-1] & \dots & R_{xx}[1-N] \\ R_{xx}[1] & R_{xx}[0] & \dots & R_{xx}[2-N] \\ \vdots & \vdots & \ddots & \vdots \\ R_{xx}[N-1] & R_{xx}[N-2] & \dots & R_{xx}[0] \end{bmatrix} \begin{bmatrix} h[0] \\ h[1] \\ \vdots \\ h[N-1] \end{bmatrix} = \begin{bmatrix} R_{yx}[0] \\ R_{yx}[1] \\ \vdots \\ R_{yx}[N-1] \end{bmatrix}$$

$$R_{ee}[m] = R_{yy}[m] - R_{yy}[m] = R_{yy}[m] - h[m] \times R_{xy}[m]$$

Where  $R_{xx}$  is the received signal,  $N$  is the length, then there are  $N$  equations in the  $N$  unrestricted values of  $h[n]$  and  $h[m]$  is the impulse response of the filter [13]. Refer to Figure 7 for the resulting image.

### B. Lee Filter

The Lee filter applies a spatial filter to each pixel which filters data based on local statistics calculated within a square window. The value of the center pixel is replaced by a value calculated using the neighboring pixels. Implements similar characteristics of the Wiener filter in the sense it uses MSE but also has edge preserving features [10]. The pixels are classified into:

Homogenous: where the pixel value is replaced by an average of the filter window

Heterogenous: where the pixel value is replaced by a weighted average

Point: where the pixel value is unchanged.

The resultant gray value  $R$  for the pixels is then represented as:

$$R = I_m \text{ for } C_i \leq C_u$$

$$R = I_m \times W + I_c \times (1 - W) \text{ for } C_u < C_i < C_{max}$$

$$R = I_c \text{ for } C_i \geq C_{max}$$

Where:

$I_m$  = mean value of intensity within the kernel

$I_c$  = center pixel in the kernel

$C_i = S/I_m$

$C_u = \sqrt{1/(\text{number of flops})}$

$W = \exp -\zeta(C_i - C_u)/(C_{max} - C_i)$

$S$  = standard deviation of intensity within the kernel [1]

Refer to Figure 8 for the resulting image.

### C. Kuwahara Filter

The Kuwahara filter is an edge preserving non-linear smoothing filter. It operates by taking a square window of some pixels in an image. The size of this window is constrained to the formula  $2a + 1$ , where  $a$  is an integer. This window can be divided into 4 smaller regions as shown in Figure 5 [15]. The letters (a,b,c,d) correspond to the regions. The highlighted center cross of the image shows the overlapping regions. The center pixel will take the mean value that is most common amongst all sub regions. The Kuwahara filter is a sliding window filter, meaning that this process will be continued over the entire image.

For processing the OCT image, we used a Matlab implementation by Luca Balbi [2].

Refer to Figure 8 for the resulting image.

a	a	a/b	b	b
a	a	a/b	b	b
a/c	a/c	a/b/c/d	b/d	b/d
c	c	c/d	d	d
c	c	c/d	d	d

FIG. 5: The Kuwahara Operator [5]

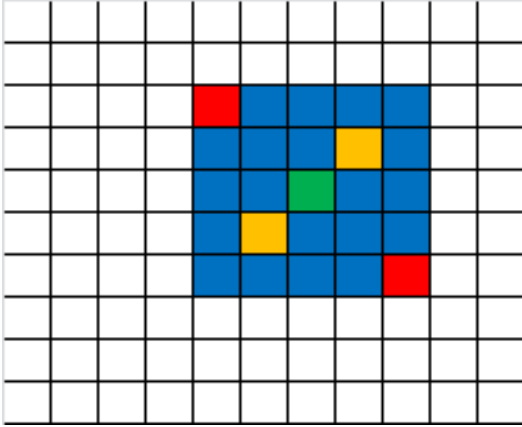


FIG. 6: Symmetric Nearest Neighbor Pixel Selection [6]

#### D. Symmetric Nearest Neighbor (SNN) Filter

The SNN filter is another sliding window filter that (like Kuwahara) aims to preserve edges, while decreasing noise. At every pixel the corners of the window are compared to the center pixel. The center pixel is the calculated mean sum of the selected pixels as shown in Figure 6 [6].

The Matlab implementation by Art Barnes was chosen for applying this filter method to the OCT images [3]. Refer to Figure 9 for the resulting image.

#### E. Hybrid Median Filter

The Hybrid Median Filter is a nonlinear filter with better edge preserving characteristics than just an ordinary median filter. This is done by ranking spatial directions individually. This particular filter, filters matrix A using an  $N \times N$  box. To apply the median filter three (3)

medians are calculated in the  $N \times N$  box. Where MR is the median of the horizontal and vertical R pixels, MD is the median of the diagonal D pixels. Using the 2 median and the central pixel value gives the filter value [17]. For a  $N = 5$  matrix would give the following matrix:

$$A = \begin{bmatrix} D & * & R & * & D \\ * & D & R & D & * \\ R & R & C & R & R \\ * & D & R & D & * \\ D & * & R & * & D \end{bmatrix}$$

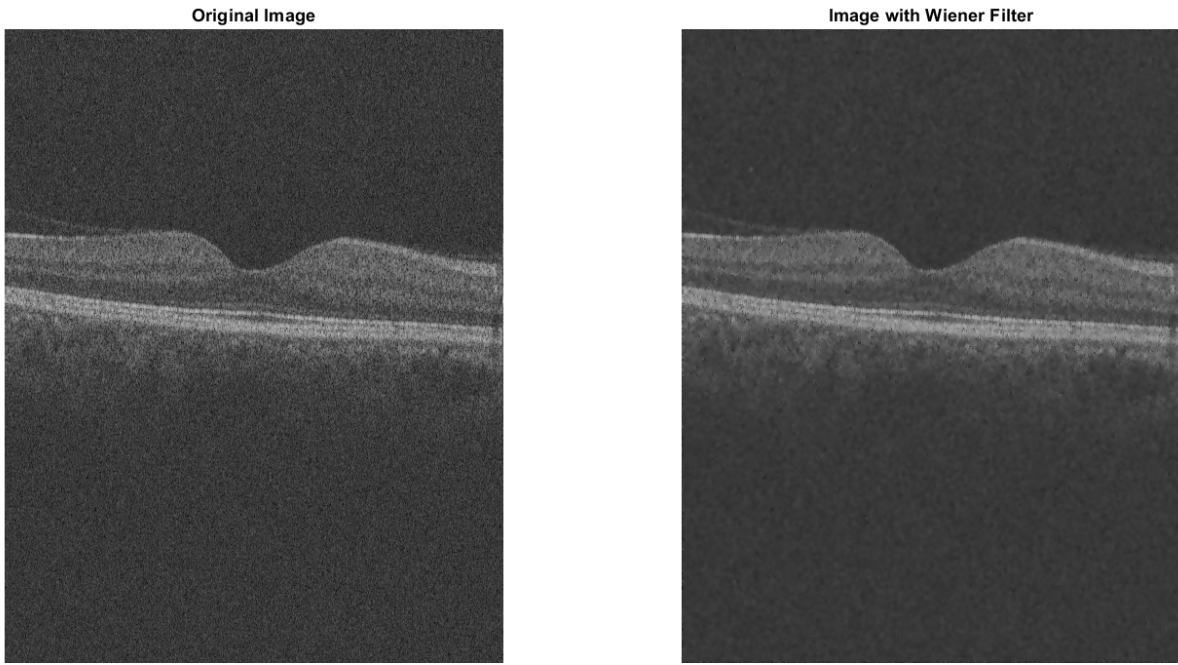
The Matlab implementation by Damian Garcia was chosen for our analysis of this filter method on OCT images[7]. The output image of this filter can be seen in Figure 9.

## VI. CONCLUSION

From the data gathered during the Matlab simulations, it is clear to see that image processing techniques greatly reduce noise and still have the ability to preserve edges. The calculated performance metrics used in Matlab confirm this. Between the 5 different filters used the Wiener Filter performed the best in having a high Contrast to Noise Ratio (CNR) and Naturalness Image Quality Score (NIQE). This filter also produced a significantly lower Estimated Noise Level (ENL) as well. Refer to Table I for the Matlab simulation quality metrics results. Although the Wiener filter performed the strongest using these quality metrics, the Hybrid Median filter and Lee filter would be more optimal for reducing speckle noise in OCT imagery. Both these filters have better edge preserving algorithms and still have quality CNR and NIQE scores.

Filter Used	CNR(dB)	ENL	NIQE
No Filter	9.0375	0.0247	10.823
Wiener	9.213	0.0077	5.95
Lee	8.316	0.0015	6.489
Kuwahara	8.878	0.0111	8.82
Symmetric Neighbor	8.788	0.0124	14.997
Hybrid Median	6.98	0.0069	7.817

TABLE I: Comparison of Filter Quality Metrics

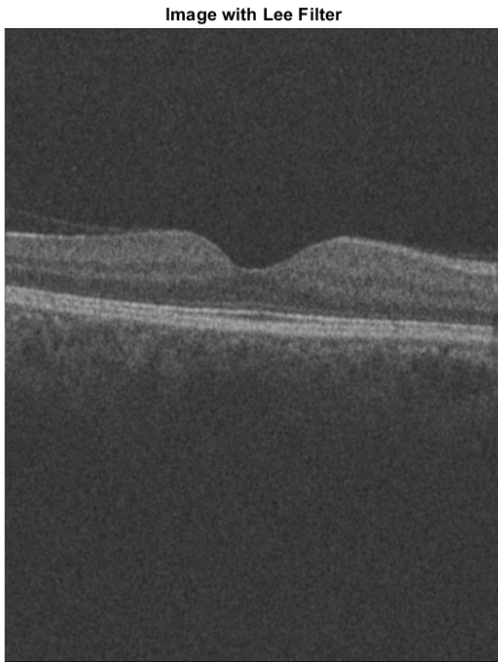


(a) Original Unfiltered Image

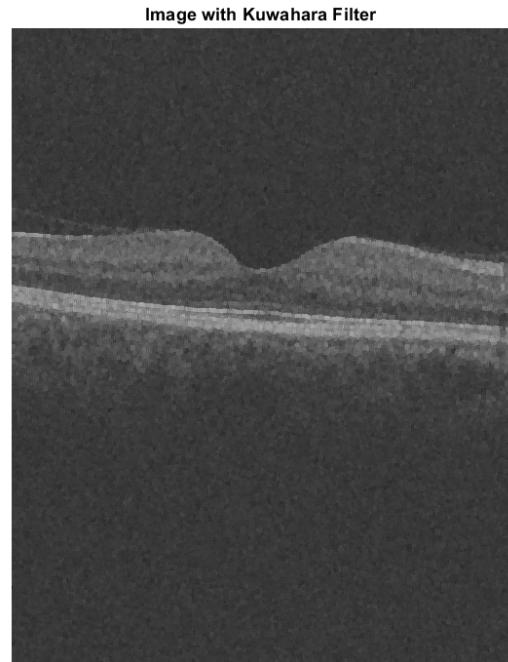
(b) Wiener Filtered Image

FIG. 7: Unfiltered and Wiener Filtered Images



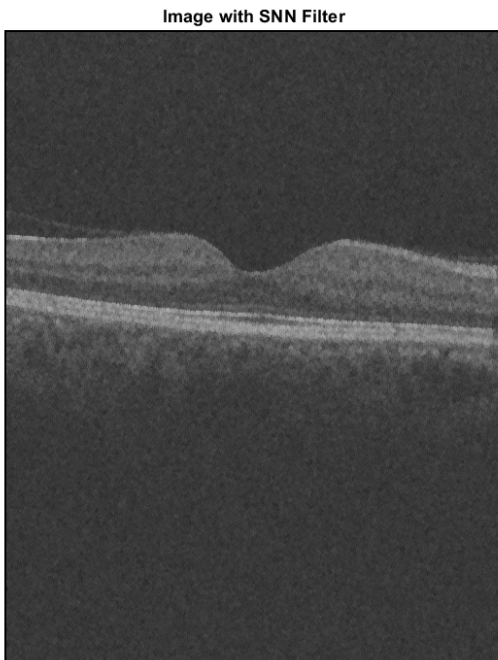


(a) Lee Filtered Image

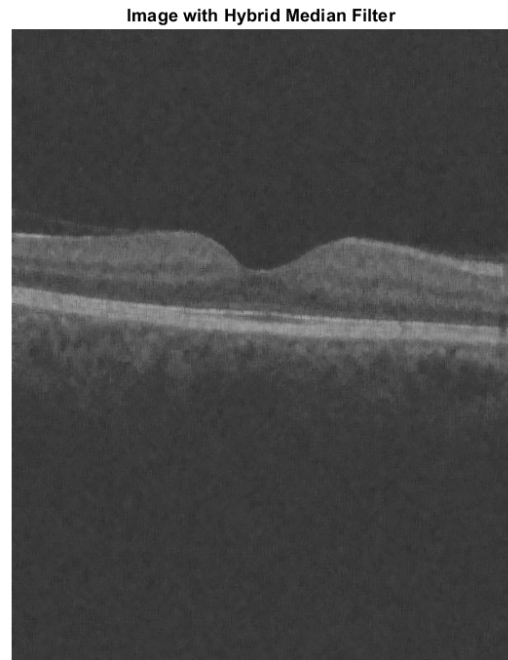


(b) Kuwahara Filtered Image

FIG. 8: Lee and Kuwahara Filtered Images



(a) SNN Filtered Image



(b) Hybrid Filtered Image

FIG. 9: SNN and Hybrid Filtered Images

```

1 %Script to Open and Slice an OCT Image, Apply Filtering , and Print Quality Metrics
2
3 clear all;
4 close all;
5
6 %Load and Slice 2D Image
7 load('1_left.mat');
8 img = mat2gray(d3(:,:,64), [0 255]);
9
10 %Original Slice
11 subplot(2,3,1), imshow(img)
12 %figure, imshow(img)
13 title('Original Image')
14
15 %hybrid Median Filter
16 M=hmf(img,9);
17 subplot(2,3,3), imshow(M)
18 %figure, imshow(M)
19 title('Image with Hybrid Median Filter')
20 [CNR,ENL,NIQEScore] = EvaluateFilter(M);
21 fprintf('\n CNR score HMF is %0.4f\n',CNR)
22 fprintf('\n ENL score HMF is %0.4f\n',ENL)
23 fprintf('\n NIQE score HMF is %0.4f\n',NIQEScore)
24
25 %Lee Filter
26 [le]=lee1((d3(:,:,64)),1);
27 subplot(2,3,4), imshow([le], [0 255])
28 %figure, imshow([le], [0 255])
29 title('Image with Lee Filter')
30 [CNR,ENL,NIQEScore] = EvaluateFilter([le]);
31 fprintf('\n CNR score Lee is %0.4f\n',CNR)
32 fprintf('\n ENL score Lee is %0.4f\n',ENL)
33 fprintf('\n NIQE score Lee is %0.4f\n',NIQEScore)
34
35 %Symmetric Nearest Neighbor Filter
36 H = snn(img,5,true);
37 subplot(2,3,5), imshow(H)
38 %figure, imshow(H)
39 title('Image with SNN Filter')
40 [CNR,ENL,NIQEScore] = EvaluateFilter(H);
41 fprintf('\n CNR score SNN is %0.4f\n',CNR)
42 fprintf('\n ENL score SNN is %0.4f\n',ENL)
43 fprintf('\n NIQE score SNN is %0.4f\n',NIQEScore)
44
45 %Kuwahara Filter
46 filtered =Kuwahara(img,5);
47 subplot(2,3,6), imshow(filtered)
48 %figure, imshow(filtered)
49 title('Image with Kuwahara Filter')
50 [CNR,ENL,NIQEScore] = EvaluateFilter(filtered);
51 fprintf('\n CNR score Kuwahara is %0.4f\n',CNR)
52 fprintf('\n ENL score Kuwahara is %0.4f\n',ENL)
53 fprintf('\n NIQE score Kuwahara is %0.4f\n',NIQEScore)
54
55 %Wiener Filter
56 K = wiener2(img,[5 5]);
57 subplot(2,3,2), imshow(K)
58 %figure, imshow(K)
59 title('Image with Wiener Filter')
60 [CNR,ENL,NIQEScore] = EvaluateFilter(K);
61 fprintf('\n CNR score Wiener is %0.4f\n',CNR)
62 fprintf('\n ENL score Wiener is %0.4f\n',ENL)
63 fprintf('\n NIQE score Wiener is %0.4f\n',NIQEScore)

```

Listing 3: Script to Generate Figure of Filtered Images and Print Quality Metrics



- 
- [1] Radar Enhanced Lee Filter. [http://www.pcigeomatics.com/geomatica-help/concepts/orthoengine\\_c/Chapter\\_825.html](http://www.pcigeomatics.com/geomatica-help/concepts/orthoengine_c/Chapter_825.html). [Online; accessed 18-April-2008].
- [2] Balbi, L. Faster Kuwahara filter - File Exchange - MATLAB Central. <http://www.mathworks.com/matlabcentral/fileexchange/15027-faster-kuwahara-filter>. [Online; accessed 18-April-2008].
- [3] Barnes, A. (2005). SNN filter - File Exchange - MATLAB Central. <http://www.mathworks.com/matlabcentral/fileexchange/8174-snn-filter>.
- [4] Bhende, M., Shetty, S., Parthasarathy, M. K., and Ramya, S. (2018). Optical coherence tomography: A guide to interpretation of common macular diseases. *Indian Journal of Ophthalmology*, 66(1):20–35.
- [5] DinoVgk. The Kuwahara Operator. <https://commons.wikimedia.org/wiki/File:Kuwahara.jpg>. [Online; accessed 18-April-2008].
- [6] fiveko (2017). Symmetric Nearest Neighbor Filter. <http://fiveko.com/tutorials/image-processing/symmetric-nearest-neighbor-filter/>. [Online; accessed 18-April-2008].
- [7] Garcia, D. (2010). Hybrid median filtering - File Exchange - MATLAB Central. <http://www.mathworks.com/matlabcentral/fileexchange/25825-hybrid-median-filtering>. [Online; accessed 18-April-2008].
- [8] Huang, D., Swanson, E. A., Lin, C. P., Schuman, J. S., Stinson, W. G., Chang, W., Hee, M. R., Flotte, T., Gregory, K., Puliafito, C. A., and Et, A. (1991). Optical coherence tomography. *Science*, 254(5035):1178–1181.
- [9] Liu, X., Tanaka, M., and Okutomi, M. (2013). Single-image noise level estimation for blind denoising. *IEEE Transactions on Image Processing*, 22(12):5226–5237.
- [10] Mastriani, M. and Giraldez, A. E. Enhanced directional smoothing algorithm for edge-preserving smoothing of synthetic-aperture radar images. <http://desktop.arcgis.com/en/arcmap/10.3/manage-data/raster-and-images/speckle-function.htm#GUID-F45976F8-09C5-4344-B249-FBAA03517B8A>. [Online; accessed 18-April-2008].
- [11] Meshram, A. (2014). Noise Level Estimation. [Online; accessed 18-April-2008].
- [12] Mittal, A., Soundararajan, R., and Bovik, A. C. (2013). Making a completely blind image quality analyzer. *IEEE Signal Processing Letters*, 20(3):209–212.
- [13] Oppenheim, A. V. and C., V. G. (2015). *Optical Coherence Tomography: Principles and Applications*. Pearson.
- [14] Ozcan, A., Bilenca, A., Desjardins, A. E., Bouma, B. E., and Tearney, G. J. (2007). Speckle reduction in optical coherence tomography images using digital filtering. *Journal of the Optical Society of America. A, Optics, image science, and vision*, 24(7):1901–1910.
- [15] Papari, G., Petkov, N., and Campisi, P. (2007). Artistic edge and corner enhancing smoothing. *IEEE Transactions on Image Processing*, 16(10):2449–2462.
- [16] Popescu, D. P., Hewko, M. D., and Sowa, M. G. (2006). Speckle noise attenuation in optical coherence tomography by compounding images acquired at different positions of the sample. *Optics Communications*, 269.
- [17] Rabie, T. (2004). Adaptive hybrid mean and median filtering of high-ISO long-exposure sensor noise for digital photography. *J. Electronic Imaging*, 13:264–277.
- [18] Sander, B. and Li, X. Q. (2011). Optical Coherence Tomography: A review of current technology and its implications for clinical applications. *Scandinavian Journal of Optometry and Visual Science*, 4(2):1–4.
- [19] Stough, J. 3d Volume Visualization - File Exchange - MATLAB Central. <http://www.mathworks.com/matlabcentral/fileexchange/37268-3d-volume-visualization>. [Online; accessed 18-April-2008].
- [20] Tahereh Mahmudi, Rahele Kafieh, Hossein Rabbani, Alireza Mehri dehnavi, and Mohammadreza Akhlagi (2014). Comparison of macular OCTs in right and left eyes of normal people. volume 9038, pages 90381W–9038–6.