Blood capillaries and vessels segmentation in optical coherence tomography angiogram using fuzzy C-means and Curvelet transform

Fariborz Taherkhani

Department of Electrical Engineering and Computer Science, University of Wisconsin-Milwaukee, Milwaukee, WI, USA

1. Abstract

Blood capillaries density and vessels structure in vivo brain provide information about diseases and hemodynamic in the brain network. Capillary density in animal brain is one of the indexes that determines if it has diabetes. Capillary density in the animal brain subjected to diabetes diminishes over time. Variation in physical structure and diameter of the vessels as a function of time gives us information about hemodynamic in the brain network as well. Capillaries and vessels segmentation is a process that can help us in predicting some of diseases such as diabetes as well as monitoring certain area of the brain that are supposed to be stimulated by Optogenetics technique for controlling blood flow in the brain network. In this work, we propose an unsupervised learning approach to distinguish capillaries from vessels. The proposed method is based on curvelet transform which measures thickness of curve singularities in Optical Coherence Tomography (OCT) images. In this work, first we enhance contrast of image by strengthening faint capillaries and softening vessels; we suppress noise level of image in this step as well. This step of the algorithm is performed by modifying curvelet coefficients using a non-linear function. Second, we extract a feature vector from each pixel of image using curvelet transform and finally, we apply a well-tuned fuzzy C-mean clustering approach to segment capillaries and vessels from background. Improvement in segmentation accuracy is one of the main motivations of this work; experimental results on test images indicate that the proposed method outperforms some of existing segmentation methods.

2. Introduction

Optical Coherence Tomography (OCT) is an optimal signal acquisition, interferometric and noninvasive method that captures 3-D surface of biological tissue microstructure with micron-scale resolution. OCT uses near infrared light to generate images with sub micrometer resolution [1]. In OCT, same as ultrasound imaging, the distance between optical reflections from different layers whiten tissue is measured by recording the time delay between reflection arrivals from the tissue layers at an optimal detector [2]. Both these imaging techniques capture structure and hemodynamic information [3, 4] . OCT resolution is higher than ultrasound imaging while penetration and imaging depth are bounded to 2mm which influences on their applications [5]. OCT imaging technique is highly used in clinical trials such as lung cancer, acute coronary syndromes, gastrointestinal tract and age related muscular degeneration. In all these applications,

we need to develop an automatic segmentation algorithm to obtain geometrical and physical parameters of blood vessels and capillaries density in OCT images.

In terms of machine learning, segmentation is an unsupervised learning problem that clusters objects from background. In this process, each pixel of an image is clustered into one of distinct groups. Blood vessels and capillaries segmentation particularly retinal vessels segmentation, has been a challenging problem in biomedical image processing field [6-8]. There are many approaches that address this problem. Method [9] proposes an unsupervised learning approach; this method after normalizing the intensity of vessels in image, applies three specific filters including matched filter [10] , Frangi's filter [11] , Gabor and Wavelet filter [12] to enhance blood vessels in the image. Finally, this method uses Fuzzy C-means, ORSF and optimize threshold approaches to segment vessels in the image. Method [13] proposes a wavelet based approach to calculate diameters of vessels along the length of each vessel; segmentation in this method is performed by adding the wavelet levels showing the best contrast for vessels and thresholding based on a percentage of highest or lowest valued coefficients. Method [14] uses gray-voting approach to enhance small vessels; and a two-dimensional Gabor wavelet is applied to extract the main vessels simultaneously. And finally, a Gaussian mixture model is used to extract vessel cluster, while small vessels are detected by another gray-voting approach. Method [15] proposes a multi-scale retinal vessel segmentation that uses a novel seeded multi-scale line-tracking procedure and morphological post-processing to extract vessels in the image. Method [16] extract vascular structures of retinal in the image. This method uses morphological preprocessing to emphasize linear structures such as vessels and then a derivative operator is used to emphasize thin vascular structures and finally a thresholding is used to prepare a segmented vascular mask. Thickness of the vessels vs capillaries is one of the distinctive properties for segmenting these two clusters. In this paper, we use Multi Resolution Analysis techniques[17, 18] to measure thickness of capillaries and vessels to segment them from each other.

Multi Resolution Analysis (MRA) are profoundly related to image processing particularly in image segmentation, image compression, image de-noising as well as computer vision. Wavelet [19], Ridgelet [20] and Curvelet transforms [21, 22] have been used as higher dimensional MRA techniques to represent images sparsely. This representation is obtained through invertible and non-redundancy transforms. Wavelet carries out MRA of a given signal by localizing wavelet functions in both spatial and frequency domains; in the other words, this transform encodes a given signal by computing correlation coefficients between given signal and a complete set of orthogonal wavelet functions constructed by translating and dilating of a single-mother wavelet function localized in both spatial and frequency domain. Furthermore, Wavelet outperforms sine-cosine transforms to reconstruct functions with point singularities by using fewer number of non-zero coefficients for a given reconstruction accuracy. Mostly, wavelet is more suitable to reconstruct point or zero dimensional singularities rather than edges or lines singularities in images; because this transform cannot insulate smoothness along edges which exists in images.

To address this problem, in 1998 Candes proposed continuous ridgelet transform (CRT) to extend wavelet transform to higher dimensional singularities as an efficient technique for directional sparse representation [23]. Ridgelet transform controls directions of line singularities existing in images by improving the idea of point-to-point mapping to point-to-line mapping singularities. Ridgelet transform can be considered as an approach that concatenates one dimensional wavelets along lines with radial directions. Ridgelet outperforms wavelet in line singularities representation because singularities are usually concatenated together along edges or contours in images. In two dimensional signals such as images, points and lines are related by Radon transform; therefore, wavelet and ridgelet transforms are related by each other using Radon transform. Ridgelet transform is optimal for straight line singularities representation; however, global straight line singularities are not seen in practical applications; therefore, this transform is not appropriate for curve singularities representation. Candes and Donoho in 2000 introduced curvelet transforms as an effective MRA technique to address the disadvantages of wavelet and ridgelet analyses[22, 24]. The curvelet transform is designed as a multiscale directional transform to represent objects consist of edge and other singularities along curves much more effectively than other traditional transforms with the advantage of using almost optimal non-adaptive sparse coefficients in the transform domain. In Curvelet transform, each sub band is obtained by dividing frequency domain in a way that higher frequency is corresponded to larger band- width and each of these sub bands is divided into several regions with different polar angles. Curvelet transform encodes curve singularity and any discontinuity in the image more sparsely with very fewer numbers of non-zero coefficients to obtain a certain reconstruction accuracy for a given two-dimensional curved signals. Curvelet captures curve singularities more meticulously because all basic functions in curvelet dictionary is determined by orientation, position and scale to be localized completely on curved signals in two-dimensional Cartesian space. All basic functions in curvelet dictionary constructs a pyramid of functions which have intensive degree of directional and parabolic or anisotropic scaling (i.e. width and length of basic functions are related to each other by length=width^2 relation) properties[25-27] . In this paper, we propose a new MRA based approach to segment blood capillaries and vessels in the brain images automatically, the proposed method is a curvelet based approach that represents image pixels in the feature space; Fuzzy C-Means [28] as a clustering method is applied on the feature vectors to cluster pixels into the different clusters.

2.2 Continues Curvelet Transform

Candes introduced curvelet transform as an optimal sparse representation of objects with curve singularities occurring in the image [27]. This transform represents curve by covering a series of curvelet functions with various lengths and widths following parabolic or anisotropic scaling law. Curvelet has two generations; first generation works based on decomposing signal into some sub bands and applying ridgelet transform on each of sub bands [22]. Because first generation worked very slowly, second generation of curvelet transform has been introduced. This

generation reduced processing redundancy by disregarding preprocessing step in the ridgelet transform and this operation ended to increasing the speed of curvelet transform significantly [25, 26].

To construct curvelet functions, we need to design a basic set of windows including radial $W(r)$ and angular $V(t)$ windows used to construct other curvelet functions using three parameters including scale, translation and rotation.

$$V(t) = \begin{cases} 1 & |t| \le 1/3 \\ \cos\left[\frac{\pi}{2}\,\alpha(3|t|-1)\right] & 1/3 \le |t| \le 2/3 \\ 0 & else \end{cases} \quad (1)$$

$$W(r) = \begin{cases} \cos\left[\frac{\pi}{2}\,\alpha(5-6r)\right] & 2/3 \le |r| \le 5/6 \\ 1 & 5/6 \le |r| \le 4/3 \\ \cos\left[\frac{\pi}{2}\,\alpha(3r-4)\right] & 4/3 \le |r| \le 5/3 \\ 0 & else \end{cases} \quad (2)$$

$$\alpha(x) = \begin{cases} 0 & x \le 0 \\ \frac{S(x-1)}{S(x-1)+S(x)} & 0 < x < 1 \\ 1 & x \ge 1 \end{cases} \quad (3)$$

$$S(x) = \exp(-[\frac{1}{(1-x)^2} + \frac{1}{(1+x)^2}]) \quad (4)$$

These two windows same as scaled Meyer window should satisfy admissibility conditions same as follows:

$$\sum_{l=-\infty}^{\infty} V^2(t-l) = 1; \qquad t \in \mathbb{R} \quad (5)$$

$$\sum_{j=-\infty}^{\infty} W^2(2^j r) = 1; \qquad r > 0 \quad (6)$$

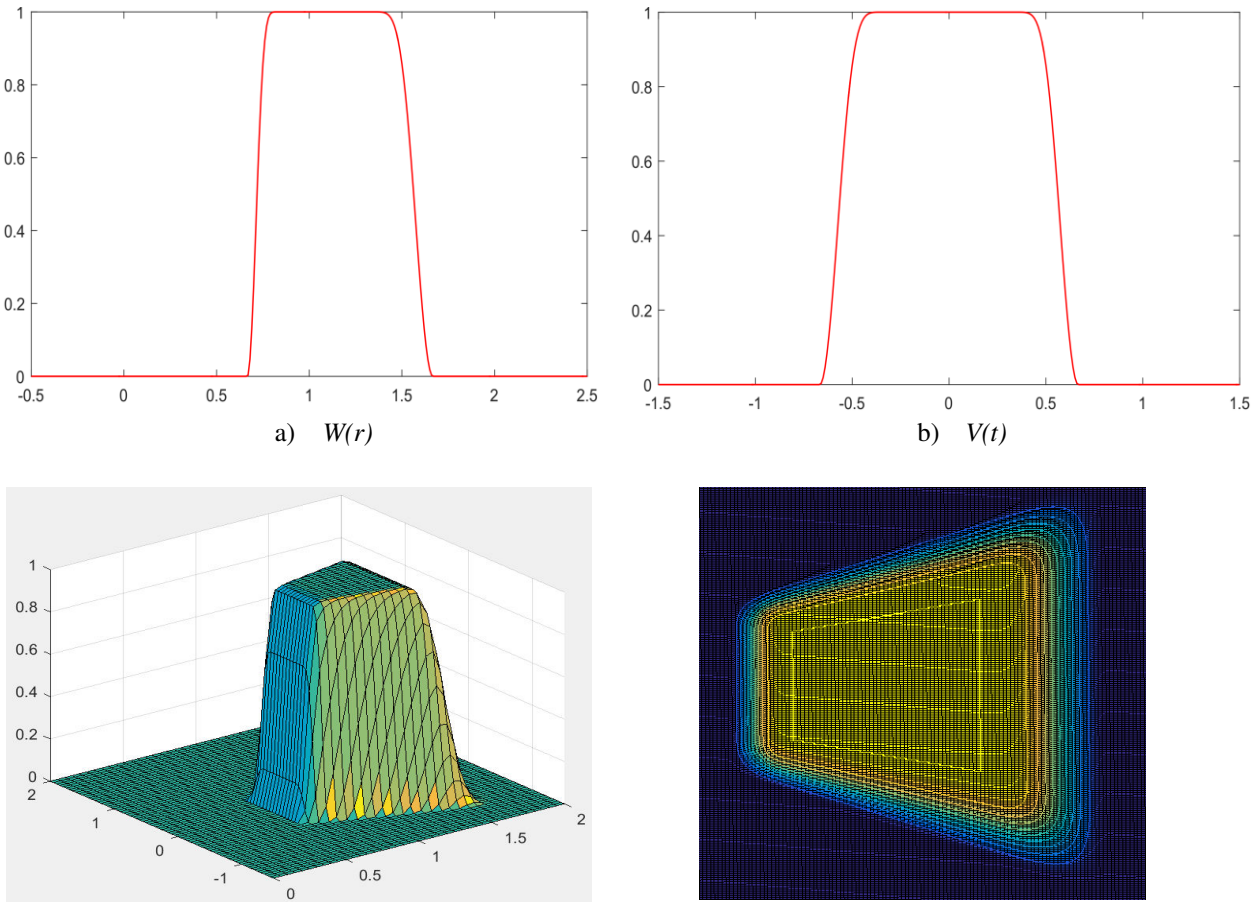


Fig.1: a) *w(r)*, b) *v(t)* the basis window functions; c) and d) is basis curvelet function and its support in frequency domain respectively

All other curvelet functions is constructed by these two window functions. Fourier transform of basic curvelet function is obtained by multiplying *W(r)* and *V(t)* window functions. Assume that $x = [x_1, x_2]^T$ and $\omega = [\omega_1, \omega_2]^T$ are samples in spatial and frequency domains respectively; and the polar coordinate of ω is represented by $r = \sqrt{\omega_1^2 + \omega_2^2}$ and $\theta = \arctan(\frac{\omega_1}{\omega_2})$. Fourier transform of basic curvelet function is obtained same as follows:

$$\Psi_a(r;\theta) = a^{(3/4)} W(ar) V(\frac{\theta}{\sqrt{(a)}}) \quad (7)$$

where $a \in (0; 1]$ is a scaling parameter that scale curvelet function and usually is set to 1/2. Assume that $\chi_{(a,0,0)}(x)$ is the basic curvelet function in spatial domain $\mathbb{R}^2$; and $\tilde{\chi}_{(a,0,0)}(\omega)$ is its Fourier transform in the frequency domain. We determine basic curvelet function in frequency domain using $\Psi_a(r;\omega)$ in Equation (7); so, the $\tilde{\chi}_{(a,0,0)}(\omega)$ is defined same as follows:

$$\tilde{\chi}_{(a,0,0)}(\omega) = \Psi_a(\omega) \quad (8)$$

all other curvelet functions is constructed by scaling, translation and rotation of basic function $\chi_{(a,0,0)}(x)$ same as follows:

$$\chi_{(a,b,\theta)}(x) = \chi_{(a,0,0)}(R_\theta(x-b)) \tag{9}$$

where b is the translation vector that translates curvelet function $b \in \mathbb{R}^2$ and θ is rotation angel that rotates curvelet functions $\theta \in [0, 2\pi)$ in the spatial domain. Rotation matrix is defined same as follows:

$$R_\theta = \begin{pmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{pmatrix} \tag{10}$$

Fourier transform of $\chi_{(a,b,\theta)}(x)$ is obtained same as follows:

$$\tilde{\chi}_{(a,b,\theta)}(\omega) = \mathrm{e}^{-i\langle b,\omega\rangle}\tilde{\chi}_{(a,0,0)}(R_\theta\omega) \tag{11}$$

using Equation (4), we can write Equation (11) same as follows:

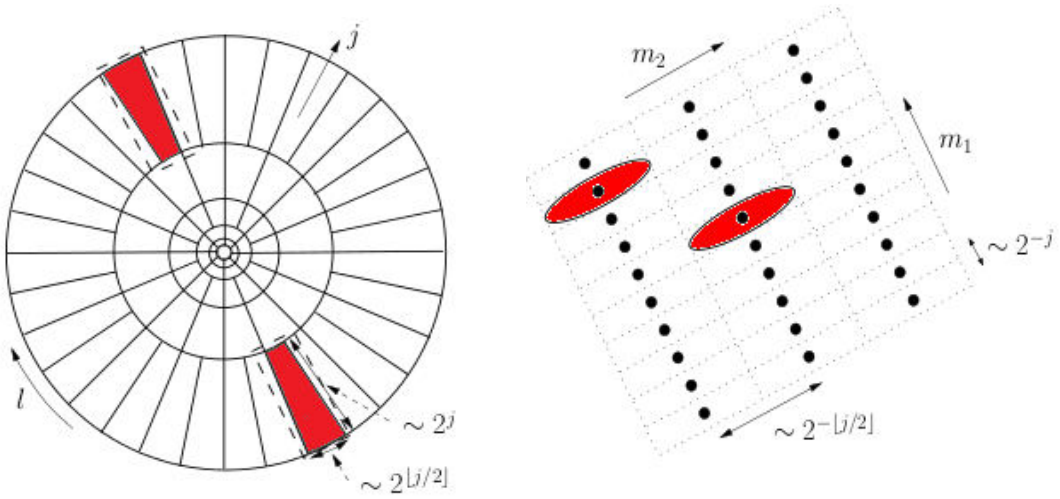
$$\tilde{\chi}_{(a,b,\theta)}(\omega) = \mathrm{e}^{-i\langle b,\omega\rangle}\Psi_a(R_\theta\omega) \tag{12}$$

continuous curvelet transform of function $f(x)$ in $\mathbb{R}^2$ is obtained by (13) where $C_f(a,b,\theta)$ is curvelet coefficient which indicates correlation between $f(x)$ and curvelet function $\chi_{(a,b,\theta)}$.

$$C_f(a,b,\theta) = \langle \chi_{(a,b,\theta)}, f\rangle = \int_{\mathbb{R}^2} \chi_{(a,b,\theta)}(x) f(x)dx \tag{13}$$

since curvelet transform is invertible, we can reconstruct $f(x)$ using (14)

$$f(x) = \sum_{a,b,\theta} C_f(a,b,\theta).\chi_{(a,b,\theta)} \tag{14}$$



a) Frequency domain b) Spatial domain

Fig.2: Continuous curvelet transform tiling in frequency and space domains

a) Continuous curvelet transform frequency domain tiling; b) Continuous curvelet function in spatial domain corresponding to given scale and orientation in (a).

Discrete Curvelet Transform

In real application, we need to discretize continuous curvelet transform; the most recent and enhanced version of discrete curvelet transform is known as Fast Discrete Curvelet Transform (FDCT)[29]. FDCT is easier to use, faster and less redundant than ridglet-based curvelet transform. There exists two ways of FDCT implementations described as follows [29]:

(1) Unequally spaced Fast Fourier transforms (USFFT)

(2) Wrapping

The difference between two implementation methods is the choosing of spatial grid of curvelet function used to translate it at each scale and angle. Both of implementation methods return same results. Each coefficient represents correlation between input signal and a curvelet function identified by one scale, one rotation angel and translation vector. Since wrapping method is faster and easier than USFFT, we use this method in our experiment. One of the concerns associated to wrapping implementation is that trapezoidal wedge is not fitted into a $2^j \times 2^{j/2}$ rectangle along the axes in the frequency plane in a way that 2D IFFT can be applied to calculate curvelet coefficients. To address this issue, wedge wrapping algorithm proposed in [29] uses a $2^j \times 2^{j/2}$ parallelogram to cover wedge data. The wrapping procedure tills spectrum inside the wedge periodically and then collects coefficients in the central rectangular. Fig.3 (b) shows wrapping procedure; in wrapping based curvelet transform, each coefficient C (j, l, k_1, k_2) indexed by spatial translation vector $[k_1, k_2]^T$, one orientation l and one scale j represents

convolution between input signal and basis curvelet function scaled by factor of 2^j, rotated by angel $\theta_l = 2\pi . 2^{-\left\lfloor \frac{j}{2} \right\rfloor} . l$ and translated by (k_1, k_2) in spatial domain. Wrapping algorithm is described same as follows:

1) Take 2-D FFT from input image $f(x, y)$ and obtain Fourier coefficients

$$\tilde{f}(u,v) = \sum_{x,y=0}^{N-1} f(x,y) e^{(-2\pi i(xu+yv)/N)} \qquad \frac{-N}{2} \le u, v \le \frac{N}{2} \tag{15}$$

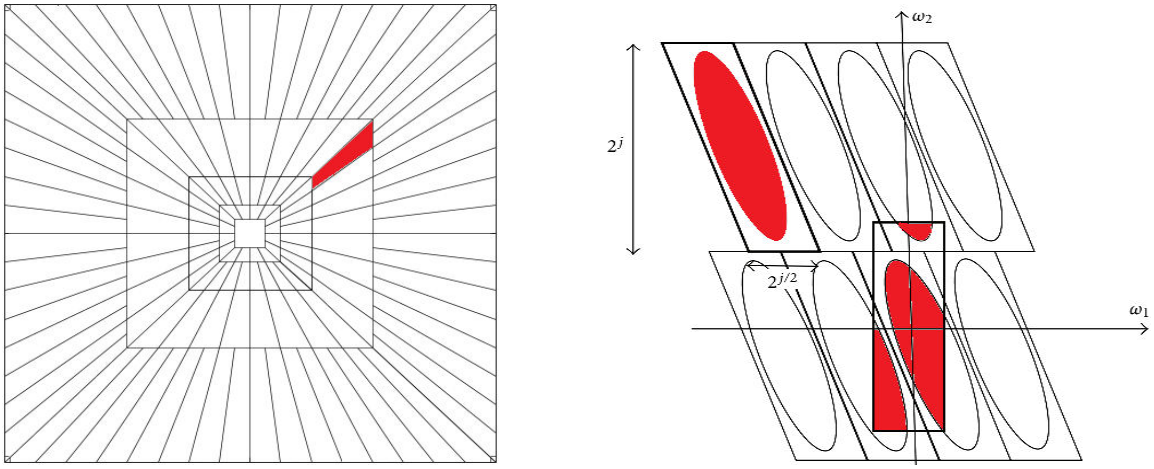
2) For each angle θ_l and scale j, construct the product of discretized localizing curvelet function and Fourier samples $\tilde{f}(u,v)\ \tilde{\chi}_{(j,\theta_l)}(u,v)$;

3) Wrap the results in (step 2) around the center and obtain

$$\tilde{F}_{(j,\theta_l)}(u,v) = Wrap(\tilde{f}(u,v)\ \tilde{\chi}_{(j,\theta)}(u,v)) \qquad 0 \le u < 2^j \text{ and } 0 \le v < 2^{j/2}\ ; \frac{-\pi}{4} \le \theta < \frac{-\pi}{4} \tag{16}$$

4) Perform inverse 2-D FFT to each $\tilde{F}_{(j,\theta_l)}(u,v)$ and obtain collected discrete curvelet coefficients $C(j, l, K)$; $K = (k_1, k_2)$; $U = 2^j; V = 2^{j/2}$

$$C(j,l,K) = \frac{1}{N^2} \sum_{u=0}^{U-1} \sum_{v=0}^{V-1} \tilde{F}_{(j,\theta_l)}(u,v)\ e^{(2\pi i(k_1 u/U + k_2 v/V))} \tag{17}$$



a) Discrete curvelet transform frequency domain tiling

b) wrapping tilling procedure

Fig. 3: Discrete curvelet transform tilling and wrapping tilling procedure

3.1 Fuzzy C-Means (FCM) clustering

FCM clustering algorithm is an unsupervised learning approach which clusters data, in our case pixels of image into several distinct groups including capillaries, vessels and background based on similarity between pixels of each group. Proposed method represents each pixel by a feature vector using curvelet transform. For each pixel in image, FCM assigns a membership value to each cluster based on the similarity between pixel and cluster prototype. The more similarity between pixel and cluster prototype, the bigger membership value is assigned to that cluster. One of the main reasons that we use fuzzy clustering method to cluster vessels and capillaries is that thickness of curve singularity in image which determines the curve as a capillary or vessel is a fuzzy concept [30]. Curvelet transform in this method is used as a measure of thickness of curve singularities. Curvelet transform can measure thickness of curve singularity by computing coefficients of fittest curvelet functions with different scales. Fig. 4 shows that regardless of curvelet orientation fitted on the curve, thinner curves output bigger coefficient values with large scaled curvelets; while thicker curves output bigger coefficient values with small scaled curvelets. In the other word, thicker curve singularities have more correlation with small scaled curvelet functions and thinner curve singularities have more correlation with large scaled curvelet functions. FCM obtains membership values using minimizing a nonlinear objective function defined in (18) subjected to one constrain condition over membership parameters.

Optimal parameters that minimize fuzzy c-means objective function with equality constrain presented in (18) can be obtained by Lagrangians multipliers and iterative minimization algorithms [31, 32].

Assume that we have a set of n pixels $X = \{x_1, x_2, x_3, \ldots x_n\}$ each of which is represented by a feature vector x_i; we aim to cluster them into m groups. So, in our case m=3 which groups are capillaries, vessels and background.

The objective function to cluster samples $X = \{x_1, x_2, x_3, \ldots x_n\}$ is defined same as follows:

$$min_{\theta,U} \; J_q(\theta, U) = \sum_{i=1}^{n} \sum_{j=1}^{m} u_{ij}^{q} \, d^2(x_i, \theta_j) \qquad q>1$$
$$\text{subject to } \sum_{j=1}^{m} u_{ij} = 1 \qquad \forall x_i \tag{18}$$

where q determines fuzziness or smoothness of memberships in the formula. we set up this parameter to 2 for simplicity in implementation; u_{ij} indicates membership value of sample x_i belonging to cluster j; $\theta = \{\theta_1, \theta_2, \ldots, \theta_m\}$ is prototype of the clusters and $d^2(x_i, \theta_j) = (x_i - \theta_j)^T (x_i - \theta_j)$ is Euclidian distance which represents similarity between x_i and θ_j . For each input pixel, we assign a membership value to each cluster using pro-prototype of the cluster. The stationary points of the objective function in (18) are determined by adding constraint defined in (18) to J_q using Lagrange multipliers as follows:

$$J = \sum_{i=1}^{n}\sum_{j=1}^{m} u_{ij}^{q}\, d^2(x_i, \theta_j) - \sum_{i=1}^{n} \lambda_i \,(\sum_{j=1}^{m} u_{ij} - 1) \tag{19}$$

To minimize the function *J*, derivatives of *J* respect to u_{rs} in (20) and θ_j in (25) should be set to zero;

$$\frac{\partial J}{\partial u_{rs}} = 0\ ; \rightarrow q u_{rs}^{(q-1)} d^2(x_r, \theta_s) - \lambda_r = 0\ ; \tag{20}$$

u_{rs} is obtained by (20) same as follows:

$$u_{rs} = [\frac{\lambda_r}{q d^2(x_r, \theta_s)}]^{(1/(q-1))}; \tag{21}$$

u_{rs} in (21) is determined by knowing λ_r; To find λ_r, we use constrain condition in (18) same as follows:

$$\sum_{j=1}^{m} (\frac{\lambda_r}{q d^2(x_r, \theta_j)})^{(1/(q-1))} = 1; \tag{22}$$

λ_r is obtained by (22) same as follows:

$$\lambda_r = \frac{q}{[\sum_{j=1}^{m} (\frac{1}{d^2(x_r, \theta_j)})^{(1/(q-1))}]^{(q-1)}}\ ; \tag{23}$$

By plugging λ_r obtained by (23) into (21); u_{rs} is determined same as follows:

$$u_{rs} = \frac{1}{\sum_{j=1}^{m} (\frac{d^2(x_r, \theta_s)}{d^2(x_r, \theta_j)})^{(1/(q-1))}} \tag{24}$$

Once u_{rs} is determined by (24); we can update prototype of each cluster θ_j using (25)

$$\frac{\partial J}{\partial \theta_j} = 0; \rightarrow \sum_{i=1}^{n} u_{ij}^{q} \frac{\partial\, d^2(x_i, \theta_j)}{\partial \theta_j} = 0\ ;\ \ j = 1, .., m; \tag{25}$$

Proposed Algorithm

In this section, we describe our proposed method to segment blood capillaries and vessels in OCT-angiograms captured from brain tissues. Proposed algorithm consists of three steps including contrast and edge enhancement, representing image pixels in feature space and finally utilizing a well-tuned fuzzy C-means clustering method to segment pixels to three groups including capillary, vessel and background.

Step 1: Image contrast and edge enhancement using Curvelet transform

Since curvelet transform is an appropriate multi resolution technique to represent images containing curve singularities such as brain tissue images, it is a proper chose to enhance contrast and edge details accruing in image. During image accusation from brain tissues, OCT miss resolution in the deeper layers; therefore, lots of information related to capillaries is lost and image gets blurry; while, deeper layers provides more information about capillaries density and homonymic information in brain vessel. To recover this information, we need to enhance contrast as well as edges resolution in the image. Typically, as it is shown in Fig.4; curvelet functions at high frequency scales are fine and needle shape; while they are coarse and non-directional at low frequency scales.

To preserve curve details accruing in image and enhance image contrast using curvelet transform; proposed method modifies curvelet coefficients using a non-linear function in a way that faintest curves (i.e. capillaries) are intensified and strong curves (i.e. vessels) are softened simultaneously. This non-linear function should not amplify noise coefficients in curvelet domain. Power spectrum of the image in curvelet domain is distributed in coefficients of noise, capillaries and vessels in an increasing order; it means that for each sub band of image in frequency domain (i.e. one specific scale and rotation), curvelet coefficients of the image is classified to three intervals. First interval is coefficients of the noise existing in the image which are smaller than coefficients of capillaries and vessels; second interval is coefficients of the capillaries which are smaller than coefficients of vessels and bigger than coefficients of noise. Finally, third interval is coefficients of the vessels which are bigger than noise and capillaries coefficients. Proposed method suppresses and weakens coefficients of the first interval to reduce noise existing in the image; it amplifies second interval to intensify capillaries existing in the image; and finally, this method weakens third interval to soften vessels existing in the image. This step of the algorithm ends up to noise suppression as well as enhancing contrast of the image. To avoid amplifying noise coefficients, we need to estimate explicitly standard deviation of the noise to regulate the coefficients modification interval. We define proposed non-linear function same as follows:

$$f(x) = \begin{cases} k_1 & |x| < \alpha c \\ k_2 \left(\frac{m}{|x|}\right)^p & \alpha c \le |x| < m \\ k_3 \left(\frac{m}{|x|}\right)^s & |x| \geqslant m \end{cases} \tag{26}$$

where *x* expresses curvelet coefficient which is going to be modified by function *f(x);* parameters *p* and *s* determine degree of the nonlinearity of the function. These two parameters should be set to a non-zero value to enhance faintest curves (i.e. capillaries) and soften strongest curves (i.e. vessels). Term α is noise standard deviation and c is a normalization parameter that prevents coefficients of noise to be amplified; m is data dependent parameter determined based on coefficients existing in each sub band (i.e. one specific scale and rotation). Curvelet coefficients

lower than *m* is amplified and curvelet coefficients higher than *m* is condensed. Therefore, we need to obtain *m* from input image experimentally. This parameter also can be obtained by product of maximum coefficient value of each sub band M_{ij} (i and j represents scale and orientation of sub band respectively) and a scaler parameter *k*. For example, assume that c=2, $\alpha = \sigma_{ij}$, and *k =0.25;* in this case, *m=k.* M_{ij}. Therefore, thresholds of the intervals in function defined in (26) are $2\sigma_{ij}$ and $0.25M_{ij}$.

Weights assigned to each term (k_1 , k_2 k_3) in (26) determine how much coefficients should be modified by this non-linear function. All these parameters should be tuned experimentally based on intrinsic characteristics of the input image to give the best contrast enhancement index. Table.1 shows intervals that parameter can be valued and chosen values that we assigned to the parameters in our experiment.

Table.1) Experimental tuned values for modifying curvelet coefficients using non-linear function defined in (26).

Parameters	**Parameters interval value**	**Set value in the experiment**
k_1	$k_1 \leq 1$	0.7
k_2	$k_2 \geq 1$	2
k_3	$k_3 \leq 1$	0.6
S	$0 \leq s \leq 1$	0.5
P	$0 \leq p \leq 1$	0.75

Here is the algorithm used to enhance image contrast.

Algorithm.1: Contrast enhancement algorithm using curvelet transform;

scales: $s = \{1, \ldots, N\}$;
rotations: r;
x is a curvelet coefficient;
obtain curvelet coefficients by applying FDCT via wrapping approach;
for s=1 to N do
for r=1 to 4. $2^{\lfloor s/2 \rfloor}$
estimate the noise standard deviation in each band: σ_{ij};
determine the value of m;
$x \leftarrow x * f(x)$
end
end
Reconstruct modified image using invers FDCT via wrapping approach;

where *N and 4.* $2^{\lfloor s/2 \rfloor}$ are number of scales and directions in sth scale respectively. To get the best contrast enhancement index, all defined parameters should be determined experimentally.

Step 2: Representing pixels of image into the feature space using Curvelet transform

In this step, we aim to represent every pixel of the image as a vector in the feature space. Capillary curves output bigger coefficients once they are convolved by curvelet functions that are fine and look like needle than curvelet functions that are coarse and non-directional; in contrast, vessel curves output bigger coefficients once they are convolved by coarse curvelet functions than fine and needle shaped curvelet functions. Curvelet transform calculates correlation between input image and all rotated and translated curvelets of a given scale. Typically, for a fixed scale and rotation, the closest curvelet to a certain pixel of a given image has more contribution to reconstruct the same pixel than other curvelet functions with same scale and rotation in inverse curvet domain; therefore, for a given pixel, the closest curvelet returns bigger coefficient than other curvelets with the same scale and rotation.

Now we keep scale and location (i.e. closest curvelet) parameters constant and change rotation of curvelet to find out the best rotated curvelet which outputs biggest coefficient among all other rotated curvelets. Once we found the best rotation and location of a given pixel, we can calculate coefficients for all scales to create feature vector. The thickness of a curve singularity can be determined by coefficients of all scales. Regardless of direction and location of the curvelets fitted on the curve, thinner curves output bigger coefficients with high scales and thicker curves output bigger coefficients with low scales. Thickness of the curve singularity is used as a discriminative feature to cluster capillaries and vessels using fuzzy C-means clustering method. Feature vector representing thickness of curve singularities for each pixel is created by coefficients of fittest curvelets in different scales. Capillaries pixels have large values at high scales and small values in low scales; while vessels pixels have large values at low scales and small values at high scales. In the other words, capillaries information mostly exists in fine and high scales; while vessels information mostly exists in coarse and low scales. The curvelet based feature extraction is described same as follows:

Algorithm.2: Curvelet based pixel representation in the feature space;

pixels of the input image: $P = \{ p_1, p_2, \ldots, p_{mn} \}$

scales: $S = \{ 1, \ldots, N \}$;

feature: $F = [1 \ldots, p_{mn} ; 1, \ldots, N]$;

rotations: r;

list= {};

for each pixel p_k *in P*

  for s=1 to N do

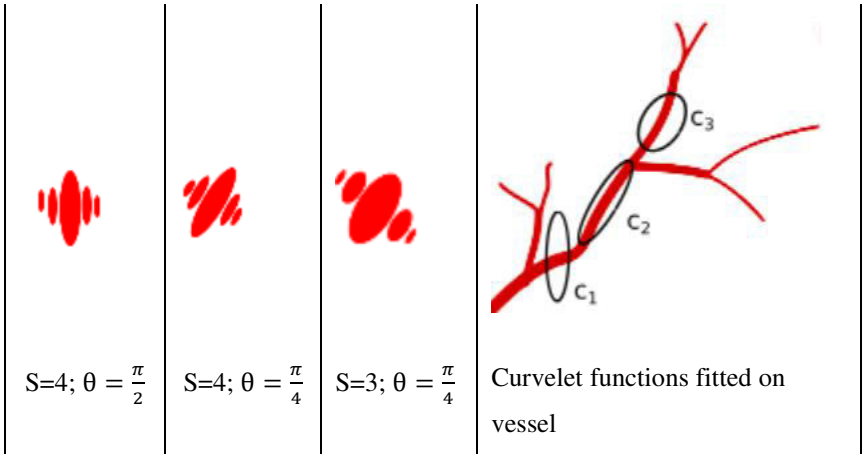
    for r=1 to 4. $2^{\lfloor s/2 \rfloor}$

    $(k_1, k_2) \leftarrow$ *index of the closest curvelet to* p_k

    list $\leftarrow$ *list* $\cup$ *C (s, r,* (k_1, k_2)*);*

    end

$F[p_k, s] \leftarrow max(list);$
$list \leftarrow \{\};$
end
end



S=4; $\theta = \frac{\pi}{2}$	S=4; $\theta = \frac{\pi}{4}$	S=3; $\theta = \frac{\pi}{4}$	Curvelet functions fitted on vessel

Fig.4 shows different curvelet functions with different scales and rotation angels fitted on vessels; curvelet coefficient related to c_2 is bigger than c_1 and c_3, c_3 is bigger than c_1; because thickness and direction of c_2 align better than c_3 and $c_1$on the input vessel.

Step 3: Segmentation using Fuzzy C-means clustering

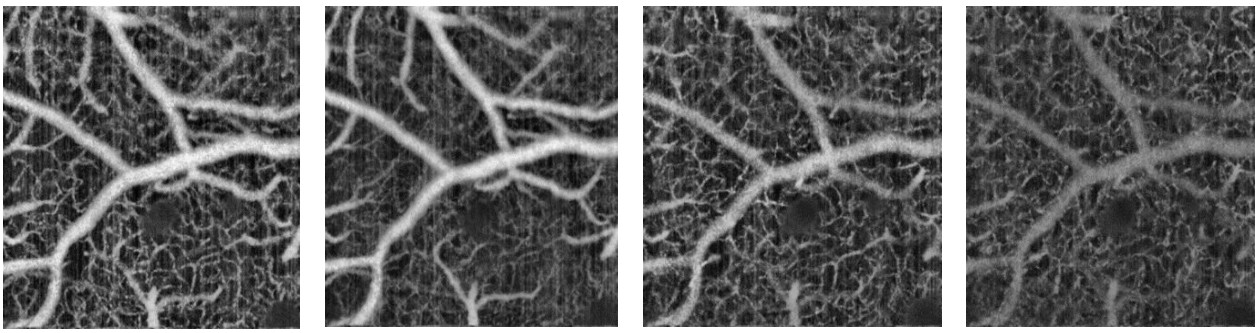
Brain tissue contains blood vessels and capillaries which have different widths. As discussed in the second step; this property (i.e. thickness of the curve singularities) measured by curvelet transform creates a set of discriminative feature vectors in the feature space in a way that pixels of the capillaries output big coefficient values with large scaled curvelets and small values with small scaled curvelets. On the other hand, pixels of the vessels output big coefficient values with small scaled curvelets and small values with large scaled curvelets. Therefore, this property of feature vectors associated to capillaries and vessels creates similarities between pixels of each cluster. Basically, by using this approach to represent pixels in the feature space, data of each cluster gets close to each other and separates from data of other clusters in the feature space. This data representation method in the feature space causes discrimination between two clusters. After extracting feature from all pixels of the image, FCM clusters them. FCM assigns a finite membership value to each cluster for a given pixel and finally this pixel is assigned to a cluster based on its maximum membership value.

Experimental result and discussion

1. Dataset

We applied our proposed algorithm on the images that have been produced by joint-collaboration of NITRO and BIST labs at the University of Wisconsin- Madison. In this experiment, we have used reinforced thinned-skull windows to recover animal after surgery as well as suppressed potential brain pulsations during data acquisition to enhance quality of the images and precision

of the measurements. To observe more capillaries density in the brain tissue, we should scan from deep layer tissues of the brain; however, due to existing noise in the imaging system and upper layer tissues shadowing, we lose resolution once we scan from deep layer tissues. Exponential results show that the quality of the image is decreased significantly once data is acquired deeper than 125 microns.



a) depth 1-150 microns

b) depth 25-60 microns

c) depth 50-85 microns

d) depth 50-125 microns

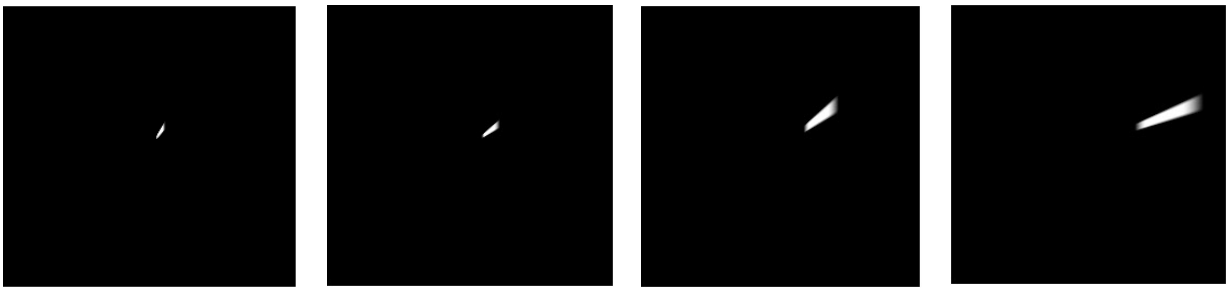
Fig.5: Produced images from different depths of the brain.

Due to higher capillary density and resolution in image (a) rather than other above images, we use image (a) in our experiment.

2. Curvelet functions in frequency and spatial domains

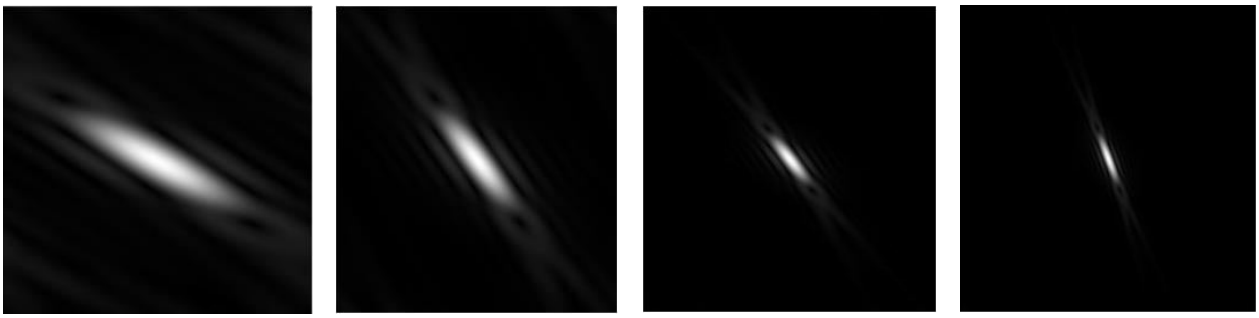
Figure 1-4 demonstrate curvelet functions and their frequency response at different scales and orientations; these are some the functions that we have used to obtain curvelet coefficients of input image; each coefficient represents convolution of input image and one of curvelet functions determined by scale, orientation and translation. Convolution of input image and curvelet function in spatial domain corresponds to their multiplication in frequency domain. Since we have used wrapping implementation of curvelet transform; after multiplying FFT of curvelet function and FFT of the Image, we need to wrap this product around the origin and then we apply inverses FFT to obtain curvelet coefficients. Figures 1-4 show that curvelet functions at higher scales cover higher and bigger frequency bands; they are very fine and look like a needle in spatial domain. These kernels are appropriate candidate for capillaries representation in the curvelet domain; while curvelet functions in lower scales, cover lower and smaller frequency bands; they are very coarse and non-directional in spatial domain. These kernels are appropriate candidate for vessel representation in curvelet domain.

Maximum number of scale depends on size of input image; tangent of rotation angels at scale *j* is obtained as follows $tan\theta_{jl} = l.2^{-\lfloor j/2 \rfloor} : \quad l = \{2^{-\lfloor j/2 \rfloor} + 1, \ldots, 2^{\lfloor j/2 \rfloor} + 1\}$. Number of scale is recommended to be *log (N)-3* where N is the size of the input image N ×N.



a) scale 3 b) scale 4 c) scale 5 d) scale 6

Fig.6: Curvelet functions with different scales and rotation of 45° in frequency domain.

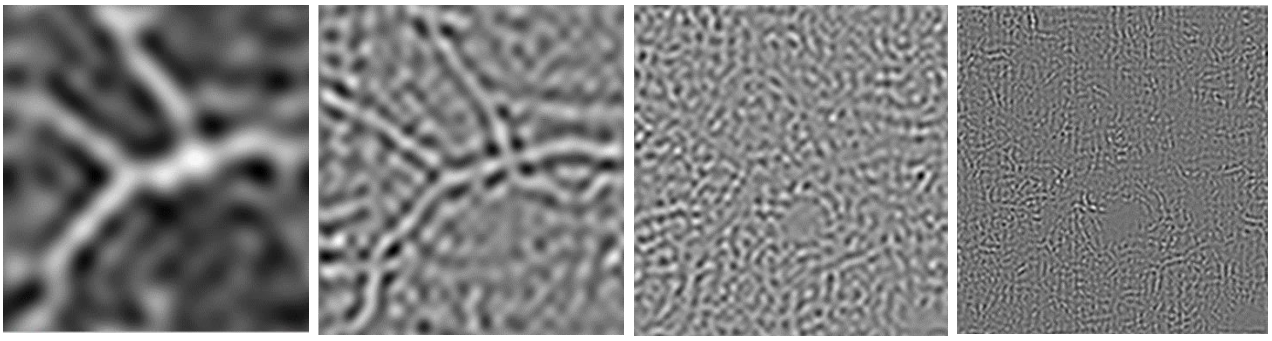


a) scale 3 b) scale 4 c) scale 5 d) scale 6

Fig.7: Curvelet functions with different scales and rotation of 45° in spatial domain.

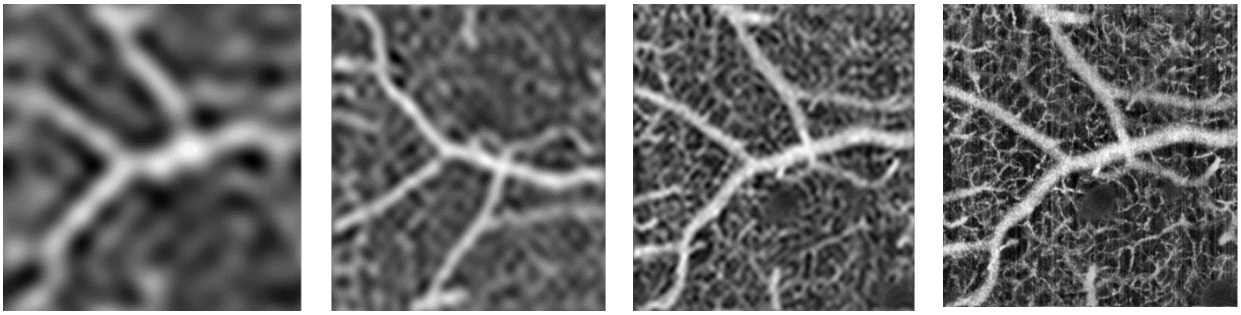
3. Response of the input image to different scaled curvelets

Fig. 8 shows convolution of input image with different curvelet functions; the results show that capillaries mostly exist at larger scales in the curvelet domain; while vessels mostly exist at smaller scales in the curvelet domain. Fig.10 (d) shows power spectrum of the image at different scales. Because most part of the energy of the image exist in the vessels rather than capillaries, power of the signal decrease as scales increase. Fig.9 shows reconstructed images from different scales using inverse FDCT via wrapping method. Fig. 9 shows, as we use higher scales coefficients to reconstruct image, we recover more details and information about capillaries. As it shown in the Fig.9 (d), we can reconstruct image completely without losing any information.



a) scale 3 b) scale 4 c) scale 5 d) scale 6

Fig.8: Convolution of input image with curvelet functions with different scales.



a) Scale 1 b) Scales 1 to 2 c) Scales 1 to 3 d) Scales 1 to 6

Fig.9: Reconstructed images using coefficients of different scales; a) shows reconstructed image using coefficients of scale 1; b) shows reconstructed image using coefficients of scale 1 and 2; c) shows reconstructed image using coefficients of scale 1 up to 3; d) shows reconstructed image using coefficients of scale 1 up to 6.

4. Contrast enhancement using tuned parameters

We tune parameters defined in (18) to give us best contrast enhancement performance. To do this, we evaluate tuned parameters quantitatively using peak signal-to-noise ratio (PSNR), structural similarity index measure (SSIM) and contrast improvement index (CII) measures. We describe all these measures comprehensively as follows.

PSNR is the ratio between the maximum possible power of a signal and the power of corrupting noise that affects the fidelity of its representation. PSNR is defined as follows:

$$MSE = \frac{\sum_i \sum_j (X(i,j) - Y(i,j))^2}{M \times N} \tag{27}$$

$$PSNR = 10 \log \frac{(255)^2}{MSE} \tag{28}$$

SSIM is an approach for measuring the similarity between two images. The SSIM index is a full reference metric. SSIM is defined as follows:

$$SSIM = \frac{\left(2\mu_x\mu_y + c_1\right)\left(2\partial_{xy} + c_2\right)}{\left(\mu_x^2 + \mu_y^2 + c_1\right)\left(\partial_x^2 + \partial_y^2 + c_2\right)} \tag{29}$$

where X and Y are the noisy and restored images, respectively. ∂_{xy} is the covariance of X and Y $c_1 = (k_1 L)^2$, $c_2 = (k_2 L)^2$ are two variables to stabilize the division with weak denominator; L is the dynamic range of pixel values (typically this is $(2^{(\#bitsperpixel)} - 1)$). $k_1 = 0.01$ and $k_2 = 0.03$ by default. The measurements or predictions of image quality are based on comparing the restored image with the original, uncorrupted and free of noise image as the reference.

Contrast improvement index (CII) is an index that quantitively measures contrast improvement of a given image. CII is defined as follows:

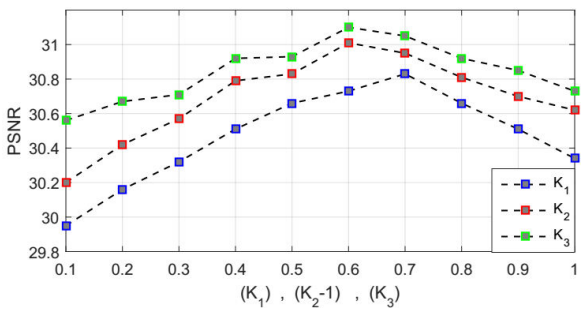
$$CII = \frac{C_{\text{enhanced}}}{C_{\ original}} \tag{30}$$

where C_{enhanced} and $C_{\ original}$ are the contrast for the proposed and original images, respectively. The contrast C of an image is defined as

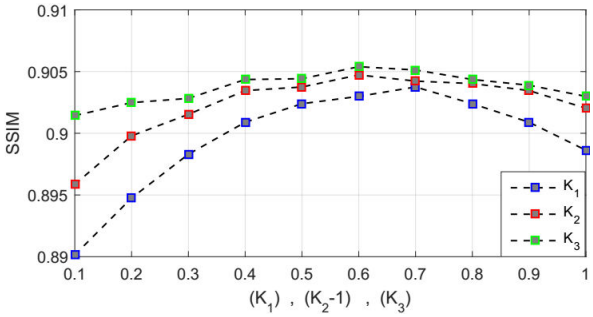
$$C = \frac{r - b}{r + b} \tag{31}$$

where r and b are the mean gray-level value of the foreground and the background, respectively.

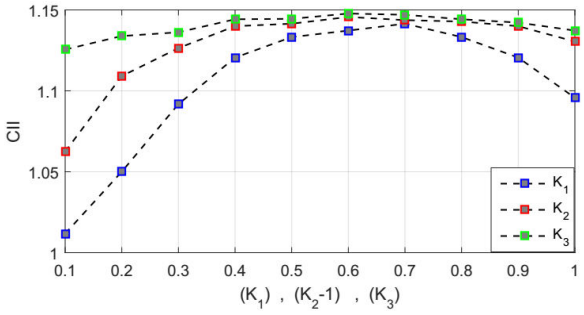
Fig.10 shows graphs of PSNR, SSIM and CII measured by different parameters values defined in (18). As it is shown, parameters should set to $k_1 = 0.7$, $k_2 = 2$, $k_3 = 0.6$ to produce the best performance in our proposed algorithm.



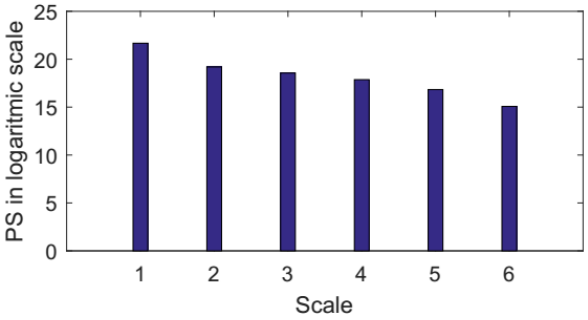
a) PSNR



b) SSIM



c) CII



d) Power spectrum

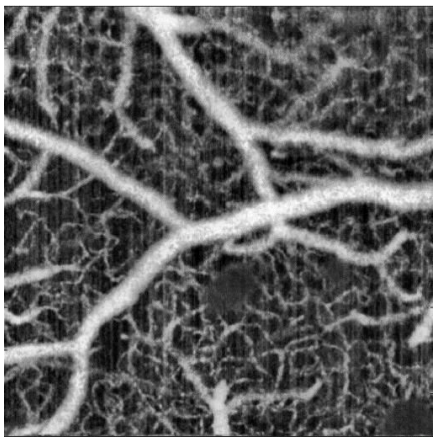
Fig.10 shows PSNR, SSIM and CII of enhanced image with different parameters value

Table.2 shows average and standard deviation of SSIM, PSNR and CII in our experiment using different parameters values.

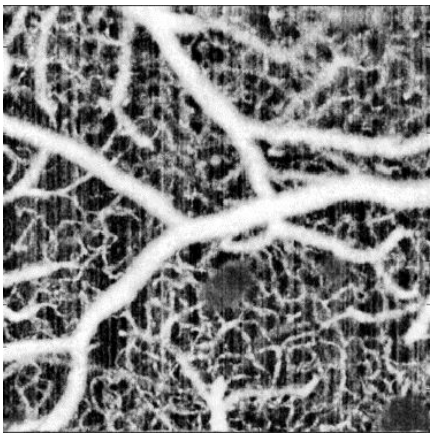
Table. 2) Image enhancement quantitative assessment results using Curvelet transform

Measure	Average	Standard deviation
SSIM	0.953	0.18
PSNR	30.53	1.67
CII	1.09	0.046

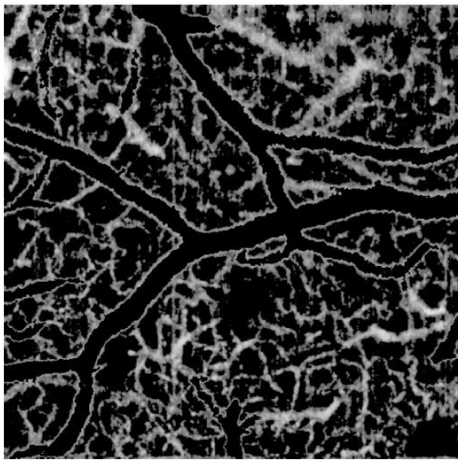
Fig. 11 shows the output of proposed algorithm for contrast enhancement, vessels and capillaries segmentation.



a) Input image

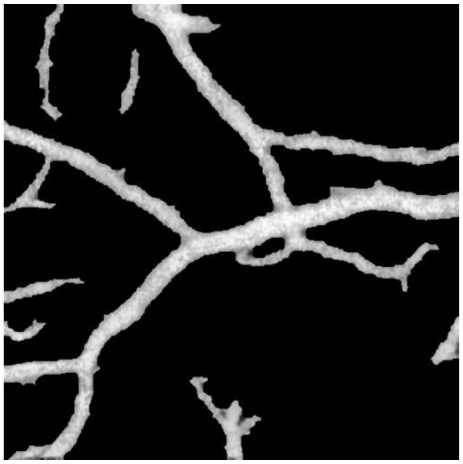


b) Contrast enhanced

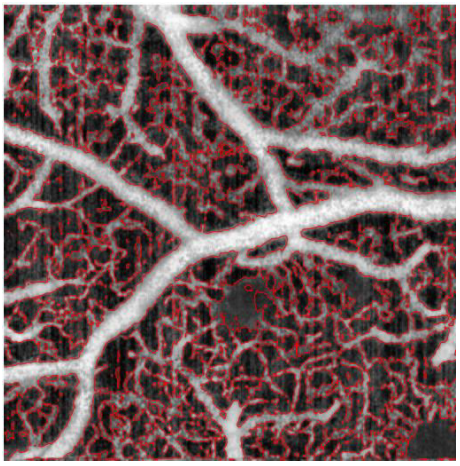


c) Segmented capilaries

d) Skeleton of capilaries



e) Segmented vessels



f) Manual segmented

Fig.11 shows output of proposed algorithm; a) input image, b) shows output of algorithm in the second step for contrast enhancement, c) shows segmented capillaries, d) shows skeleton of the capillaries that can be an index of capillary density for diseases detection, e) show segmented vessels and f) represents manual segmented of input image for comparing of different algorithms performance.

5. Segmentation Assessment:

To assess the algorithm performance, three measures are used: true positive rate (TPR), false positive rate (FPR), and accuracy (ACC). Assume TP and TN show the blood vessel and capillaries pixels and background pixels which correctly detected respectively. FP shows the pixels not belonging to a vessel, but is recognized as blood vessel pixels, and FN shows the pixels belonging to a vessel but is recognized as background pixels mistakenly. Based on definitions, TPR, FPR, and Accuracy are defined as follows:

$$TPR = \frac{TP}{TP + FN} \tag{32}$$

$$FPR = \frac{FP}{FP + TN} \tag{33}$$

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} \tag{34}$$

Table.3 and 4 show performance evaluation of different algorithms for capillaries and vessels segmentation. Experimental results show that proposed method outperforms some of existing segmentation methods in terms of accuracy.

Table.3) Performance accuracy comparison between proposed algorithm and other methods for capillaries

Method	**FPR**	**TPR**	**Accuracy**
Method [9]	3.14%	83.72%	92.96%
Method [13]	4.09%	82.64%	92.12%
Method [14]	3.81%	82.95%	92.48%
Method [15]	4.52%	81.78%	91.01%
Method [16]	4.88%	81.15%	90.71%
Fuzzy C-means	5.06%	80.23%	90.08%
Proposed method	**2.98%**	**84.77%**	**93.54%**

Table.4) Performance accuracy comparison between proposed algorithm and other methods for vessels

Method	**FPR**	**TPR**	**Accuracy**
Method [9]	1.68%	88.61%	95.61%
Method [13]	1.92%	88.11%	95.09%
Method [14]	1.75%	88.34%	95.26%
Method [15]	2.05%	87.31%	94.19%

Method [16]	3.07%	85.97%	93.21%
Fuzzy C-means	3.46%	85.01%	92.69%
Proposed method	**1.03%**	**89.77%**	**96.82%**

Conclusion

In this paper, a curvelet based method has been proposed to segment capillaries and vessels in OCT angiograms. OCT imaging system has limitation to capture information from deep layer tissues in the brain but still there exists high amount of information about hemodynamic in the vessel and capillaries density. Even though there exists higher capillaries density in deeper layer tissues in the brain; we lose image resolution if we acquis image from deeper layer tissues. In this work, curvelet transform as one of multi resolution analysis is applied to strengthen faint capillaries and soften vessels in OCT angiogram captured from deeper layer tissues and then a feature vector is defined for each pixel based on fittest curvelet functions in different scales. This step of the algorithm determines thickness of curve singularities in test image. Experimental results show this feature representation is a proper way to represent capillaries and vessels in a discriminative feature space. And finally, a well-tuned clustering approach namely fuzzy C-means is applied to cluster capillaries, vessels and background into three different clusters. The experimental results show superiority of the proposed method in comparison to some existing segmentation methods.

Acknowledgment

We would like to thank BIST lab at the University of Wisconsin Milwaukee for providing and sharing the images that we can use in this experiment.

References

1. Atry, F. and R. Pashaie, *Analysis of intermediary scan-lens and tube-lens mechanisms for optical coherence tomography.* Applied optics, 2016. **55**(4): p. 646-653 %@ 1539-4522.
2. Schendel, A.A., et al., *The effect of micro-ECoG substrate footprint on the meningeal tissue response.* Journal of neural engineering, 2014. **11**(4): p. 046011 %@ 1741-2552.
3. Atry, F., et al., *Monitoring cerebral hemodynamics following optogenetic stimulation via optical coherence tomography.* IEEE Transactions on Biomedical Engineering, 2015. **62**(2): p. 766-773 %@ 0018-9294.
4. Azimipour, M., F. Atry, and R. Pashaie, *Effect of blood vessels on light distribution in optogenetic stimulation of cortex.* Optics letters, 2015. **40**(10): p. 2173-2176 %@ 1539-4794.
5. Park, D.-W., et al., *Graphene-based carbon-layered electrode array technology for neural imaging and optogenetic applications.* Nature communications, 2014. **5**.

6. Niemeijer, M., et al. *Comparative study of retinal vessel segmentation methods on a new publicly available database*. 2004. SPIE.
7. Fraz, M.M., et al., *Blood vessel segmentation methodologies in retinal images–a survey.* Computer methods and programs in biomedicine, 2012. **108**(1): p. 407-433 %@ 0169-2607.
8. Taherkhani, F., *Modelling of spatial causality among distinctive properties of an image using conditional random field for image classification.* The Imaging Science Journal, 2017: p. 1-9 %@ 1368-2199.
9. Oliveira, W.S., et al., *Unsupervised retinal vessel segmentation using combined filters.* PloS one, 2016. **11**(2): p. e0149943 %@ 1932-6203.
10. Chaudhuri, S., et al., *Detection of blood vessels in retinal images using two-dimensional matched filters.* IEEE Transactions on medical imaging, 1989. **8**(3): p. 263-269 %@ 0278-0062.
11. Frangi, A.F., et al. *Multiscale vessel enhancement filtering*. 1998. Springer.
12. Soares, J.V.B., et al., *Retinal vessel segmentation using the 2-D Gabor wavelet and supervised classification.* IEEE Transactions on medical Imaging, 2006. **25**(9): p. 1214-1222 %@ 0278-0062.
13. Bankhead, P., et al., *Fast retinal vessel detection and measurement using wavelets and edge location refinement.* PloS one, 2012. **7**(3): p. e32435 %@ 1932-6203.
14. Dai, P., et al., *A new approach to segment both main and peripheral retinal vessels based on gray-voting and gaussian mixture model.* PloS one, 2015. **10**(6): p. e0127748 %@ 1932-6203.
15. Vlachos, M. and E. Dermatas, *Multi-scale retinal vessel segmentation using line tracking.* Computerized Medical Imaging and Graphics, 2010. **34**(3): p. 213-227 %@ 0895-6111.
16. Heneghan, C., et al., *Characterization of changes in blood vessel width and tortuosity in retinopathy of prematurity using image analysis.* Medical image analysis, 2002. **6**(4): p. 407-429 %@ 1361-8415.
17. Kim, B.-G., J.-I. Shim, and D.-J. Park, *Fast image segmentation based on multi-resolution analysis and wavelets.* Pattern Recognition Letters, 2003. **24**(16): p. 2995-3006 %@ 0167-8655.
18. Wang, J.Z., et al., *Unsupervised multiresolution segmentation for images with low depth of field.* IEEE Transactions on Pattern Analysis and Machine Intelligence, 2001. **23**(1): p. 85-90 %@ 0162-8828.
19. Daubechies, I., *The wavelet transform, time-frequency localization and signal analysis.* IEEE transactions on information theory, 1990. **36**(5): p. 961-1005 %@ 0018-9448.
20. Do, M.N. and M. Vetterli, *The finite ridgelet transform for image representation.* IEEE Transactions on image Processing, 2003. **12**(1): p. 16-28 %@ 1057-7149.
21. Ma, J. and G. Plonka, *The curvelet transform.* IEEE signal processing magazine, 2010. **27**(2): p. 118-133 %@ 1053-5888.
22. Candes, E.J. and D.L. Donoho. *Curvelets, multiresolution representation, and scaling laws*. 2000. International Society for Optics and Photonics.
23. Candes, E.J., *Ridgelets: theory and applications*. 1998, Stanford University.
24. Candès, E.J. and D.L. Donoho, *Curvelets and curvilinear integrals.* Journal of Approximation Theory, 2001. **113**(1): p. 59-90 %@ 0021-9045.
25. Candes, E.J. and D.L. Donoho, *Continuous curvelet transform: I. Resolution of the wavefront set.* Applied and Computational Harmonic Analysis, 2005. **19**(2): p. 162-197 %@ 1063-5203.
26. Donoho, D. and E. Candes, *Continuous curvelet transform: II. Discretization and frames.* Applied and Computational Harmonic Analysis, 2005. **19**(2): p. 198-222.
27. Candès, E.J. and D.L. Donoho, *New tight frames of curvelets and optimal representations of objects with piecewise C2 singularities.* Communications on pure and applied mathematics, 2004. **57**(2): p. 219-266 %@ 1097-0312.
28. Bezdek, J.C., R. Ehrlich, and W. Full, *FCM: The fuzzy c-means clustering algorithm.* Computers & Geosciences, 1984. **10**(2-3): p. 191-203 %@ 0098-3004.

29. Candes, E., et al., *Fast discrete curvelet transforms.* Multiscale Modeling & Simulation, 2006. **5**(3): p. 861-899 %@ 1540-3459.
30. Klir, G. and B. Yuan, *Fuzzy sets and fuzzy logic*. Vol. 4. 1995: Prentice hall New Jersey.
31. Cannon, R.L., J.V. Dave, and J.C. Bezdek, *Efficient implementation of the fuzzy c-means clustering algorithms.* IEEE Transactions on Pattern Analysis and Machine Intelligence, 1986(2): p. 248-255 %@ 0162-8828.
32. Bezdek, J.C., et al., *Convergence theory for fuzzy c-means: counterexamples and repairs.* IEEE Transactions on Systems, Man, and Cybernetics, 1987. **17**(5): p. 873-877 %@ 0018-9472.