Algorithm for Optimization in Medical Image Processing applied in Heterogeneous Architecture

Wilver Auccahuasi [1], Sandra Meza [2], Emelyn Porras [3], Milagros Reyes [4], Oscar Linares [5], Karin Rojas [6], Miryam Inciso-Rojas [7], Tamara Pando-Ezcurra [8], Gabriel Aiquipa [9], Yoni Nicolas-Rojas [10], and Aly Auccahuasi [11]

[1, 3, 4] *Universidad Científica del Sur, Lima, Perú*

[2] *Universidad ESAN, Lima, Perú*

[5] *Universidad Continental, Huancayo, Perú*

[6] *Universidad Tecnológica del Perú, Lima, Perú*

[7] *Universidad Privada del Norte, Lima, Perú*

[8] *Universidad privada Peruano Alemana, Lima, Perú*

[9] *Universidad Tecnológica de los Andes, Apurímac, Perú*

[10] *Escuela Superior la Pontificia, Ayacucho, Perú*

[11] *Universidad de Ingeniería y Tecnología, Lima, Perú*

Abstract

In these times of pandemic, hospitals are being the focus of many innovations, not only for the adaptation to telemedicine, but also from the perspective of the use and processing of the multiple modalities of medical images, where we find images made up of a single Image such as x-rays, images that are made up of a sequence of images such as tomography and Magnetic Resonance, or in video format as is the case with ultrasound and angiography. One way of working with images is through popular image servers that connect to medical equipment for transfer and storage. In the process of visualization and processing, special workstations with good computational capacity are required for these purposes, in most cases these workstations are connected in the network of medical offices, therefore they are presented in a normal working image display requests at the same time. The methodology presented uses a heterogeneous architecture based on CPU and GPU, in such a way that by means of an algorithm it analyzes the type and dimension of the image to be able to choose where the processing will be carried out, thereby optimizing the use of computational resources. and we can achieve a parallel job that the CPU and GPU are working simultaneously with different imaging modalities. As a result, we present the execution mode of the algorithm where it automatically chooses what type of image is processed by the CPU and what type is processed in the GPU, as well as the execution time in each of them. Finally we can indicate that the algorithm can be scalable towards workstations to optimize its use in clinical practice.

Keywords

Programming, GPU, medical imagining, algorithms, methodology

1. Introduction

Medical images are one of the most used techniques in diagnosis and medical research, therefore its use is increasing considerably, for this, multiple medical images of different modalities are

ACM-2022: Algorithms Computing and Mathematics Conference, August 29 – 30, 2022, Chennai, India.
EMAIL: wilver.auccahuasi@upn.edu.pe (Wilver Auccahuasi)
ORCID: 0000-0001-8820-4013 (Wilver Auccahuasi)

analyzed each time, these are processed on different platforms, both on personal computers, as in workstations where they are connected to image servers, for simultaneous processing. Carrying out a review of the state of the art, we found work related to processing directly in graphic processing units better known as (GPU), where large images such as satellite images are worked in various processes, such as analysis of the characteristics chromatic, parallel processing, demonstrating that through the use of GPUs, processing times can be reduced [1] [2] [3] [4].

One of the current uses in the processing of medical images is related to deep learning considered in medical segmentation, with a manual design of a neural network applied to segmentation with a long time of training with large volumes of data, for which proposes the search for multiobjective neural architecture (NAS) with which the design of precise and efficient segmentation architectures can be automated, for which we present EMONAS-Net within the framework of multiobjective NAS used for the segmentation of medical images in 3D where segmentation has been specified according to the size of the network, it has 2 important components where a configuration of the micro and macro structure of the architecture is considered together with an algorithm that is based on multiobjective evolution assisted by substitutes seeking to improve the hyperparameter values, where this SaMEA algorithm uses a collection that is collected when in The beginning of the evolutionary process in which to identify the subproblems of the hyperparameter values, improving the performance during the mutation, which improves the speed of convergence, the Random Forest surrogate study model is also incorporated, which accelerates the evaluation of the aptitude about the architecture, however EMONAS-Net was tested in the prostate segmentation of the MICCAI PROMISE challenge12, hippocampal segmentation of the Medical Segmentation Decathlon challenge and cardiac segmentation of the MICCAI ACDC challenge, where the proposed framework had favorable results compared to the NAS methods. since they are smaller and reduce search time by 50% [5].

In artificial intelligence applications, we analyze deep neural architecture, which have limitations, therefore an adaptive neural architecture optimization model (ANAO) is proposed to optimize the structure of the convolutional neural network (CNN) that is based on neural blocks, it has an integer propagation for the optimization process in order to maximize the precision of the designed model and the speed of convergence, with restrictions which restrict the design requirements of CNN for which a function is proposed. which has consideration about the precision and the convergence trend of the training, the neural network is applied to evaluate the performance of the models in the increase on the efficiency of the optimization process, through the heuristic process to perform the optimization with the ANAO model is applied to the diagnosis of retinal disorders have been performed 8 CNN which were compared with the ANAO model from the perspective of precision and convergence trend, pressing very high performance results which can be adapted to CNN architectures for data sets [6].

One of the methods used in the processing of medical images, semi-supervised classification and segmentation methods are widely investigated in the analysis of medical images, however, both approaches can improve the performance of fully supervised methods with additional unlabeled data, for which he proposes a semi-supervised Medical Image Detector (SSMD) method, having a motivation behind SSMD which is to provide free supervision resulting in effective unlabeled data, regularizing predictions that are consistent, for what has been developed an adaptive coherence cost function regularizing the different components in the predictions, incorporating heterogeneous perturbation strategies which work for feature spaces such as spaces with images so that the detector can produce images and solid predictions, we obtained experimental results ex SSMD strains which have performed in a wide range of environments, demonstrating the strength of the modules [7].

Various classifiers and fusion architecture with 2 critical characteristics are also analyzed, where the learning method for homogeneous and heterogeneous sets stands out, considered to build a successful multiple classifier system, which is why a 2-level method is presented in hierarchical fusion of homogeneous multiclassifiers. and heterogeneous (HF2HM), where the classification models produced for the heterogeneous classifiers will be integrated with homogeneous training data sets projected at random, considering a valid hierarchical fusion scheme using public ICU data sets and three clinical data sets. , demonstrating the superiority of the HF2HM framework over the other base classifiers, being a potential tool for medical decision making [8].

One of the most used techniques is convolutional neural networks (CNN) considered with a computer vision technique where the classification of images is carried out, so a new approach is presented to train CNN by generalizing the heterogeneous data sets that are originate from several sources and without local annotations, channeling the data analysis with the Gleason classification on prostate images where 2 sequence models have been included with sequences of teacher / student training paradigms, the teacher model will annotate the automatic a set of pseudo-labeled patches used to train the student model, then these 2 models are trained with 2 different approaches such as semi-supervised learning and semi-weekly supervised learning, where each approach will present 2 training variants of students, having as a baseline a training of the student model only with data They are strongly annotated, where the performance on the classification is evaluated with the student model at the patch level, global level, allowing both models to be generalized despite the heterogeneity between data sets and the small amount of local annotations used, the performance of the classification was improved at the patch level (up to $\kappa = 0.6127 \pm 0.0133$ of $\kappa = 0.5667 \pm 0.0285$), at the basic level of TMA (Gleason score) (up to $\kappa = 0.7645 \pm 0.0231$ of $\kappa = 0.7186 \pm 0.0306$) and at the WSI level (Gleason score) (up to $\kappa = 0.4529 \pm 0.0512$ of $\kappa = 0.2293 \pm 0.1350$) from which the results of the teacher / student paradigm can be shown, it can be trained by generalizing data sets from different sources despite their heterogeneity [9] [10].

After having presented the state of the art, where the use of GPUs is being used more frequently, which indicates that this hardware tool is very important to be able to configure the different techniques of neural networks, configure convolutional networks among other techniques that provides us with Artificial Intelligence, our proposal consists of being able to present an algorithm that, faced with a need to be able to process images of different types and modalities, simultaneously, for which it analyzes the type of image through its dimensions and sends the image to be processed in the CPU when the image is small and sends to the GPU when the image is very large or has a set of images, the results show that the proposed methodology is applicable to various solutions where there is a load of images and the need to speed up the processing, we present how to implement the solution.

2. Materials and Methods

In this section, we present the methodological proposal, where it is presented in four steps, starting from the description of the problem, going through the analysis of medical images, then we make a description of the architecture that is available and with which we will carry out the tests of the methodology and finally with the implementation where we present the implemented code.

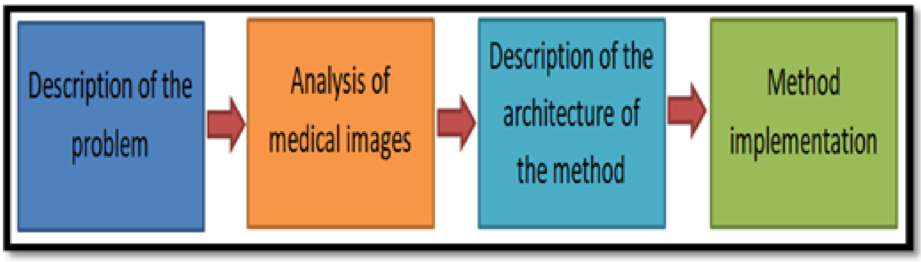

Description of the problem
Analysis of medical images
Description of the architecture of the method
Method implementation

Figure 1: Block Diagram of the Proposal

2.1. Description of the problem

The problematic description, we can describe it from two perspectives, the first from the increasingly used and necessary diagnostic power through the analysis of medical images, we can indicate that many of the pathologies are resolved by analyzing the medical images, from the analysis of the images of X-rays, for pulmonology problems, traumatology, such as computed tomography images to evaluate brain problems such as aneurysms, magnetic resonance images, for the analysis of muscle lesions, mammography images, to evaluate the condition of the breasts and stages of possible cases of cancer, fetal ultrasound images, for the analysis of the status of the fetuses in the mother's womb to assess their growth status, angiography images, where heart operations are analyzed, among other modalities.

The second refers to the hardware resource that is available, in hospitals, we find workstations connected to medical equipment to be able to view the images, which have a CPU architecture in most cases, then we have one workstations that have a CPU configuration as a heterogeneous architecture based on CPU + GPU, these workstations are connected to the PACS servers where most of the medical images are accessed, these workstations have a important workload, because it receives requests to send and visualize the images from the different users that would be the clinics and emergency centers, these stations must request the image from the PACS server, visualize and carry out the different processes, in these cases where the workstation has a strong need for processing, it is necessary to have a mechanism to power r optimizing the use of existing architecture.

The methodology presented is based on the use of these heterogeneous architectures, where the use of the CPU and the GPU is proposed individually and simultaneously when required, the decision to choose where it is processed depends on the algorithm proposed, who analyzes the dimensions of the images and sends it to be processed.

2.2. Analysis of medical images

The analysis of medical images is related to being able to describe the types of medical images, the image format, as well as how they are formed and the size of the image, this information is important, because depending on the characteristic that the images present, the algorithm will be able to decide if it is processed in the CPU or in the GPU, below is a description of each modality of medical imaging.

Table 1: Description of medical images

Image modality	Image format	Amount of the image that make up the image	Image size inMB
X-rays	Image	1	5 MB
Mammography	Image	4	15.5 MB
Ultrasound	Video	Many	1.91 GB
Angiography	Video	Many	90 MB
Computed tomography	Image	Many	90 MB
Magnetic resonance	Image	Many	200 MB

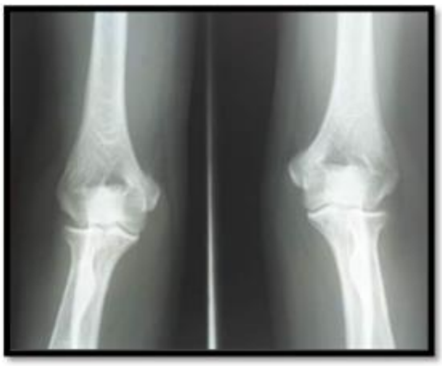


Figure 2: X-ray image

Table 1, we present a table with the description of the images that will be subjected to the algorithm tests, in which each of the characteristics of each modality is indicated, which is described below:

In figure 2, an x-ray image is presented, which has a fundamental characteristic, that a study of this modality is made up of a single image, this image modality is widely used in the analysis of bone tissue, therefore its use is very continuous.

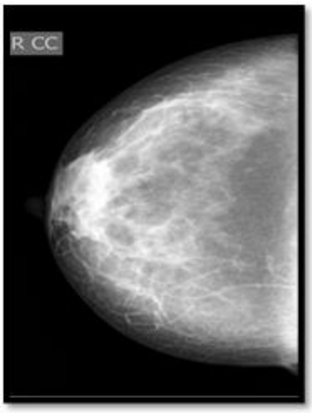


Figure 3: Mammogram image

In figure 3, a mammography image is presented, this modality is used in breast cancer studies, a study of this modality is made up of 4 images, made up of two images for each breast, the image has a larger size in Compared with the x-ray image for the level of detail, to analyze the breast tissue, normally the applications of this modality refer to being able to visualize the four images simultaneously.

In figure 4, the ultrasound modality is presented, in the use of fetal gynecology, this type of modality has the characteristic that in its formation, it is composed of a sequence of images that make up a video, the result at the end of the study allows view sequentially to assess the status of the fetus, mainly age and morphological manifestations. The final weight of the file is related to the study time.

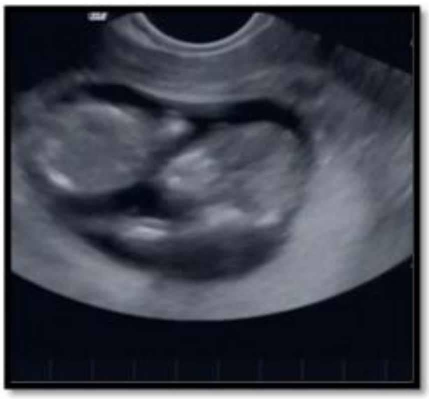


Figure 4: Ultrasound image

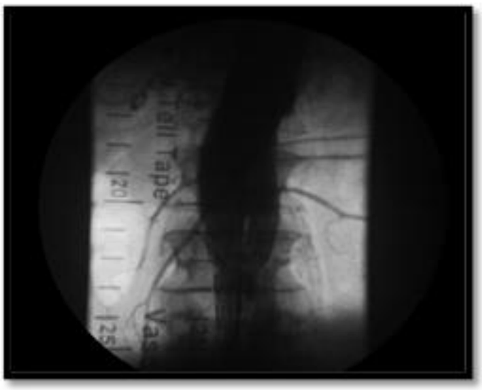


Figure 5: Angiography image

Figure 5 presents an angiography study, like the ultrasound modality, the final image is composed of a video, where the blood flow in the vessels that make up the arterial tree of the heart is shown, the weight of the image is composed of the time the study takes, the longer the study lasts, the larger the file will be.

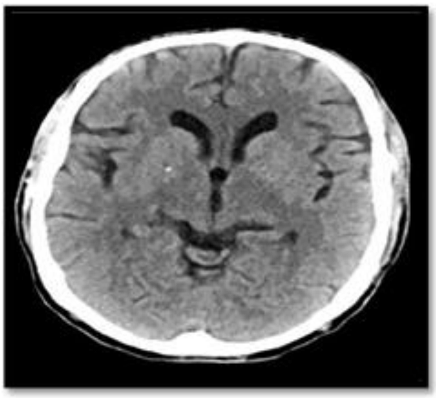


Figure 6: Computed tomography image

Figure 6. It refers to the modality of computed tomography, its characteristic is that the study is made up of a sequence of images that corresponds to each slice of the image, in a regular exam, it consists of 300 images taken sequentially, the size of the file will depend on the number of sequences and the organ under study.

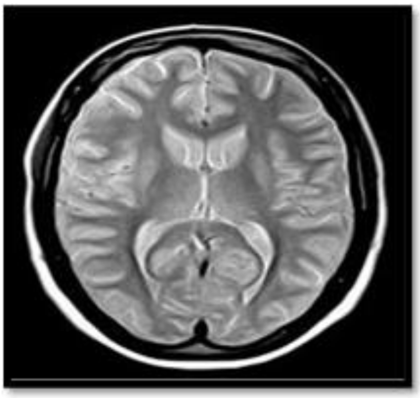


Figure 7: Magnetic resonance imaging

Finally, we present in figure 7, the magnetic resonance examination, where, like the computed tomography modality, the study is made up of a sequence of images, the examination weight, corresponds to the number of images, which is analyzed, when requires analyzing these images, the entire sequence of images is loaded into memory, which makes it heavier and requires greater computational resources.

2.3. Description of the architecture of the method

The architecture that we have available, for the algorithm demonstration, is composed of an I7 CPU with 32GB of internal memory and a GTX 1050ti model GPU with 768 Cuda cores with 4 GB of dedicated memory, we can indicate that the way to work with the GPU In any type of application, the data is first loaded into the system memory and then sent to the GPU, in this working mode it is considered a transfer time, which is the time it takes to pass the data from the system memory to GPU memory. In the present proposal an algorithm is worked on when sending an image to be processed. The algorithm decides where it will be processed on the CPU or on the GPU.

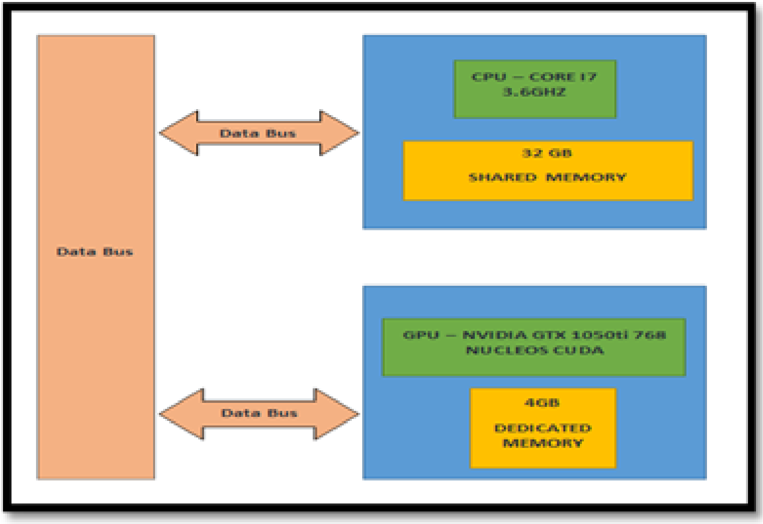


Figure 8: Architecture diagram, where you can see the way of communication distinguishes communication with the CPU and GPU

2.4. Method Implementation

For the implementation method, we resort to the use of the MATAB computational tool, which allows us to work directly with the hardware we have, through this tool we can program to load and exchange information between the system memory and the GPU memory. , as an example of implementation we make two files in MATLAB, where we create a function that analyzes and decides where it will be processed and a file where we present lines of code to load the image and a basic process, which constitutes in making the negative of the image, the intention of presenting the algorithm is to verify who performs the processing, the CPU or the GPU. Here is the flow chart of the algorithm.

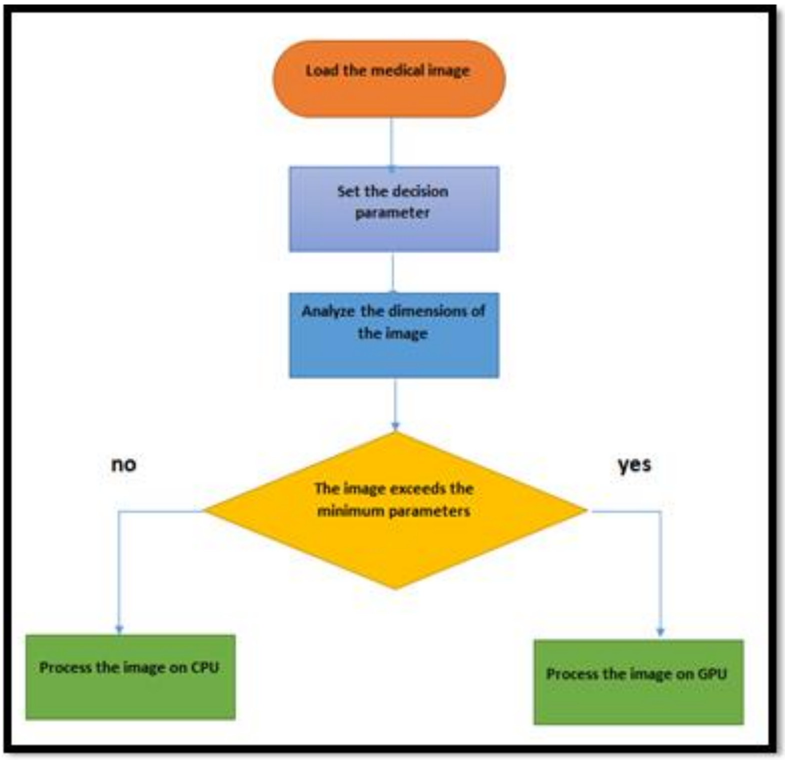
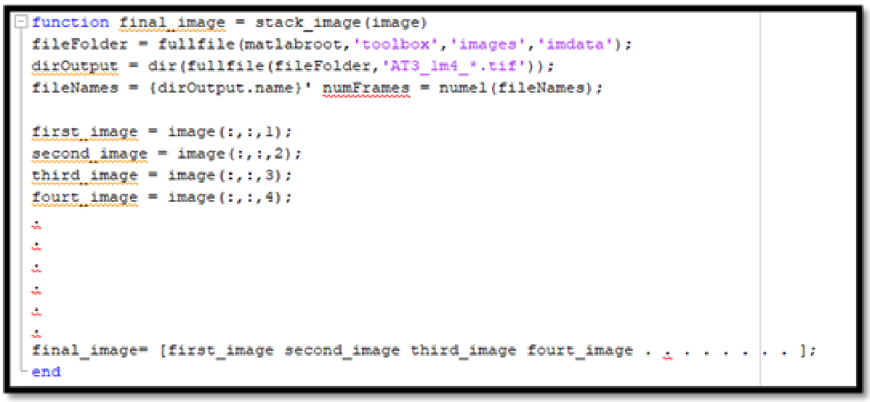

Load the medical image
Set the decision parameter
Analyze the dimensions of the image
no
yes
The image exceeds the minimum parameters
Process the image on CPU
Process the image on GPU

Figure 9: Algorithm Flowchart

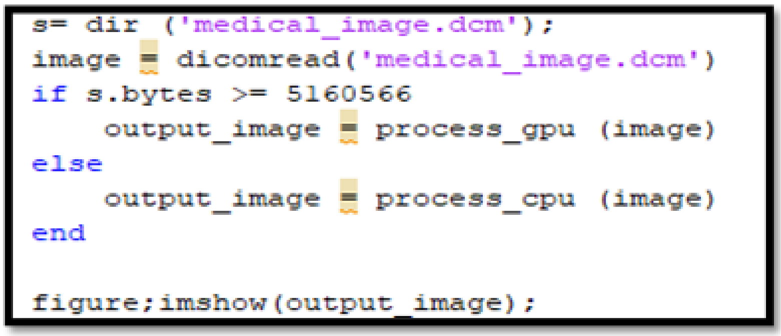


```
function final_image = stack_image(image)
fileFolder = fullfile(matlabroot,'toolbox','images','imdata');
dirOutput = dir(fullfile(fileFolder,'AT3_1m4_*.tif'));
fileNames = {dirOutput.name}' numFrames = numel(fileNames);

first_image = image(:,:,1);
second_image = image(:,:,2);
third_image = image(:,:,3);
fourt_image = image(:,:,4);
.
.
.
.
.
.
final_image= [first_image second_image third_image fourt_image . . . . . . . . ];
end
```

Figure 10: Function to read several images

In figure 10, a function is presented to be able to create an image from a sequence of images, for which one image has to be aligned followed by the other and so on, this technique must be implemented for magnetic resonance images and computed tomography, in order to be able to calculate the total study size.

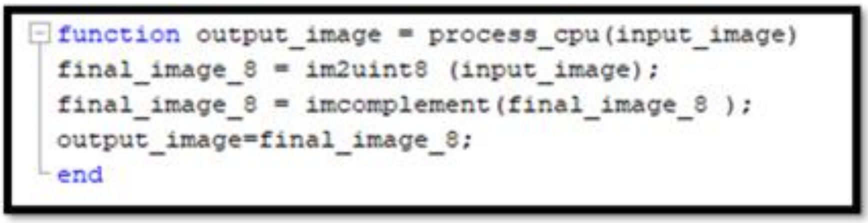


```
s= dir ('medical_image.dcm');
image = dicomread('medical_image.dcm')
if s.bytes >= 5160566
    output_image = process_gpu (image)
else
    output_image = process_cpu (image)
end

figure;imshow(output_image);
```

Figure 11: Function to read several images

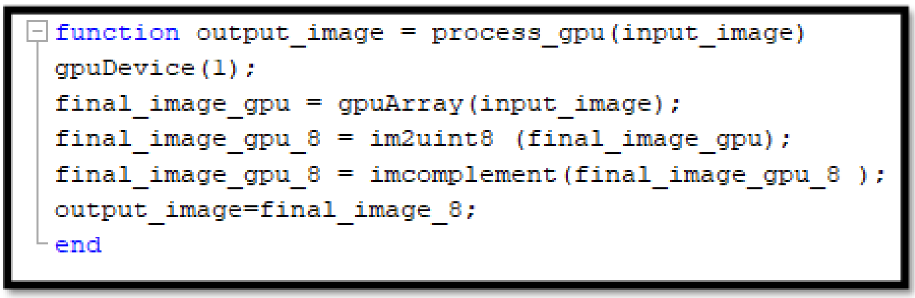
Main code for the execution of the algorithm, where the main method to obtain the weight of the medical image is appreciated, in our case, as we are working with images that are being analyzed directly from the medical equipment, they are in their original format, DICOM format (* .DCM), in this way it can be implemented in clinical services. The weight of the file obtained is evaluated through an "if" statement where it is asked if the file has a weight greater than 5 MB, in our particular case, this size that was used as a reference, is taken from a mammography image of high density, so normally the X-ray images are oriented to be analyzed in the CPU and the others in the GPU, due to the size of the files that make up the medical image. As can be seen in figure 11.



```
function output_image = process_cpu(input_image)
final_image_8 = im2uint8 (input_image);
final_image_8 = imcomplement(final_image_8 );
output_image=final_image_8;
end
```

Figure 12: Function for medical image processing in CPU

When the image is smaller than the reference image, a call is made to the GPU processed function, which allows working with the image that is in the system memory, the function receives the image as input reference and returns a resulting image after applying proper processing. Figure 12 shows the code for this function, it can be adapted and easily implemented.



```
function output_image = process_gpu(input_image)
gpuDevice(1);
final_image_gpu = gpuArray(input_image);
final_image_gpu_8 = im2uint8 (final_image_gpu);
final_image_gpu_8 = imcomplement(final_image_gpu_8 );
output_image=final_image_8;
end
```

Figure 13: Function for medical image processing in GPU

When the image has a value greater than the reference weight, the process is carried out on the GPU; For which figure 13 presents the detail of the function that receives the medical image as an input parameter, carries out the transfer of information to the GPU memory, where the defined procedures are carried out, the function returns the output image that will be sent to the main function for viewing, the code mentioned in the function, can be easily scaled and adapted.

3. Results

The results that are presented at the end of the demonstration of the proposal is characterized, in the evaluation of a group of images formed by X-ray images, digital mammography, fetal ultrasound, angiography, computed tomography and magnetic resonance, where they were tested simultaneously using parallel tasks and evidencing that in each of the cases the call to the processing function was made, both the one executed on the CPU and the one executed on the GPU; As can be seen in table 2. Where the type of medical imaging modality is evidenced, the size of the file that contains the image and the function used for processing.

The results show that the use of the hardware available in health establishments can be optimized, when even more, there is no budget to carry out the acquisition of modern equipment with greater computational capacity, the results demonstrate the practicality of the implementation and the easy handling of the images, with the functionality that these calls to the functions, can be sequentially, which can be called several times and from several users, giving the feeling of working in parallel.

Table 2: Processed images with their respective size and function used for processing

Image modality	Image size in MB	Read function used
X-rays	5 MB	Process_cpu
Mammography	15.5 MB	Process_gpu
Ultrasound	1.91 GB	Process_gpu
Angiography	90 MB	Process_gpu
Computed tomography	90 MB	Process_gpu
Magnetic resonance	200 MB	Process_gpu

The results also show that the MATLAB computational tool allows the design of solutions for different uses according to the hardware that is available, in this case it is important to indicate that it was worked with a NVIDIA brand GPU due to the compatibility and the libraries that make I practice the use and integration with CPU-based systems.

4. Conclusion

The conclusion that are reached at the end of the investigation, is characterized that the algorithm that is proposed achieves the results that were planned at the beginning of the investigation, having a workstation with heterogeneous architecture is one of the hardware solutions that must be exploit to the maximum, due to the high computational capacity of the graphics processors (GPU), not only in the display process, but also in the calculation process, in most cases when applications are installed for reading and processing any type of images, most work with the CPU to perform the calculation, using the GPU only for display like a video card.

Being able to configure the GPU as a calculation unit, considerably frees the work of the CPU, in our case, the heaviest images are processed by the GPU and the less heavy and therefore are the ones that are used the most are worked on the CPU. One of the characteristics that we must take into account and it is what has been to indicate that not every image is processed faster in the GPU, because there is a transfer time of the image from the system memory to the memory of the GPU, where in many cases when the image is very small, it takes longer to process it on the GPU, which is suggested to process it on the CPU, and only when the image deserves the use of the GPU, it can be processed on the GPU, This is one of the criteria for the design of the algorithm, being able to choose this decision manually can be complex, when it is unknown how it is possible to send the process to the GPU, in this situation the algorithm proves to be helpful because it automatically starts assigning processing tasks to the CPU and GPU.

Finally, we can indicate that the algorithm can be scaled and implemented without a problem and in any programming language and in any type of hardware that is counted from the most sophisticated, to the new embedded computers, you just have to make sure that it has a CPU and a GPU in its architecture.

5. References

[1] Auccahuasi, W., Sernaque, F., Flores, E., Garzon, A., Barrutia, A., & Oré, E. (2020). Analysis of the chromatic characteristics, on land cover types using synthetic aperture images. Procedia Computer Science, 167, 2524-2533.

[2] Auccahuasi, W., Bernardo, M., Núñez, E. O., Sernaque, F., Castro, P., & Raymundo, L. (2018, December). Analysis of chromatic characteristics, in satellite images for the classification of vegetation covers and deforested areas. In Proceedings of the 2018 the 2nd International Conference on Video and Image Processing (pp. 134-139).

[3] Aiquipa, W. A., del Carpio, J., Garcia, J., Benites, R., Grados, J., & Flores, E. (2019, October). Analysis of High Resolution Panchromatic Satellite Images, Based on GPGPU Programming. In Proceedings of the 2019 2nd International Conference on Sensors, Signal and Image Processing (pp. 45-48).

[4] Auccahuasi, W., Castro, P., Flores, E., Sernaque, F., Garzon, A., & Oré, E. (2020). Processing of fused optical satellite images through parallel processing techniques in multi GPU. Procedia Computer Science, 167, 2545-2553.

[5] Baldeon Calisto, M., & Lai-Yuen, S. K. (2021). EMONAS-Net: Efficient multiobjective neural architecture search using surrogate-assisted evolutionary algorithm for 3D medical image segmentation. Artificial Intelligence in Medicine, 119.https://doi.org/10.1016/j.artmed.2021.102154

[6] Wang, H., Won, D., & Yoon, S. W. (2021). An adaptive neural architecture optimization model for retinal disorder diagnosis on 3D medical images. Applied Soft Computing, 111. https://doi.org/10.1016/j.asoc.2021.10768Zhou, H. Y., Wang, C., Li, H., Wang, G., Zhang, S., Li, W., & Yu, Y. (2021). SSMD: Semi-Supervised medical image detection with adaptive consistency and heterogeneous perturbation. Medical Image Analysis, 72. https://doi.org/10.1016/j.media.2021.102117

[7] Wang, L., Mo, T., Wang, X., Chen, W., He, Q., Li, X., … Zhen, X. (2021). A hierarchical fusion framework to integrate homogeneous and heterogeneous classifiers for medical decision-making. Knowledge-Based Systems, 212. https://doi.org/10.1016/j.knosys.2020.106517

[8] Marini, N., Otálora, S., Müller, H., & Atzori, M. (2021). Semi-supervised training of deep convolutional neural networks with heterogeneous data and few local annotations: An experiment on prostate histopathology image classification. Medical Image Analysis, 73. https://doi.org/10.1016/j.media.2021.102165

[9] Chanchal, A. K., Kumar, A., Lal, S., & Kini, J. (2021). Efficient and robust deep learning architecture for segmentation of kidney and breast histopathology images. Computers and Electrical Engineering, 92. https://doi.org/10.1016/j.compeleceng.2021.107177

[10] Hussain, T. (2017). ViPS: A novel visual processing system architecture for medical imaging. Biomedical Signal Processing and Control, 38, 293–301.https://doi.org/10.1016/j.bspc.2017.06.003