



# Brain Blood Vessel Segmentation in Hyperspectral Images Through Linear Operators

Guillermo Vazquez<sup>1</sup>, Manuel Villa<sup>1</sup>, Alberto Martín-Pérez<sup>1</sup>,  
Jaime Sancho<sup>1</sup>, Gonzalo Rosa<sup>1</sup>, Pedro L. Cebrián<sup>1</sup>, Pallab Sutradhar<sup>1</sup>,  
Alejandro Martínez de Ternero<sup>1</sup>, Miguel Chavarrías<sup>1</sup>,  
Alfonso Lagares<sup>2</sup>, Eduardo Juárez<sup>1</sup>, and César Sanz<sup>1</sup>

<sup>1</sup> Universidad Politécnica de Madrid, Madrid, Spain

{guillermo.vazquez.valle,manuel.villa.romero,a.martinp,jaime.sancho,  
gonzalo.rosa.olmeda,pl.cebrian,pallab.sutradhar,a.mruiz,  
miguel.chavarrias,eduardo.juarez,cesar.sanz}@upm.es

<sup>2</sup> Instituto de Investigación Sanitaria Hospital 12 de Octubre, Madrid, Spain  
alfonso.lagares@salud.madrid.org

**Abstract.** Tissue classification tasks that rely on multidimensional data, such as spectral information, sometimes face issues related to the nature of their own characteristics when different biological components share similar spectrum. In a situation of in-vivo brain tumor location during a surgical operation, especially when applying machine learning techniques, relying solely on the spectral information of each sample may not be enough to provide a correct identification of all the tissues involved. In order to overcome this problem, in this work we propose to reduce conflicting classification pixels, i.e. vascular versus tumor tissues. To do so, morphological operators can supply support to a pixel-wise classification by exploiting the spatial characteristics present in vascular tissue. Hence, we have evaluated the suitability of linear operators for brain vessels segmentation in a context of hyperspectral video classification. The parameters of the operator along with the selection of the most suitable spectral band to process were chosen via optimization of the amount of vascular pixels detected and error metrics. The segmentation algorithm was implemented for both CPU and GPU platforms achieving a performance compatible with real-time classification purposes on the last one. Objective results show an average segmentation of the 68% of the vein and arteries present in the ground truth with less than a 10% of error selecting pixels from other tissues of interest such as healthy brain and tumor.

**Keywords:** Hyperspectral · segmentation · brain vessels · GPU

---

This work was supported by the Spanish Government through TALENT-HIPSTER project (PID2020-116417RB-C41).

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2023  
M. Chavarrías and A. Rodríguez (Eds.): DASIP 2023, LNCS 13879, pp. 28–39, 2023.  
[https://doi.org/10.1007/978-3-031-29970-4\\_3](https://doi.org/10.1007/978-3-031-29970-4_3)

## 1 Introduction

Non-invasive diagnosis through different digital imaging methods have accumulated a long history in which their integration into medicine has made the detection and treatment of certain diseases easier. In this field, hyperspectral imaging (HSI) has proven its usefulness in the optical characterization of the tissues [1, 2]. Through the spectral sampling of the reflectance captured from the scene, each biological component can provide what is called their spectral signature: a particular way each tissue disperses the incident light across the spectrum. This source of information, when combined with machine learning algorithms, can serve as a classification tool for certain pathologies such as tumor formations [3]. The problem comes when the spectral signatures are very similar between tissues, causing the classifier to mistake them. The consequence of this issue can be observed in [3], where the classification maps show a tendency to mark certain regions of the veins and the arteries on the surface of the brain as tumor despite their morphology being completely different. The ultimate goal of this classifier is to assist the surgeon performing the resection of the tumor by indicating its limits through a classification map. Therefore, any source of tumor false positives must be suppressed.

The mapping of the brain blood vessels through non-invasive techniques supposes a useful mechanism for the diagnosis of vascular pathologies and also assistance in surgical interventions. Normally, this mapping process is performed out from magnetic resonance images (MRI) or time of flight magnetic resonance angiography (ToF-MRA) where a series of images from transversal sections con-forms a volumetric representation of the head. The MRI and MRA images are capable of providing high spatial resolution of the internal structures of the brain. The existing literature that deals with the blood vessel segmentation matter from this kind of imaging is abundant: whereas deep learning-based techniques have gained popularity [4] in the last decade due to its prominent results [5], more traditional approaches based on morphological operations [6] are still capable of providing sufficient accuracy compared to nowadays standards. However, in situations where it is required to perform a segmentation of the cortical blood vessels of the brain through an open craniotomy, the images provided by an MRI scan become harder to exploit. This is because of the shifts the brain suffers during the surgical procedure causing the MRI to be difficult to match with the image from an external camera capturing the brain cortex. The set of techniques that relies only on external camera captures for the segmentation of vascular tissue is more scarce than those using the MRI and MRA imaging. Most of the literature focuses on ophthalmology applications for identifying the blood vessel structures [7] but when it comes to a brain surgery context, one of the few examples can be found in Wu et al. [8] where deep learning techniques are used to carry on the segmentation of the vessels of the mouse cerebral cortex. Also, in Fabelo et al. [9] this problem is addressed by classifying brain vascular tissue as a stage of a brain tumour detector. The use of neural networks for segmentation, such as the U-Net [10], has a particular aspect related to the training of the net: it requires a dense ground truth that has all the vascular elements to

be segmented labeled. If only a sparse ground truth is available, reconstruction processes like [11] can provide adequate training of the network, but they are only effective when there are certain gaps in the ground truth.

Some of the latest camera models equipped with snapshot sensors can capture hyperspectral video (HSV) bringing the opportunity and the challenge of performing a real-time classification over the captured sequence [12]. In this work we propose an efficient implementation of morphological operators based on GPU platforms. The goal is to perform a real-time segmentation of the brain vascular structures captured in a hyperspectral video from an in-vivo surgical intervention. This segmentation is intended to refine a classification map obtained from a classifier based on support vector machines (SVM) trained to detect brain tumor tissue. Hence, the importance of the efficiency of this correcting stage.

## 2 Background

The work described in this paper rests on two main basis:

1. Line operators: the work introduced by E. Ricci and R. Perfetti in [7] can be considered as the core of the proposed algorithm. Along with [6], it serves as an example of the suitability of this kind of operator for detecting blood vessels in an RGB image. Although in this paper the work material consists on hyperspectral images, they are processed as if they were regular captures. In [7], through a series of linear filters oriented by a constant increasing angle of  $15^\circ$ , the detector processed the inverted green channel of a non-mydratiac retinal image obtained from the DRIVE [13] and STARE datasets [14]. The aim of the detector is to capture those vessels aligned with any of the linear filters to mark them in the output image. This linear detection is combined with an SVM to perform a binary classification. According to the results presented in the tables III and IV of the section IV in [7], the simple linear operator only performs a 0,8% less in the area under the curve of the ROC curve worse than the linear detector and SVM combined, at worst over the STARE dataset.
2. GPU data processing: medical image processing has experienced an important step forward thanks to the usage of GPU acceleration [15], either to shorten the computing time while processing heavy inputs such as MRI scans or to make real-time assistance possible. Particularly, filtering algorithms can greatly benefit from the parallelization the GPU architecture offers, outperforming substantially its homologous CPU implementations [16].

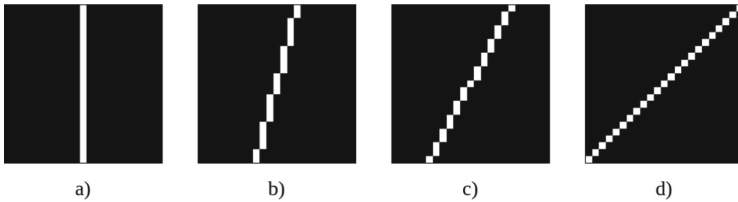
## 3 Algorithm and Implementation

As it was described in the previous section, the algorithm proposed in this work is based on the linear detector introduced in [7], with the difference that instead of processing RGB images, the proposed algorithm receives as input a single hyperspectral frame provided by a first generation Ximea MQ022HG-IM-SM5X5

snapshot camera. This camera can stream up to 170 FPS with a resolution of  $2045 \times 1085$  pixels capturing 25 different spectral bands that go from 968.93 nm up to 693.74 nm. The 25 filters employed to extract each band are arranged in a mosaic pattern of  $5 \times 5$ , conforming the information of a single pixel. This pattern is replicated across the entire sensor, therefore, the hyperspectral cube formed from each frame has a shape of  $409 \times 217$  pixels for its spatial resolution and 25 bands for each one of them.

Once the hyperspectral cube is built from the raw frame it needs to be black/white calibrated and spectral corrected. This process can also be accelerated in a GPU platform as it is shown in the work presented by M. Villa et al. [12]. The algorithm proposed is designed to work with gray scale images so, once the calibration and the spectral correction are performed, a single band from the cube is selected and its luma is inverted so the dark contours corresponding to the vessels are marked with high brightness values. As it will be described in Subsect. 3.1, the decision on which band is to be taken is based on an optimization process to analyze which one is the most suitable for the segmentation task.

The next step is to apply the linear operators to the selected band. The operator as such, is composed by 12 different kernels, each of them defined as a zero matrix with a straight line running across the center of the matrix composed by ones, as exemplified in Fig. 1. Each one of these 12 lines is therefore intended to cover 12 possible orientations a contour could take in the image. The angular stride between each linear operator is fixed at  $15^\circ$ .

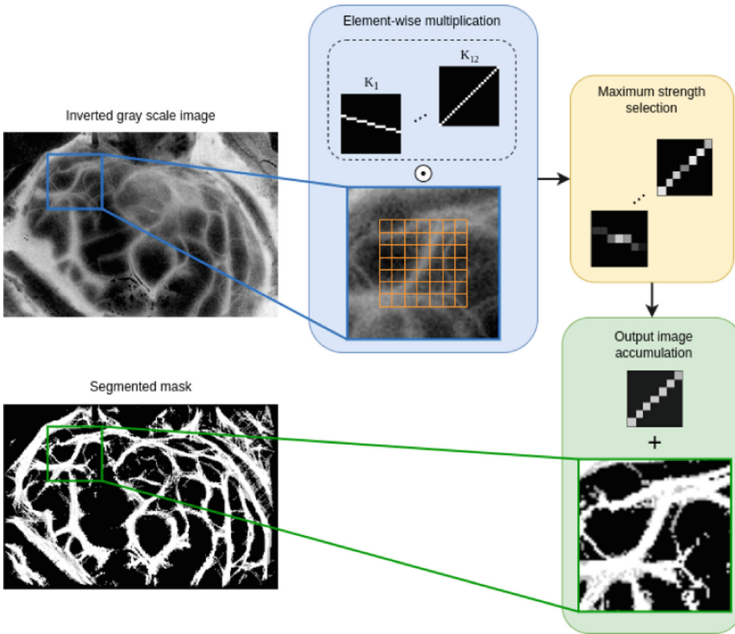


**Fig. 1.** Examples of different kernels at: (a)  $0^\circ$ , (b)  $15^\circ$ , (c)  $30^\circ$  and (d)  $45^\circ$

Due to the wide range of width brain blood vessels present, one single window filter size cannot cover all its variety. To overcome this issue, two different squared window sizes are applied independently: a smaller window within the range of 11 to 15 of pixels size for detecting the capillaries and thin vessels, and a bigger window for the arteries and thick veins that covers the 15 to 31 size range. As well as the spectral band to be used, the window size of both operators is selected via optimization.

For performing the detection of any delineation that can be part of a blood vessel, every single kernel of the linear operator multiplies element-wise an aligned region of the same window size from the gray scale image. Through

this, each kernel registers the gray level that falls into its line for all the 12 possible orientations defined. Then, for evaluating which orientation is more likely to have captured an actual vessel, the average brightness of the region of the gray scale image under the position of the operator is subtracted from each average gray level captured by the kernels. In [7], the resulting value is denoted as the line strength of the kernel. The kernel with the highest strength is selected and its value is accumulated in an output image across the length and orientation of its line. This process is repeated across the entire image moving the operator as a sliding window with stride 1. Because of the summation of the strengths delivered by the operator on the output image, the sharpest contours will reach values that exceed the maximum gray level that can be represented with the bit resolution of the gray scale image. Figure 2 illustrates the process of applying one operator to the selected spectral band for obtaining the accumulated image that conforms the segmented mask.



**Fig. 2.** Block diagram of the proposed algorithm for a single operator.

To reduce the noise introduced by the linear operator caused by partial contours, a threshold is applied to the output image by a minimum level of gray. This threshold is also fixed by the optimization process.

Since thin vessels can result in much lower strength values than thick veins or arteries, the rescaling of the output image to a range of values that can be represented with an unsigned integer, cannot be performed linearly. Given the

approximate exponential decay shape the output image histogram has, a linear rescale would cause the weaker values to be truncated to zero. Therefore, to avoid this issue, the following logarithmic correction is proposed:

$$I = \frac{\ln(S + 1) \cdot (2^n - 1)}{\ln(|\max S| + 1)} \quad (1)$$

where it is described the operations performed pixel-wise to the output image  $S$  to obtain the final image  $I$  where  $n$  is the resolution in bits.

### 3.1 Optimization

As described in Sect. 3, the value of certain parameters of the algorithm, such as the window size of the linear operator or the band to be processed and the thresholds of gray level for both operators, have a big impact on the output image. To ensure and accelerate the search for the best combination of parameters, their choice was made through an optimization process using the framework Optuna [17]. As it can be seen in [18] and in [19] Optuna is commonly used for optimizing hyperparameters in ML based systems for improving the training process.

The Optuna framework examines possible combinations of a set of parameters given a range to explore in a certain number of trials. For that process, it needs at least one output variable computed by the function that uses the parameters that are to be optimized so its outcome can be whether maximized or minimized. In this paper, Optuna is employed to maximize the number of vascular pixels that fall under the segmented region. Because of the particular use case addressed in this work, it is also decided to use Optuna to simultaneously minimize the tumor pixels selected. The optimization is carried out using a set of 10 hyperspectral images captured from 10 different brain tumor surgeries at Hospital Universitario 12 de Octubre in Madrid (Spain). The study was conducted according to the guidelines of the Declaration of Helsinki, and approved by the Research Ethics Committee of Hospital Universitario 12 de Octubre, Madrid, Spain (protocol code 19/158, 28 May 2019). Each hyperspectral image has its corresponding ground-truth map where not every sample but a certain amount of pixels from healthy and tumor tissue, blood vessels and dura mater are labeled. Despite the sparse content of the ground-truth, it can be calculated the average percentage of vascular and tumor ground-truth samples that have been included in the segmented area by using certain combinations of spectral band, window sizes and gray thresholds. This process is carried out through 90 trials.

Once the optimization is finished, Optuna provides the combination of parameters that delivers the highest value for the maximized metric and the set of parameters that offers the lower value for the minimized metric. Since none of these two cases are the most desired, it is necessary to analyze the trade-offs in the Pareto front that is formed with the results of the rest of the trials. Among all of the 90 calculated results, the selection is made around 3 combinations whose parameters can be seen in Table 1. Here, the 3 combinations of parameters chosen, denoted with the letters A, B and C are shown. Each combination consists

of the spectral band selected to be processed, the window size and the threshold for the gray level filtering for each operator, and the metrics obtained over the optimization set of images. In this case, the two bands selected, 15 and 16, have a wavelength of 891.79 nm and 900.40 nm respectively. As it can be seen, these 3 points are selected because they offer the best balance between high numbers of blood vessel samples segmented with low tumor error.

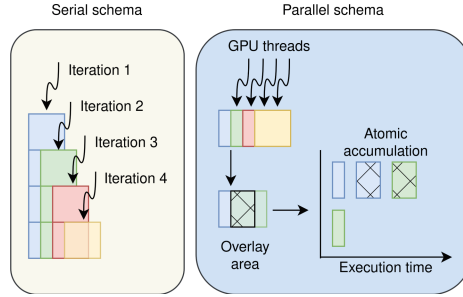
**Table 1.** Combination of parameters for the 3 selected points with their corresponding percentage of selected tissue.

Comb.	Band	Window 1	Thresh 1	Window 2	Thresh 2	Avg. Vessel	Avg. Tumor
A	16	11	266	23	274	0.81	0.08
B	15	13	120	29	348	0.85	0.11
C	16	11	266	31	274	0.90	0.15

### 3.2 Acceleration

One of the requirements of the algorithm is to be able to perform the segmentation at a sufficient speed that allows its integration on a real-time classification chain. Therefore, its acceleration is a central matter that conditions its usefulness. This process was carried on using a GPU programmed in the CUDA language.

The efficient parallelization of the algorithm described in 3 has certain hurdles to overcome. The usage of different kernels to each individual pixel in the image is the most resource-intensive stage of blood vessel segmentation. These kernels are considered as windows that surround each pixel. Implementing this component in a GPU is challenging due to the serial nature of this operation, which causes an overlap between the windows of various pixels. To perform this procedure in parallel, each thread is responsible for applying the filters per pixel and computing the element-wise multiplication between the filters and the window created around each pixel, followed by the computation of the strength as specified in 3. The different kernels are accessed several times during the application of the filters to the pixels of the image. The values are then stored in shared memory at the start of the process to reduce the impact of the massive memory access pattern. In this manner, the global memory access is reduced, improving memory throughput. Finally, in order to prevent race circumstances, the strength between various threads has been added in an atomic fashion, obtaining the strength accumulation per pixel at the conclusion of this process. Figure 3 depicts the differences existing between the serial implementation, where there only exists one overlapping area each iteration, and the parallel implementation, in which the calculation of these overlapping areas is performed in separate threads. Therefore, to ensure the proper accumulation, it must be necessarily atomic.



**Fig. 3.** Comparison between the serial implementation and the parallelization in the GPU

Thanks to there is no data-dependencies in the processing of the pixels, the other components of the segmentation process do not require any specific implementation. Instead, grid-stride loops are used in the GPU kernels in this scenario for maximum GPU performance.

## 4 Experiments and Results

To evaluate the linear operator under the set of selected parameters, a list of 5 hyperspectral images, different from the ones used during the optimization 3.1, were chosen. These five images also were taken during brain tumor operations at the same healthcare center than the former ones. The experiments performed are aimed 1) to prove the suitability of the linear operator as part of a classification chain that is intended to work with hyperspectral video. Therefore, 2) the accelerated algorithm must be able to achieve real-time performance not to introduce a bottleneck into the processing chain.

The experiments were conducted on two platforms: The first one, a CPU based platform consisting of a 10th generation Intel core i5-10400F working at its regular frequency of 2.9 GHz with 32 GB of DDR4 RAM. On this platform the algorithm was executed on its Python implementation with no parallelization. The second platform is a GPU (Nvidia RTX 3080) with 12GB of GDDR6X memory, 8960 CUDA cores, 384 bits of memory bus and Ampere architecture. In this case, the code executed was accelerated according to the Sect. 3.2.

### 4.1 Objective Results

The results presented on Table 2 show the mean percentage of vascular and tumor samples from the ground-truth detected in the segmented area and the average time in milliseconds the operator took on both platforms to generate the segmentation mask. All the results were averaged for the 5 hyperspectral images testing the 3 sets of parameters detailed in Table 1. Each one of the combinations offers a trade-off between a gain in the percentage of blood vessel

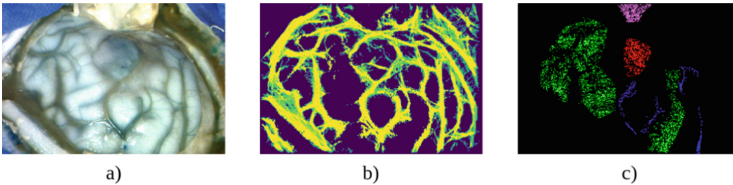


samples segmented and an increase of the tumor pixels included in the segmented mask. The line detector proposed by Ricci et al. is only implemented on CPU with a window size set to 15, as described in [7]. Band 16 is selected and the chosen gray level threshold is 266. The band and the gray threshold are taken from the set of parameters A because, as it will be seen in Subsect. 4.2, it is the combination that yields the best results. To fully evaluate the implications of the percentages shown in Table 2, its interpretation must be supported by the corresponding subjective results depicted in Fig. 5.

Figure 4 shows the synthetic RGB image extracted from a hyperspectral cube, an example of a segmented mask processed from that image and the sparse ground-truth from which the metrics of Table 2 are calculated. In the ground-truth image, black pixels refer to unlabeled samples, the green samples correspond to healthy tissue, the pink ones indicate the dura mater samples and the red and the blue pixels are used for tumor and vascular samples, respectively. The number of vein and artery samples labeled is remarkably low, especially for training any of the supervised ML algorithms mentioned in Sect. 1. Since none of them is designed to work with such sparse and scant ground truth, the inclusion of their results would make an unfair comparison with the proposed algorithm.

**Table 2.** Percentage of selected tissue by each combination of parameters and their computation time.

Method	Avg. Vessel	Avg. Tumor	Avg. CPU (ms)	Avg. GPU (ms)
Ricci et al.	0.47	0.02	$2.00 \cdot 10^3$	–
A	0.68	0.07	$4.94 \cdot 10^3$	5.44
B	0.72	0.18	$4.39 \cdot 10^3$	8.38
C	0.75	0.19	$4.95 \cdot 10^3$	8.83

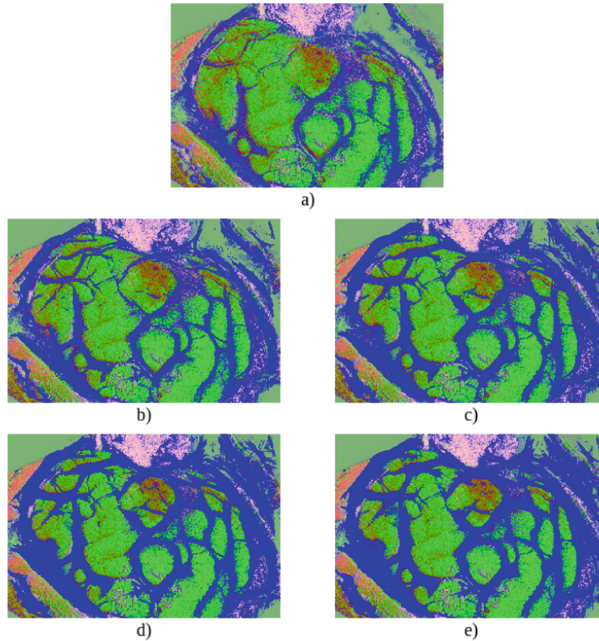


**Fig. 4.** (a) RGB image extracted from the hyperspectral cube, (b) Example of the strength levels in the segmentation mask for the set of parameters B, (c) Ground-truth image

The numerical results show an average speed up of 664 times of the GPU implementation over the CPU execution. Thus, achieving a performance, for the worst case, over 200 frames per second, which guarantees a real-time classification.

## 4.2 Subjective Results

Figure 5 shows the comparison between the original classification map and the result of overlapping on it the colorized segmented mask obtained for each one of the 3 parameters tested. Since the segmented mask is not binary, those pixels where its values are not at their maximum, are proportionally combined with the color information of the classification map.



**Fig. 5.** (a) Original classification map, (b) masked classification map with linear detector from Ricci et al., (c) masked classification map with set of parameters A, (d) masked classification map with set of parameters B and (e) masked classification map with set of parameters C. (Color figure online)

In Fig. 5 it can be seen how the linear operator [7] improves the original classification map, but it is the method that leaves the most tumor false positives uncovered. The set of parameters A, despite having the lowest percentage of vessel samples segmented 2 among the three combinations tested, is able to mark the contours of most of the arteries and veins in a very similar way as the rest of the sets. Parameters B and C provide thicker contours around the majority of arteries and veins but do not increase the sensitivity for thinner vessels. This characteristic may imply that vessels surrounding tumor tissue are more prone to extend its limits inside the tumor area, increasing the false negative rate when detecting the tumor. Therefore, set of parameters A has proven to be the most suitable for the improvement of the classification maps, showing the robustness

acquired through the optimized selection of parameters and the use of different window sizes.

## 5 Conclusions and Future Work

In this work, a brain blood vessel segmentation algorithm for hyperspectral images captured during in-vivo tumor resection surgeries based on linear operators has been presented. Its capability to detect blood vessels without including samples from other kinds of tissues has been measured through objective metrics. These very metrics played a fundamental role on the parameter setting by serving as variables to be optimized. Through this optimization process it was possible to explore different candidate combinations to end up selecting the most convenient set according to the subjective observations when overlapping the segmented mask on the classification map. The combination of the algorithm with the map proved to be helpful in correcting the tumor false positives issue the classification suffers from, particularly on the contours of the blood vessels. Moreover, the proposed solution has been optimized in terms of computing performance by porting the source code to a GPU architecture. Thanks to this extent, the processing chain remains into the real-time processing constraint, i.e. 200 frames per second.

In the future, it will be explored refinement techniques for achieving smoother and better connected contours on the segmented mask. Also, current results leave room for improvement in the detection of thinner vessels and lighter contours that will be studied through filtering techniques that make use of the GPU acceleration.

## References

1. Lu, G., Fei, B.: Medical hyperspectral imaging: a review. *J. Biomed. Opt.* **19**(1), 010901 (2014). <https://doi.org/10.1117/1.JBO.19.1.010901>
2. Leon, R., et al.: Hyperspectral imaging for in-vivo/ex-vivo tissue analysis of human brain cancer. In: Linte, C.A., Siewerdsen, J.H. (eds.) *Medical Imaging 2022: Image-Guided Procedures, Robotic Interventions, and Modeling*, vol. 12034, p. 1203429. International Society for Optics and Photonics, SPIE (2022). <https://doi.org/10.1117/12.2611420>
3. Urbanos, G., et al.: Supervised machine learning methods and hyperspectral imaging techniques jointly applied for brain cancer classification. *Sensors* **21**(11) (2021). [www.mdpi.com/1424-8220/21/11/3827](http://www.mdpi.com/1424-8220/21/11/3827)
4. Goni, M.R., Ruhaiyem, N.I.R., Mustapha, M., Achuthan, A., Che Mohd Nassir, C.M.N.: Brain vessel segmentation using deep learning-a review. *IEEE Access* **10**, 111322–111336 (2022)
5. Nazir, A., et al.: OFF-eNET: an optimally fused fully end-to-end network for automatic dense volumetric 3d intracranial blood vessels segmentation. *IEEE Trans. Image Process.* **29**, 7192–7202 (2020)
6. Babin, D., Pizurica, A., De Vylder, J., Vansteenkiste, E., Philips, W.: Brain blood vessel segmentation using line-shaped profiles. *Phys. Med. Biol.* **58**, 8041–8061 (2013)

7. Ricci, E., Perfetti, R.: Retinal blood vessel segmentation using line operators and support vector classification. *IEEE Trans. Med. Imaging* **26**(10), 1357–1365 (2007)
8. Wu, Y., et al.: Blood vessel segmentation from low-contrast and wide-field optical microscopic images of cranial window by attention-gate-based network. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pp. 1864–1873 (2022)
9. Fabelo, H., et al.: Deep learning-based framework for in vivo identification of glioblastoma tumor using hyperspectral images of human brain. *Sensors* **19**(4) (2019). [www.mdpi.com/1424-8220/19/4/920](http://www.mdpi.com/1424-8220/19/4/920)
10. Ronneberger, O., Fischer, P., Brox, T.: U-net: convolutional networks for biomedical image segmentation (2015). [arxiv.org/abs/1505.04597](https://arxiv.org/abs/1505.04597)
11. Li, J., Udupa, J., Tong, Y., Wang, L., Torigian, D.: Segmentation evaluation with sparse ground truth data: simulating true segmentations as perfect/imperfect as those generated by humans. *Med. Image Anal.* **69**, 101980 (2021)
12. Villa, M., et al.: Data-type assessment for real-time hyperspectral classification in medical imaging. In: Desnos, K., Pertuz, S. (eds.) *Design and Architecture for Signal and Image Processing*, pp. 123–135. Springer International Publishing (2022). [https://doi.org/10.1007/978-3-031-12748-9\\_10](https://doi.org/10.1007/978-3-031-12748-9_10)
13. Staal, J., Abramoff, M., Niemeijer, M., Viergever, M., van Ginneken, B.: Ridge-based vessel segmentation in color images of the retina. *IEEE Trans. Med. Imaging* **23**(4), 501–509 (2004)
14. Niemeijer, M., Staal, J., van Ginneken, B., Loog, M., Abramoff, M.D.: Comparative study of retinal vessel segmentation methods on a new publicly available database. In: *SPIE Medical Imaging* (2004)
15. Eklund, A., Dufort, P., Forsberg, D., LaConte, S.M.: Medical image processing on the GPU - Past, present and future. *Med. Image Anal.* **17**(8), 1073–1094 (2013). [www.sciencedirect.com/science/article/pii/S1361841513000820](http://www.sciencedirect.com/science/article/pii/S1361841513000820)
16. Wang, X., Shi, B.E.: Gpu implementation of fast Gabor filters. In: *Proceedings of 2010 IEEE International Symposium on Circuits and Systems*, pp. 373–376 (2010)
17. Akiba, T., Sano, S., Yanase, T., Ohta, T., Koyama, M.: Optuna: a next-generation hyperparameter optimization framework (2019). [arxiv.org/abs/1907.10902](https://arxiv.org/abs/1907.10902)
18. Martin-Perez, A.: Hyperparameter optimization for brain tumor classification with hyperspectral images. In: *5th Euromicro Conference on Digital System Design (DSD)* (2022)
19. Srinivas, P., Katarya, R.: hyoptxg: optuna hyper-parameter optimization framework for predicting cardiovascular disease using xgboost. *Biomed. Signal Process. Control* **73**, 103456 (2022). [www.sciencedirect.com/science/article/pii/S1746809421010533](http://www.sciencedirect.com/science/article/pii/S1746809421010533)