

FPGA Implementation for Fast and Accurate Medical Image Segmentation

Fatma Zohra Hamidatou-Hamadi, Mohamed Lamine Hamidatou and Latifa Hamami-Mitiche

Abstract—This work presents a comprehensive study and hardware implementation of two advanced medical image segmentation methods: Gradient Vector Flow and Distance Regularized Level Set Evolution. These approaches are widely used in image processing due to their ability to accurately extract complex contours, particularly in concave and noisy regions. However, their high algorithmic complexity leads to significant computational cost, limiting their use in real-time applications. To address these limitations, implementation on a Field-Programmable Gate Array (FPGA) is proposed, leveraging hardware parallelism to enhance performance. A simulation phase was conducted using MATLAB on real medical images (brain and breast MRI scans). The results demonstrate that the GVF method is effective in handling concave boundaries, while the DRLSE method exhibits greater robustness to topological changes and multiple objects. The hardware implementation was carried out using Vitis HLS and Vivado on a PYNQ-Z2 platform. The results show strong agreement with MATLAB simulations, with negative discrepancies due to numerical precision, along with a significant improvement in processing time.

Keywords—DRLSE, FPGA, GVF, Level Set.

NOMENCLATURE

DRLSE	Distance Regularized Level Set Evolution
FPGA	Field-Programmable Gate Array
GVF	Gradient Vector Flow
LUT	Look-Up Table
MRI	real medical images

I. INTRODUCTION

Medical image segmentation is a fundamental step in many clinical applications, including diagnosis, tumor detection, and therapeutic monitoring [1]. It involves extracting structures of interest from medical images, such as MRIs or CT scans. Among the most widely used methods, active contours and Level Set methods hold an important place due to their ability to model complex shapes. However, these methods have certain limitations. Traditional active contours are highly dependent on initialization and have difficulty converging in concave areas [2]. To improve this performance, the GVF method was introduced, allowing better capture of concavities [3]. Level Set methods, while efficient at handling topological changes, require frequent resetting, which increases computation time. The DRLSE model has been proposed to eliminate this constraint by incorporating a regularization term [4].

Despite their performance, these methods remain computationally expensive. Therefore, FPGA implementation

Manuscript received April 9, 2026; accepted June 28, 2026.

Fatma Zohra Hamidatou-Hamadi, Mohamed Lamine Hamidatou and Latifa Hamami-Mitiche are with the Ecole Nationale Polytechnique, Algiers, ALGERIA.

Digital Object Identifier (DOI): 10.53907/enpesj.v6i1.367

offers a promising solution for accelerating their execution.

II. ACTIVE CONTOURS AND GVF

1. ACTIVE CONTOURS

Active contours are evolving curves defined by a parametric function (Figure 1). Their evolution is controlled by minimizing a total energy composed of an internal energy and an external energy [5].

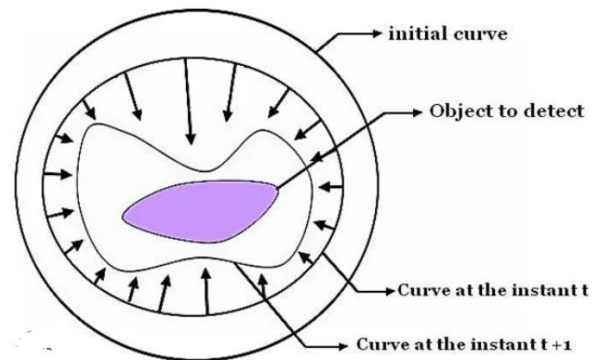


Fig. 1: Representation of the evolution of the active contour.

The outline C defined by a parametric representation [5], the active contour assimilated has a curve C represented according to equation (1):

$$C = \left\{ \begin{array}{l} V(s, t) = \\ [x(s, t), y(s, t)] \end{array} \right\}, \forall (s, t) \in [0, 1] \times [0, \infty] \quad (1)$$

with :

- s : abscissa curvilinear along the outline.
- t : evolution temporal of the curve in the image.
- $V(s, t)$: position of a point of the curve C at an instant t .

The energy functional (equation 2) associated with the active contour consists of two terms:

$$E_{snack}(v) = E_{interne}(v) + E_{externe}(v) \quad (2)$$

➤ Energy internal.

The internal energy of an active contour regulates the shape of the curve by ensuring its continuity and controlling its curvature (equation 3).

$$E_{int}(v) = \frac{1}{2} \int_0^1 (\alpha(s) \cdot |v'(s)|^2 + \beta(s) |v''(s)|^2) ds \quad (3)$$

With :

$$|v'(s)|^2 = \left(\frac{dx(s)}{ds}\right)^2 + \left(\frac{dy(s)}{ds}\right)^2$$

$v'(s)$: derivative first of $v(s)$ by report has s .

v'' : derivative second of $v(s)$ by report has s .

α : elasticity of outline.

β : rigidity of outline.

The internal energy of an active contour can be broken down into two fundamental components: the continuity term and the curvature term.

➤ Energy external

According to Kass , Witkin and Terzopoulos , the external energy of an active contour (equation 4) consists of two principal components [2] :

$$E_{ext}(v) = E_{image}(v) + E_{contrainte}(v) \quad (4)$$

Image energy,

Image energy guides the curve C toward object boundaries by relying on image features such as gradients and contours (equation 5):

$$E_{image}(v) = - \int_0^1 \lambda(s) \left\| \nabla (I(v(s))) \right\|^2 ds \quad (5)$$

where $\nabla I(v(s))$ is the image gradient and $\lambda(s)$ is a weighting factor.

High-gradient regions attract the curve toward relevant edges. Internal energy enforces smoothness, while external energy drives the contour to image boundaries. However, this approach has limitations, particularly in handling concave shapes.

2. GRADIENT VECTOR FLOW.

To improve the basic snake method, C. Xu and JL Prince developed the GVF snake method [3] introducing a new external force called Gradient Vector Flow (GVF) (equation 6):

$$F_{ext} = v = [u(x, y), v(x, y)] \quad (6)$$

The calculation of the GVF field takes place in two stages:

A. Calculation of the Contour Map (Edge) Map)

B.

C.

The contour map $f(x, y)$ of image $I(x, y)$ is determined by the derivative of the image. It is often expressed as follows (equation 7):

D.

$$f(x, y) = |\nabla(G_\sigma(x, y) * I(x, y))|^2 \quad (7)$$

where ∇ is the gradient and G_σ is a Gaussian filter with standard deviation σ .

This contour map generates vectors perpendicular to the contours, with high amplitudes near the contours and low amplitudes in homogeneous areas. In these homogeneous areas, a diffusion process is applied to orient the vectors also towards concave areas. This approach improves the convergence of active contours in areas of high concavity and around acute angles.

B. Calculation of flow of vector gradient:

The flux of the gradient vector is defined as a vector field

$V(x, y) = (u(x, y), v(x, y))$ that minimizes the following energy functional (equation 8):

$$\varepsilon = \iint (\mu_1 (u_x^2 + u_y^2 + v_x^2 + v_y^2) + |\nabla f|^2 \cdot |V - \nabla f|^2) dx dy \quad (8)$$

with :

$$u_x = \frac{\partial u}{\partial x}, u_y = \frac{\partial u}{\partial y}, v_x = \frac{\partial v}{\partial x}, v_y = \frac{\partial v}{\partial y}$$

in which, μ is a setting of regularization between the first and the second term of the integral.

This energy functional allows the GVF vector field to be optimized to achieve better convergence of active contours in complex areas, particularly those with concavities and acute angles.

3. DESCRIPTION OF THE GVF ALGORITHM

To concisely illustrate the various steps involved in the GVF segmentation method, the flowchart presented in Figure 2 provides an overview of the algorithm workflow, beginning with image preprocessing and concluding with the generation of the final segmented image.

Prior to applying the GVF method for image segmentation, an essential preprocessing stage is performed to enhance the quality of the extracted contours. Anisotropic filtering is particularly effective in this context, as it reduces image noise while preserving important structural features and edge information within the medical image.

Following preprocessing, the contour map is obtained by applying a Laplacian-of-Gaussian (LoG) filter, which enhances edge detection and improves the identification of relevant object boundaries. The external force components, (u) and (v), are calculated from the image gradients derived using Sobel operators. These components are subsequently regularized through Laplacian smoothing to stabilize their values during the iterative GVF diffusion process.

During each iteration, the contour evolution is updated by computing the total energy, (E), and comparing it against a predefined convergence tolerance. To ensure a uniform distribution of contour points along the snake, an interpolation procedure is incorporated to maintain an appropriate average spacing between consecutive points. This process requires defining a minimum distance, (dmin), and a maximum distance, (dmax), which establish the allowable range for the distance between neighboring contour points. The GVF method improves the convergence and robustness of active contours, particularly when segmenting objects with concave boundaries. This is achieved by extending the influence region of the image gradient forces, enabling the snake to effectively detect and adapt to complex object geometries.

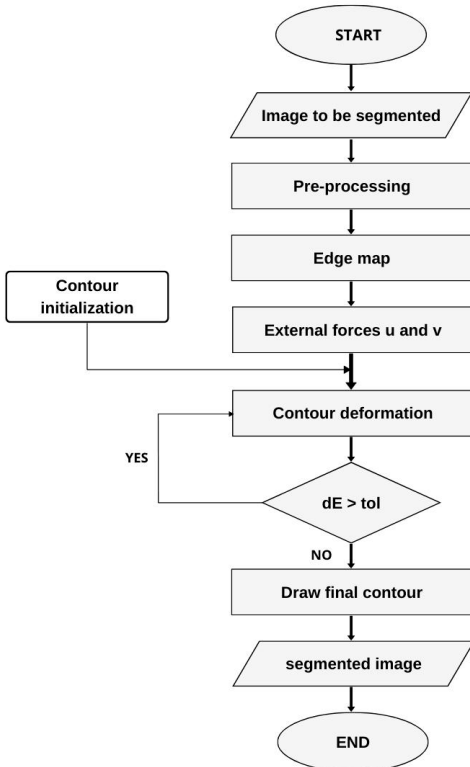


Fig 2: Flowchart of The GVF method

III. Regularized Level-Set Evolution Distance Method DRLSE

The DRLSE method is an evolution of Level Set methods. It introduces a regularization term that allows the distance function to be maintained without requiring resetting. This new approach uses an energy function composed of three terms, equation (9):

$$E(\phi) = \mu P(\phi) + \lambda L_g(\phi) + \alpha A_g(\phi) \quad (9)$$

Regularization term that allows the function to maintain its distance function form throughout the evolution.

$$P(\phi) = \int \frac{1}{2} (|\nabla\phi| - 1)^2 dx dy$$

Length term: The energy function $L_g(\phi)$ Allows the active contour to position itself close to the contour of the object of interest.

$$L_g(\phi) = \int g \delta_\epsilon |\nabla\phi| dx dy$$

δ_ϵ : Dirac function, defined as follows equation (10):

$$\delta_\epsilon(x) = \begin{cases} 0 & |x| > \epsilon \\ \frac{1}{2\epsilon} \left[1 + \cos\left(\frac{\pi x}{\epsilon}\right) \right] & |x| \leq \epsilon \end{cases} \quad (10)$$

With ϵ : regularization parameter.

Surface term : the energy function $A_g(\phi)$ Allows the speed of evolution of the active contour to be accelerated when the latter is far from the object of interest for a faster convergence.

$$A_g(\phi) = - \int g H(-\phi) dx dy.$$

Where $H(-\phi)$ is the Heaviside function,

The DRLSE model [4] incorporates a distance regularization term to stabilize the contour evolution function, thereby improving segmentation speed and eliminating the need to reset the function. The evolution equation then becomes (equation 11):

$$\frac{\partial\phi}{\partial t} = \mu \operatorname{div}(d_p (|\nabla\phi|) \nabla\phi) + \lambda \delta_\epsilon(\phi) \operatorname{div}\left(g \frac{\nabla\phi}{|\nabla\phi|}\right) + \alpha g \delta_\epsilon(\phi) \quad (11)$$

μ : Distance regularization parameter.

λ : Contour length regularization parameter.

α : Surface repulsion parameter.

d_p is defined by equation 12:

$$d_p(s) \cong \frac{p'(s)}{s} \quad (12)$$

Where p is a potential function used for distance regularization. For better distance regularization, the double-well potential function p_2 is used, which is expressed as follows:

$$P_2 = \begin{cases} \frac{1}{2\pi} (1 - \cos(2\pi s)), & \text{si } s \leq 1 \\ \frac{1}{2} (s - 1)^2, & \text{si } s > 1 \end{cases}$$

1. DRLSE ALGORITHM.

The organizational chart of the DRLSE method, describing its operation, is presented in Figure 3.

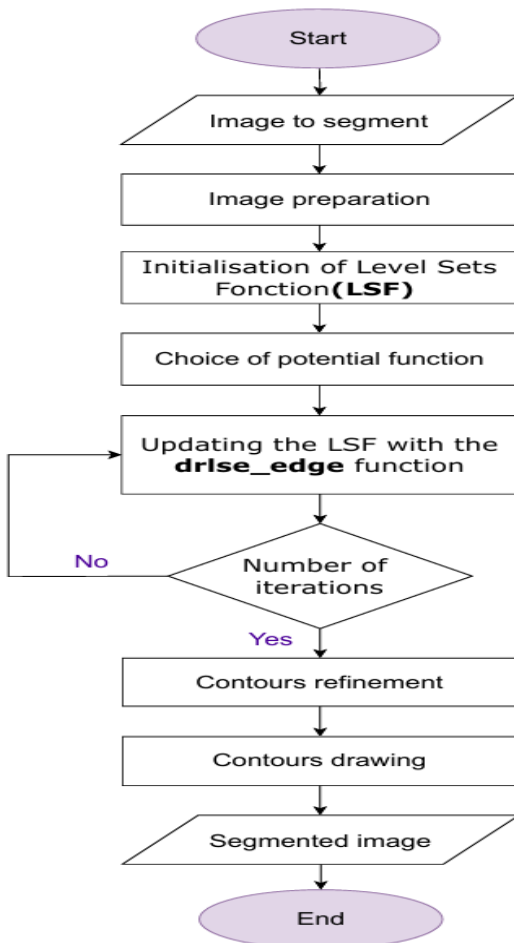


Fig 3: Flowchart of the DRLSE Method.

The proposed DRLSE algorithm starts with an image preparation stage, where the input image undergoes preprocessing to enhance its suitability for segmentation. During this stage, the user also specifies the parameters required for the segmentation process. Next, the level set function (LSF) is initialized as a binary function, with a

rectangular region defining the initial contour that will evolve toward the object boundaries. If required, multiple contours can be initialized simultaneously by specifying several initial regions.

The contour evolution process is then performed through a main iterative loop executed over a predefined number of outer iterations. During each iteration, the `drlse_edge` function updates the level set function (ϕ) according to the Distance Regularized Level Set Evolution (DRLSE) model.

The DRLSE formulation combines several energy terms to guide the contour evolution. The weighted length encourages smooth and regular contours by minimizing the contour length. The distance regularization term maintains the stability and regularity of the level set function by preserving its signed-distance property. In addition, the weighted area term drives the contour toward the target object by promoting the segmentation of interior regions while suppressing irrelevant boundaries. Together, these energy terms enable accurate and stable object segmentation.

The choice of the potential function is a key factor in the performance of level set-based segmentation methods. A single-well potential function is generally suitable for segmenting homogeneous regions, where object boundaries are relatively simple and a clear separation between regions is required. On the other hand, a double-well potential function is more effective for images containing complex structures, as it can simultaneously capture edge information and homogeneous interior regions. By providing a richer energy formulation, the double-well potential improves the stability of the evolving contour and enables it to accurately follow both sharp transitions and smooth image features.

Level set-based image segmentation methods, particularly those based on Distance Regularized Level Set Evolution (DRLSE), offer effective control of contour regularization while maintaining numerical stability throughout the evolution process. In addition, they provide accurate boundary detection and robust edge localization, making them well suited for segmenting complex images with irregular shapes and varying intensity distributions.

IV. IMPLEMENTATION ON FPGA.

Our implementation is based on the PYNQ-Z2 FPGA board, a development platform using the SoC Xilinx Zynq-7000. This board integrates both a Processing System (PS) and Programmable Logic (PL), offering an integral solution for the development of embedded hardware and software [8].

1. IMPLEMENTATION STEPS

Figure 4 shows a synoptic diagram of the steps we followed during the implementation.

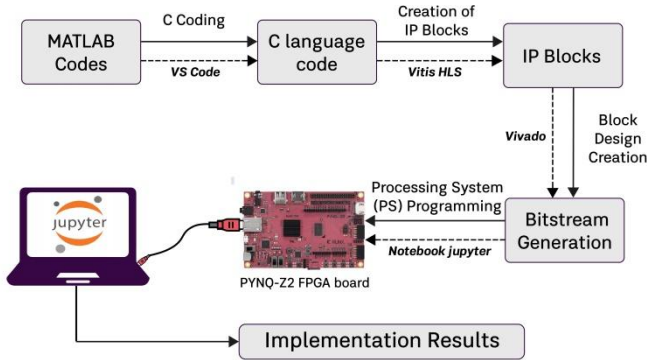


Fig. 4: Simplified synoptic diagram of the HLS approach

First, we write our program in C using Visual Studio Code (VS Code), then we run a simulation to ensure the program functions correctly by comparing the results with those generated in MATLAB. Before generating VHDL code, we perform exhaustive tests and simulations on our C code in Vitis HLS. This allows us to detect and correct potential errors earlier in the development process [6]. Vitis HLS also allows us to generate an IP Block. Using Vivado, we create a design block for the main system, which will be deployed on an FPGA and will include the previously generated IPs [7], [8]. Once the Block Design approved, we created A HDL Wrapper Who converts the plan of architecture in program HDL (Hardware Description Language). We synthesize the project, then we carry out the implementation and generate the binary file " Bitstream ".

The FPGA card is programmed using the generated Bitstream file and the system's proper functioning is verified.

2. GVF ALGORITHM ON FPGA

In this section, we detail the implementation steps of the GVF algorithm on an FPGA circuit (Figure 5). After developing a complete version of the algorithm, we will explain each step of the process in detail, as illustrated in the block diagram above.

We divided the first part of the GVF algorithm into two distinct blocks: one block for calculating the contour map f and another for the Sobel filter as well as the calculation of the variables bn , $c1$, and $c2$. The outputs of the first block are stored in DDR memory and then duplicated for retrieval by the second block. This process is also applied to the other intermediate variables for the two remaining blocks, which are dedicated to calculating the external force fields and the contour deformation.

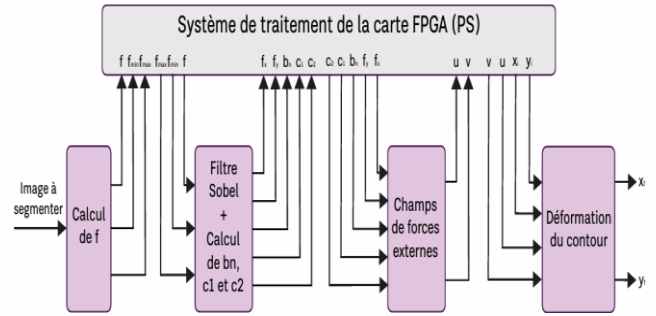


Fig. 5: Complete diagram of the GVF algorithm

Each block corresponds to a top function that we develop in C language and simulate on VS Code, and therefore to an IP block that we create on Vitis HLS.

The "contour map" function takes a filtered image as input to calculate the horizontal and vertical gradients (f_x and f_y), the gradient amplitude (bn), and the curvature characteristics ($c1$ and $c2$) necessary for contour detection. The "external force fields" function uses these values to generate the external force fields u and v . Finally, the "contour deformation" function receives the external fields u and v , along with the coordinates of the initial contour (x_i and y_i), and calculates the coordinates of the final contour (x_f and y_f).

Each block is designed to perform a specific function optimally, enabling robust and efficient implementation of the algorithm in an FPGA environment. Each step is rigorously tested, and the overall system operation is verified to achieve performance and accuracy targets.

3. DRLSE ALGORITHM ON FPGA.

The first stage of the implementation is divided into three distinct blocks (see Figure 6): convolution, calculation of (g) , and gradient computation. This subdivision enables more efficient management of hardware resources by storing the output data from each block in DDR memory and subsequently duplicating it for use by the following processing stages. Each block consists of a top-level function developed in the C language and simulated using Visual Studio Code, as well as an IP block generated using Vitis HLS for hardware implementation.

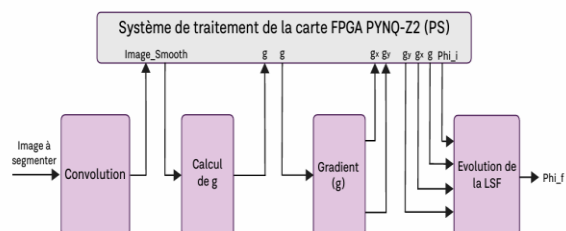


Fig. 6: Complete diagram of the DRLSE algorithm

The "LSF Evolution" function takes the initial distance function ϕ_i , g , g_x , and g_y and outputs ϕ_f . At each iteration, ϕ_f is calculated using an evolution equation that combines these data. The image contours (g) act as an external force that pulls the boundary of ϕ towards the detected contours in the image, allowing the LSF to accurately follow the contours. The gradients (g_x) and (g_y) are used to adjust the boundary to follow changes in image intensity and texture, ensuring precise and detailed segmentation.

The resulting block design provides a modular structure for the efficient implementation of the DRLSE algorithm on FPGAs. Each block is designed to optimally execute a specific function, enabling a robust and high-performance implementation of the algorithm within an FPGA environment. Each step is rigorously tested, and the overall system operation is verified to achieve performance and accuracy targets.

V. RESULTS AND INTERPRETATIONS.

The segmentation results obtained using the two algorithms simulated in MATLAB for medical images are presented. Additionally, the FPGA implementation is described. The efficiency of the proposed method in segmenting these images is evaluated by adjusting the algorithm parameters and analyzing their impact on the segmentation performance.

For the FPGA design, a block-based approach is adopted, in which predefined modules are utilized. Each block represents a specific function, such as processing units or memory controllers, and the blocks are interconnected through standardized interfaces to facilitate system integration and evaluate their influence on the segmentation results.

1. MATLAB SIMULATION

Before segmentation, preprocessing is applied to improve image quality. This includes:

- anisotropic filtering
- noise reduction

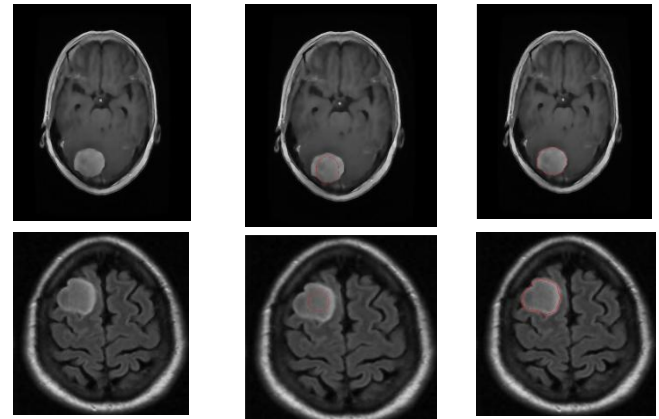
The GVF and DRLSE methods were tested on medical images (brain MRI, breast).

A. Results of Segmentation by the GVF Method .

➤ *Results of MRI images of a brain.*

After optimizing the parameters and initialization of the active contour for the GVF method, we are now applying this segmentation technique to medical images, including brain and breast MRIs.

Figure 7 illustrates the results obtained by applying the GVF method to MRI images of a brain with the parameters, $\mu = 0.15$, $\alpha = 0.1$, $\beta = 0.06$, $\text{tol} = 0.004$, $\text{itmax} = 300$.



(a) Picture initial (b) Outline initial (c) Result

Fig. 7: Segmentation with GVF On MRI of a brain

➤ *Image results MRI of a breast.*

Breast MRI segmentation using the GVF method has encountered difficulties. Breast MRIs, characterized by intensity variations and complex textures, complicate the application of GVF. Noise and similar anatomical structures disrupt the external force field, leading to incorrect convergence of the active contours. Consequently, the segmented contours do not accurately correspond to the actual boundaries of the structures of interest.

B. Results of segmentation using the DRLSE method.

➤ *Results of MRI images of a brain.*

To segment these images, we empirically adjusted the parameters α , λ and μ to find the optimal combination that allows efficient segmentation (figure 8).

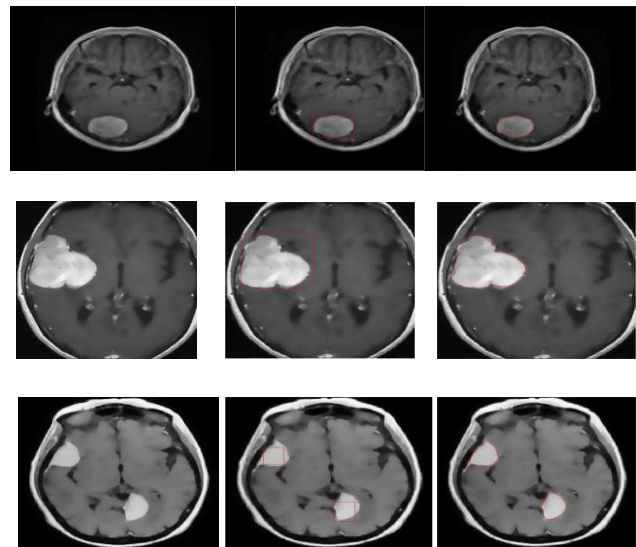
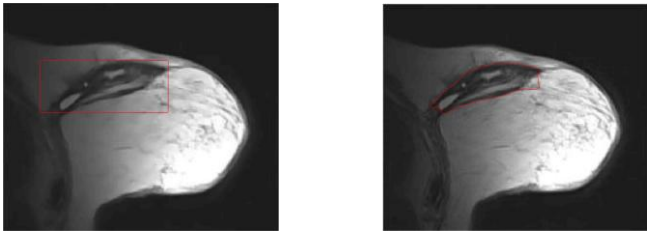


Fig. 8: Segmentation on brain MRIs $\alpha = 0.5$, $\lambda = 6$ and $\mu = 0.02$.

➤ **Results on MRI images of a breast**

After several parameter adjustments, applying the DRLSE algorithm to this new medical image yielded a less precise result in terms of segmenting the structures of interest. This observation led us to consider other approaches to improve the segmentation quality. Figure 9 shows the segmentation result with the parameters: $\alpha = 1.5$, $\lambda = 5$, and $\mu = 0.02$.



(a) Outline initial (b) Picture segmented

Fig. 9: Segmentation result with the DRLSE algorithm on breast MRI

2. FPGA IMPLEMENTATION

A. Implementation using the GVF method on FPGA

Each block represents a specific function, such as processing units or memory controllers, and the blocks are interconnected through standardized interfaces to enable communication between them. This modular approach facilitates efficient utilization of FPGA resources, thereby simplifying the overall FPGA design process. Figure 10 presents the final block-based design of the proposed architecture for the complete implementation of the GVF algorithm [9].

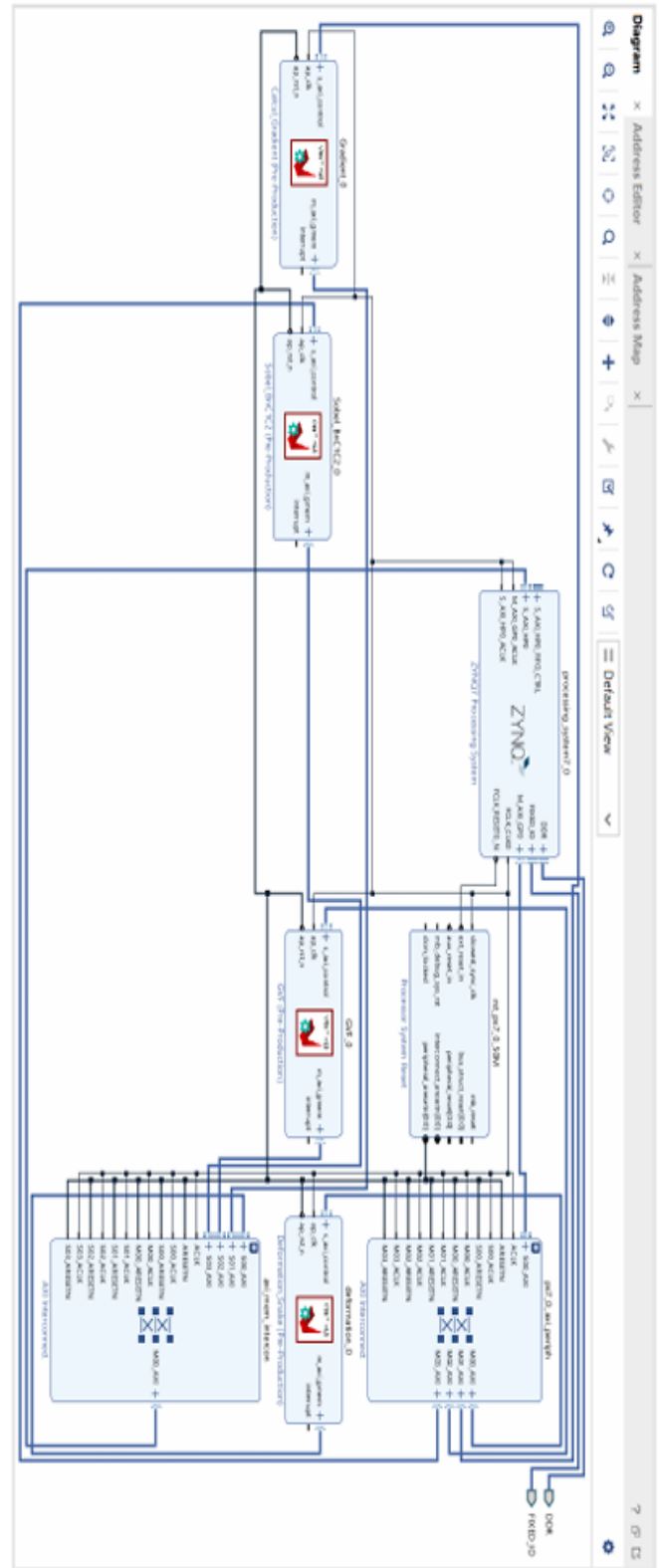
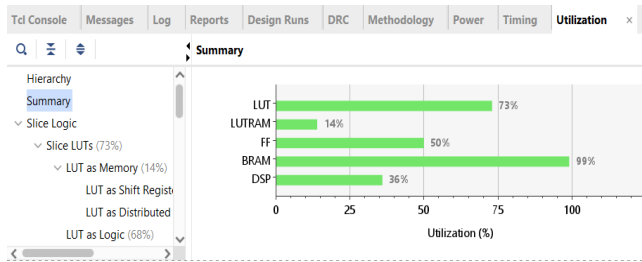
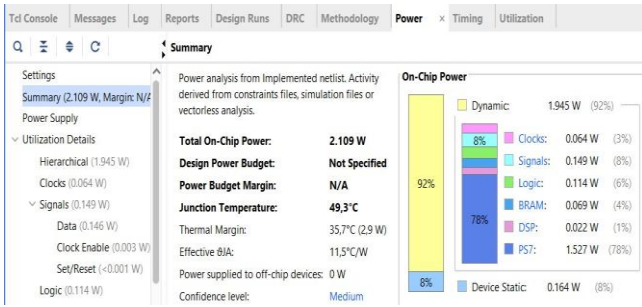


Fig. 10: Block design for the complete implementation of the active contour algorithm using the GVF method

Figure 11 presents the utilization and power ratios of complete architecture



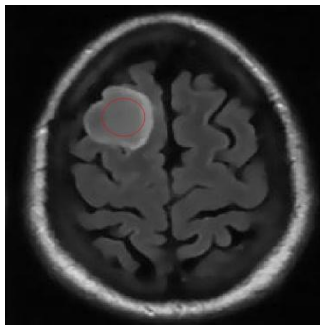
Usage report



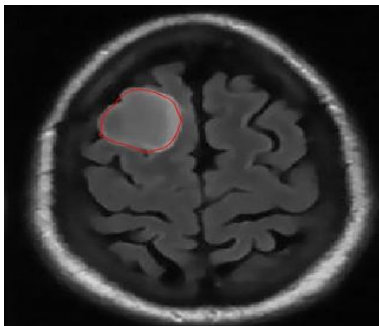
Report on power

Fig. 11: Implementation of The GVF algorithm on FPGA

By maintaining identical parameter settings and the same initial contour, the results obtained from the MATLAB simulation and the FPGA implementation are visually equivalent. It can be observed that segmentation accuracy is preserved, with the FPGA-based results showing a similar visual appearance to those obtained in MATLAB (Figure 12). The execution time of the complete GVF architecture is measured at 12.786 s.



Initial contour



segmented image on FPGA

Fig. 12: Result of the implementation of GVF on FPGA

The slight differences observed stem primarily from numerical precision. MATLAB uses double-precision (64-bit) floating-point numbers, while the FPGA implementation relies on single-precision (32-bit). This reduced precision leads to small deviations during calculations, particularly in repetitive operations such as gradient calculations (f , f_x , f_y), where rounding errors can accumulate. Furthermore, optimizations applied during synthesis with Vitis HLS can also introduce minor variations.

B. Implementation of the DRLSE

The final block-based design of the proposed architecture for the complete implementation of the DRLSE algorithm is illustrated in Figure 13 [10].

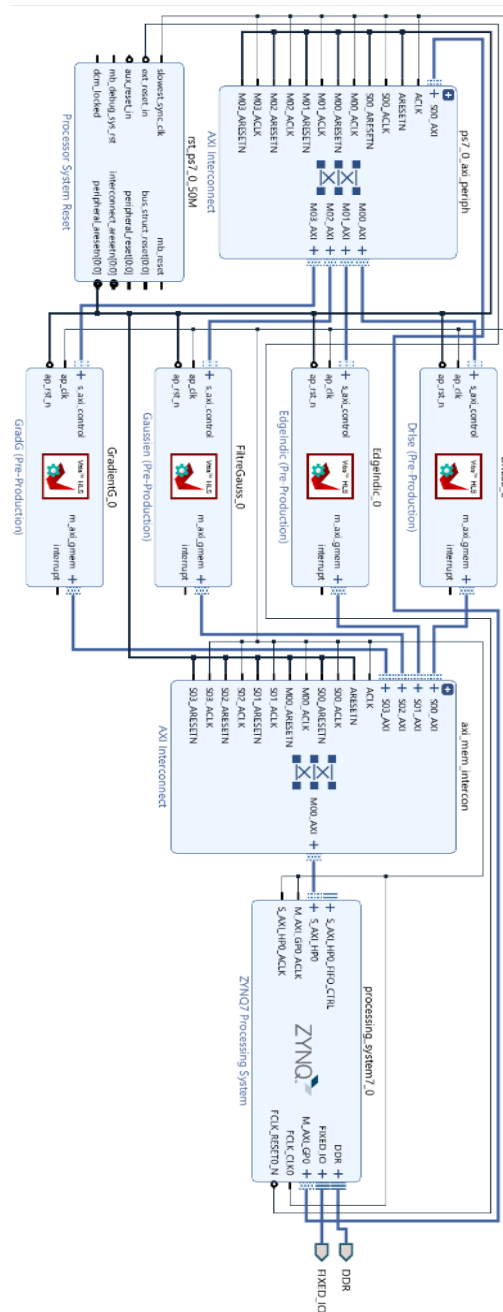


Fig. 13: Design by blocks for the implementation of the algorithm DRLSE

Each block represents a specific function, such as processing units or memory controllers, and the blocks are interconnected through standardized interfaces to enable communication between them. By maintaining the same parameter configuration and initial contour used in the MATLAB simulation, the segmentation results obtained from the FPGA implementation of the DRLSE method are visually equivalent to those obtained previously. The corresponding segmentation result is presented in Figure 14.

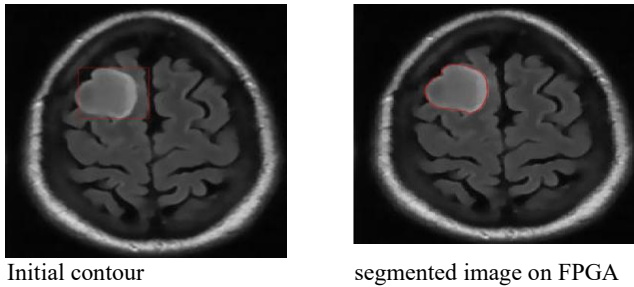


Fig. 14: Result of the implementation of their method DRLSE

The FPGA implementation of the DRLSE algorithm demonstrated satisfactory segmentation accuracy, with a rigorous visual comparison performed against the results obtained from the MATLAB simulation. Figure 15 presents the resource utilization and power consumption analysis of the DRLSE algorithm implemented on the FPGA. It highlights the main hardware resources, including lookup tables (LUTs), block RAMs (BRAMs), and digital signal processing slices (DSPs), as well as the estimated power consumption.

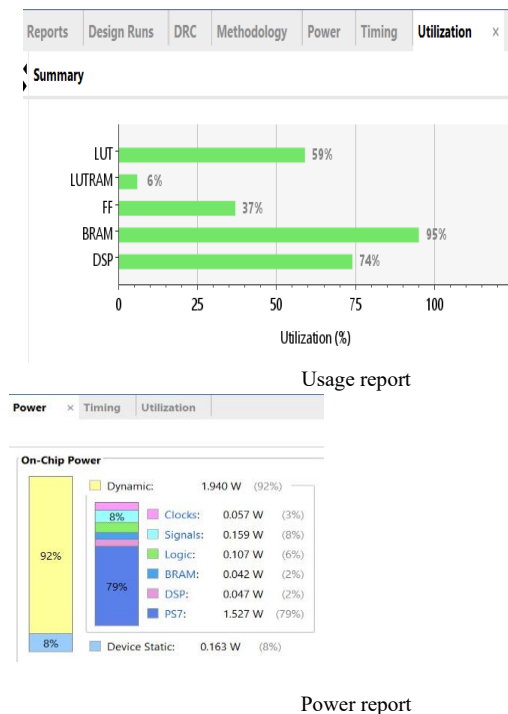


Fig. 15: Implementation of the algorithm DRLSE on FPGA

VI. COMPARISON OF SEGMENTATION RESULTS: GVF VS DRLSE ON FPGA.

The two algorithms, GVF and DRLSE, were implemented on an FPGA and tested under similar conditions. The results show good visual accuracy for both approaches, with detected contours close to those obtained using MATLAB. GVF is efficient for concave and complex shapes, with moderate FPGA resource consumption. DRLSE provides sharper and more precise contours, in particular

For geometric structures, but with greater use of hardware resources (DSP, BRAM). In terms of performance:

- The execution time is slightly faster for GVF compared to DRLSE.
- The numerical differences between the FPGA and MATLAB results are minimal and do not impact visual quality.

GVF is lighter, while DRLSE offers finer segmentation, at the cost of higher hardware complexity.

To achieve these results, we optimized our calculations by grouping them into a single execution. This required the use of buffers for matrix updates, thus increasing the use of computing resources and extending the overall execution time, measured at 105.8523 seconds. A distinctive feature of our approach was the frequent subdivision of a block into sub-blocks, favoring the use of DDR memory rather than the FPGA's BRAM. This strategy aimed to maximize the efficiency of available memory, even though data accesses memory is generally slower than the logic part of the FPGA.

VII. CONCLUSION

This work made it possible to compare two advanced methods of medical image segmentation and to demonstrate the value of their implementation on FPGA.

The GVF is effective for concave contours, while the DRLSE is more robust for complex structures.

The FPGA implementation demonstrates remarkable efficiency in utilizing hardware resources, fully leveraging the inherent parallelism of this architecture. It enables a significant reduction in execution time while maintaining high accuracy. FPGAs prove particularly well-suited to complex medical image segmentation algorithms, as they facilitate the optimization, reuse, and adaptation of modules according to processing needs. They thus offer an excellent balance between performance, accuracy, and flexibility, while paving the way for more scalable and high-performance hardware architectures for real-time image processing.

REFERENCES

- [1] S. Bakas et al., "Identifying the best machine learning algorithms for brain tumor segmentation, progression assessment, and overall survival prediction in the BRATS challenge," arXiv preprint arXiv:1811.02629, 2018.
- [2] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active contour models," *International Journal of Computer Vision*, vol. 1, no. 4, pp. 321–331, 1988, doi: 10.1007/BF00133570.
- [3] C. Xu and J. L. Prince, "Gradient vector flow: A new external force for snakes," in *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, San Juan, PR, USA, 1997, pp. 66–71, doi: 10.1109/CVPR.1997.609407.
- [4] Y. Chen, G. Chen, Y. Wang, and N. Dey, "A distance regularized level-set evolution model based MRI dataset segmentation of the brain's caudate nucleus," *IEEE Access*, vol. 7, pp. 133093–133104, 2019, doi: 10.1109/ACCESS.2019.2937964.
- [5] J.-J. Rousselle, "Active contours: A segmentation method. Application to medical imaging," Ph.D. dissertation, 2003.
- [6] A. Jarrah and S. Amri, "Optimized FPGA-based implementation of brain tumor detection by combining K-means and gray wolf optimization algorithms," *Traitement du Signal*, vol. 39, no. 6, pp. 1879–1891, 2022, doi: 10.18280/ts.390601.
- [7] AMD Xilinx, *Vitis High-Level Synthesis User Guide (UG1399)*, Version 2021.2, 2021. [Online]. Available: <https://docs.amd.com/r/2021.2-English/ug1399-vitis-hls>
- [8] AMD Xilinx, *Vivado Design Suite User Guide (UG973)*, Version 2021.1, 2021. [Online]. Available: <https://docs.amd.com>
- [9] M. L. Hamidatou, F. Z. Hamadi, L. Hamami, B. Bouzid, S. A. Belkacem, and F. Z. Allam, "Hardware implementation of the GVF approach for deformable contour detection on FPGA," *Engineering, Technology & Applied Science Research*, vol. 15, no. 6, pp. 30239–30245, Dec. 2025.
- [10] F. Z. Hamadi, M. L. Hamidatou, L. Hamami-Mitiche, S. A. Belkacem, and B. Bouzid, "MATLAB simulation and FPGA implementation of the DRLSE segmentation algorithm," *Traitement du Signal*, vol. 42, no. 2, pp. 719–727, Apr. 2025, doi: 10.18280/ts.420210..

computer science and physics. He is also a Research Master at the LMCS laboratory of the Higher National School of Computer Science (ESI) in Algiers. After obtaining a graduate diploma in Electronics from USTHB, Algiers, he earned a master's degree in electronic communications and Signal Processing from the University of Blida, Algeria, followed by a Doctorate (Doctorat en Sciences) in the field of image and speech processing from the National Polytechnic School (ENP) of Algiers. Several final-year student projects have been carried out under his supervision in the areas of image and speech processing.

Latifa Hamami-Mitiche (M.Sc. 1988, Ph.D. 2002) is a full Professor in the Department of Electronics at Ecole Nationale Polytechnique of Algiers, Algeria. She is author more than 150 scientific papers. Her research interests include Medical Image Processing, ISAR Image Processing, Optical Character Recognition, Biometrics and Embedded System, FPGA and ASIC Design, System on Chip.

Fatma Zohra Hamidatou-Hamadi is an Assistant Professor at the National School of Agronomy. She teaches physics courses at ENSA. She is the author of several international publications. Her research focuses on image processing, particularly medical images. Her objectives are multiple, including the implementation of intelligent embedded systems dedicated to medical diagnosis support on FPGA-based embedded circuits.

Hamidatou Mohamed Lamine is currently a Senior Lecturer (Maître de Conférences, Classe A) at the National School of Agronomy (ENSA) in Algiers, where he teaches courses in