Application of geomatics techniques and genetic programming to medical image segmentation

Emanuela Genovese [1], Clemente Maesano [1], Elena Barrile [3], Davide Chiffi [3], Vincenzo Barrile [2]

[1] Department of Civil, Building and Environmental Engineering, La Sapienza University of Rome, Italy –
emanuela.genovese@uniroma1.it, clemente.maesano@uniroma1.it
[2] Department of Civil Engineering, Energy, Environment and Materials (DICEAM), Mediterranea University of Reggio Calabria –
vincenzo.barrile@unirc.it
[3] Department of Nervous and Mental Diseases – La Sapienza University of Rome, Italy –
elena.barrile, davide.chiffi@uniroma1.it

Keywords: Genetic Programming, denoising, Watershed, medical imaging, segmentation.

Abstract

Ultrasound imaging is a promising approach in medicine due to it being a non-intrusive technique, with the ability to obtain important data from a large number of patients. Despite its potential, images generated using this technique are still severely corrupted with high levels of noise, especially speckle, which hinders their further processing. The present study introduces an alternative, innovative method for medical image segmentation, combining a bilateral filter, a specially developed genetic programming algorithm, the CLAHE algorithm, and the Watershed segmentation technique, which is typically used for geotopographic images. The results from this methodology indicate that the proposed approach delivers results comparable to those of conventional neural networks, while requiring a smaller image dataset. The integration of the genetic algorithm offers a novel solution by enhancing local contrast, reducing image noise, and improving the Watershed segmentation process.

1. Introduction

Ultrasound imaging is a valuable technique in medicine due to its non-invasive nature and its capacity to collect data from a large number of patients. However, despite its advantages, ultrasound images are often severely affected by noise, particularly speckle, which compromises further processing. Recently, machine learning has gained significant attention in the areas of medical image denoising and segmentation, as it allows for better generalization to new datasets compared to traditional methods. While neural networks are the most commonly used algorithms in this field, their adoption is limited by the requirement for large datasets during the training phase. This has prompted research into alternative methods, such as algorithms and Cellular Automata, which do not have this particular limitation.

For example, a genetic algorithm was used in (Sahe and Singh, 2024) to reduce speckle noise in Optical Coherence Tomography (OCT) images, with the primary objective of optimizing wavelet transform coefficients. These coefficients are essential in defining wavelets, a type of signal decomposition where the Fourier Transform is a specific case. Evolutionary algorithms have been highlighted in various studies as promising tools in medical image processing, as noted in a critical review of bio-inspired techniques (Ezhilarasan et al., 2024).

Another interesting approach to image denoising is Cellular Automata, which also does not require large training datasets, although there is limited research on its application in this field. In (Bhardwaj et al., 2019), a cellular automata technique was introduced for despeckling medical images. This method works by calculating a threshold value based on a Moore neighborhood (a 3x3 grid around a central pixel). If the threshold exceeds a certain value, the individual pixel computes the median of its surrounding pixels; otherwise, it computes a different median.

There are also studies that explore the combination of cellular automata with neural networks to improve performance when working with small datasets. In (Korevaar et al., 2024), the study demonstrated the effectiveness of U-Nets with limited training data, noting how the network's performance can be influenced by changes in data distribution. An intriguing aspect is that this integration can lead to a three-fold reduction in network size. The key to this integration is that the neural network can automatically generate rules, as opposed to cellular automata which typically rely on a fixed set of rules. Another example of this combined approach in medical image denoising is found in (Bhardwaj et al., 2022).

Despite the potential of cellular automata, they can have drawbacks, such as longer inference times. In contrast, Genetic Algorithms (GA) and Genetic Programming (GP) offer an alternative that, while more computationally intensive in terms of training, can produce excellent results after optimization. These techniques require extensive experimentation to find the ideal evolutionary parameters (e.g., crossover, mutation, population size, number of generations, and in the case of genetic programming, also the functions used by the algorithm). Beyond (Fajardo-Delgado et al., 2024), there are additional studies that employ GP for denoising medical images. In (Vanneschi and Poli, 2022), GP was applied to ultrasound image denoising, with GP used to detect noisy pixels rather than directly filtering them. This "switching scheme" prevents the potential corruption of clean pixels, a common issue when applying filters directly to the entire image (Javed et al., 2018). In (Khan et al., 2018), GP was also used for removing Rician noise from MRI images.

Many of these techniques can also be applied to medical image segmentation, often with the addition of other methods like Watershed segmentation, which is widely used in geomatics. In (Cortacero et al., 2023), Cartesian Genetic Programming (CGP) was employed for medical image segmentation, using a Cartesian grid to represent different functions with assigned coordinates. This method is particularly useful for working with limited datasets and offers a clear interpretability of the models. For example, the algorithm can automatically determine the optimal sequence of operators (such as Gaussian Blur, image difference, and Median Blur) to improve segmentation accuracy. As in other GP studies, careful selection of functions and parameters is crucial. The study also explores the possibility of integrating data from multiple sources, such as demographic

This contribution has been peer-reviewed.
https://doi.org/10.5194/isprs-archives-XLVIII-2-W9-2025-105-2025 | © Author(s) 2025. CC BY 4.0 License.

information, using the Mixed-Type Cartesian Genetic Programming (CGP) approach.
Another application of GP to medical image segmentation is discussed in (Poli, 1996), where it was used for segmenting blood vessels in MRI images of the brain, with simple functions like arithmetic operations and average filters. GP outperformed neural networks in this case. Additionally, in (Dhot, 2099), GP was applied to medical image segmentation, focusing on the evolution of small-sized programs, which is a key issue in GP. It is worth noting that these algorithms not only serve as alternatives to neural networks but can also aid in the development of more efficient neural network architectures. In (Ramesh et al., 2024), GP was used to design Convolutional Neural Networks (CNNs) that require less training time and fewer computational resources for segmentation tasks.
In our approach, we integrate Genetic Programming with Contrast Limited Adaptive Histogram Equalization (CLAHE) for improved image segmentation, a technique already explored in previous studies (Rai et al., 2012), (Intyanto et al., 2024). Other studies, such as (Barrile et al., 2023a), (Barrile et al., 2023b), also train Spiking Neural Networks (SNNs) to identify noise patterns that interfere with Watershed segmentation, which is applied after denoising. Additionally, [18] proposes a switching filter for impulse noise, designed to filter noisy pixels selectively.

In this context, we present a pipeline where images are first filtered to focus on the pixels with the most noise. This is followed by CLAHE histogram equalization and Watershed segmentation. Genetic programming is then employed to determine the most effective sequence of filters for accurate segmentation of a given medical image. The main distinction of our approach is that it requires only a single image for training, and the filtering process is applied only to selected pixels, improving efficiency and reducing the risk of corrupting clean image data.

2. Mathodology

The proposed approach follows different phases shown in Figure 1.

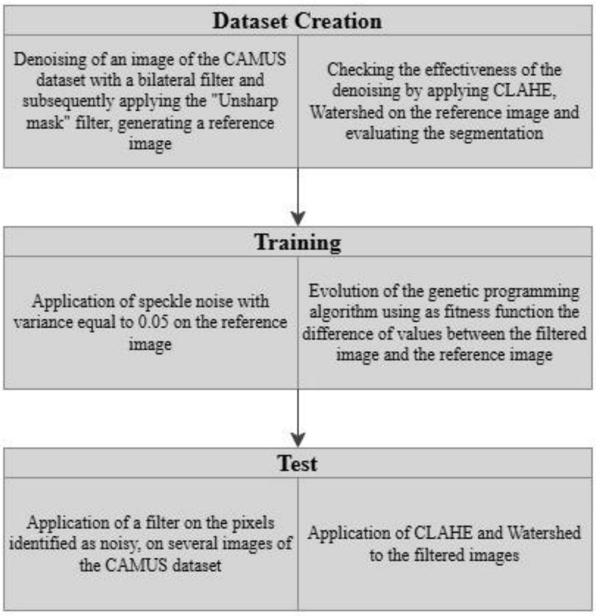

Figure 1. Proposed methodology.

The procedure used can be summarized as follows:

Training Image Creation:
Application of a bilateral filter to an image from the dataset → Application of the "Unsharp-masking" filter to the bilaterally filtered image → Evaluation of the bilateral filter's performance through histogram equalization followed by Watershed segmentation → Generation of speckle noise on the image obtained from the bilateral filter and the "unsharp-masking" process.

Training:
Genetic programming is trained on the reference image, which includes the bilateral filter, unsharp masking, and speckle noise generation. The evolutionary process identifies the best individual whose denoising method maximizes the PSNR/ratio calculated between the speckle-noise-corrupted image and the original (bilaterally filtered and unsharp-masked) image.

Testing:
The filter derived from the genetic programming evolution is applied to other images from the CAMUS dataset → The CLAHE algorithm is applied to the filtered image → Finally, Watershed segmentation is performed.

2.1 Phase 1: Dataset Creation

The first step in the methodology involves generating the dataset. The training set is created by applying a sequence of algorithms to an image, which will later be used for comparison with an unfiltered original image during the training phase. Since the dataset images are in .nii format, they are imported using the nibabel library. Following the approach used in (Barrile et al., 2023a), a stretching operation is applied to the central part of the image. This step is motivated by the observation in the referenced article that stretching the central region emphasizes it more effectively.
The image is then converted into an 8-bit unsigned integer format, compatible with OpenCV. A bilateral filter is applied next, as it has shown to deliver better segmentation results through Watershed compared to the “Non-local Means Denoising” filter. Despite its effectiveness, the bilateral filter is relatively slow, which can be a disadvantage in scenarios requiring rapid image processing.
After filtering the image, histogram equalization and Watershed segmentation are applied to assess the filter's performance. The filtered image is then processed using the "unsharp masking" filter to enhance fine details by subtracting a Laplacian-filtered version of the image. Subsequently, speckle noise is added to this image using the "scikit-learn" library.

2.2 Phase 2: Genetic Programming Implementation

The next step involves implementing the genetic programming algorithm. In this work, a genetic algorithm is designed to remove speckle noise from degraded images. Individuals in the genetic programming population represent complex mathematical functions applied locally to image pixels. Each individual consists of a syntactic tree with mathematical and logical functions that operate on local image properties, such as variance, standard deviation, and median values from central windows of size 3x3 or 5x5, in combination with the original and pre-filtered image versions.

These individuals are transformed into executable functions that are applied to the noisy image pixel by pixel. The result is a binary mask: pixels deemed noisy are replaced with filtered values (e.g., via smoothing), while non-noisy pixels remain unchanged. The quality of the resulting filtered image is evaluated by comparing it to the reference image using metrics like PSNR, SSIM, and MAE. The fitness function is defined as the ratio between PSNR and MAE, with possible penalties for execution time.

During the development of the algorithm, a non-deterministic behavior was observed, meaning multiple executions with the same input produced slightly different results, making the metrics inconsistent and destabilizing the genetic selection process. Initially, this variability was attributed to floating-point errors in the calculations. However, after further investigation, the cause was traced to the multi-threaded nature of the underlying numerical libraries (e.g., BLAS or OpenBLAS, used by Numpy). The concurrent execution of threads alters the order of operations, which affects the final result. To resolve this, a constraint was imposed to use only one thread for the BLAS backend, ensuring deterministic execution and avoiding race conditions.

The best individuals from the evolutionary process were saved via binary serialization using the Pickle library for later use, avoiding the need to retrain the algorithm. The entire evolutionary process was implemented using the DEAP library, where a strongly typed primitive set was used. The primitives are rigorously typed to act on specific input types, ensuring semantic consistency in the generated trees. Primitives include protected arithmetic operations, trigonometric and logarithmic functions, sigmoid functions, and a logical "decide" function that performs point-to-point comparisons between arrays. The evolutionary tree structure ensures the root node is the decision function, intermediate nodes are unary or binary functions, and terminals are either constant values or functions computing local image properties.

To avoid tree bloating, the initial population was created using the GenGrow method, which allows for greater structural diversity compared to the GenFull method. Evolution was driven by an evaluation function that considers both the image quality and computational efficiency, favoring individuals that produce high-quality results in a shorter time.

2.3 Phase 3: Filtering and Segmentation

In the final phase, a combination of the bilateral filter, CLAHE algorithm, and Watershed segmentation is applied. The CLAHE algorithm, as previously mentioned, adapts histogram equalization to enhance contrast locally. The image is divided into small blocks, called "tiles," for which the histogram is calculated. Values exceeding a certain threshold are redistributed to improve contrast, and the cumulative distribution function is used to normalize the resulting image. After this, the blocks are merged back together.

Through the various experiments (both in the image creation phase and the validation of the evolutionary algorithm), it was found that applying histogram equalization significantly improved segmentation performance. The application of CLAHE, in particular, enhanced Watershed segmentation, suggesting the need for contrast enhancement algorithms to improve segmentation precision.

Watershed segmentation, originating from geomatics, simulates the flooding of regions and applies distinct colors to different segments, making it ideal for real-time applications due to its high execution speed (Barrile and Genovese, 2024), (Barrile et al., 2024). However, one challenge with Watershed segmentation is the risk of over-segmentation, where too many small regions are created, leading to less meaningful results.

3. Results

As mentioned, the first phase of the proposed methodology involved generating the dataset by applying the bilateral filter. The parameters used for the bilateral filter, CLAHE filter, and the Genetic Programming (GP) algorithm are summarized in Table 1 below:

Parameters	**Bilateral filter**	**CLAHE filter**	**GP algorithm**
Diameter	19	-	-
Sigma_Color	20	-	-
Sigma_Space	20	-	-
Clip Limit	-	2	-
Tile Grid Size	-	8x8	-
Evolution strategy	-	EaSimple	-
Crossover probability	-	0.9	-
Mutation probability	-	-	0.09
Population size	-	-	100
Max tree height	-	-	15
Number of generations	-	-	30
Tree generation function	-	-	GenGrow

Table 1. Pipeline parameters.

The algorithm was designed to generate a binary mask, where each pixel is assigned a value of 1 if it is identified as noisy, and 0 if it is considered clean. For pixels marked as 1 (noisy), a 5x5 Gaussian filter is applied to smooth the noise. Pixels marked as 0 (unaffected) remain unchanged, ensuring that only noisy pixels are selectively denoised while the original image content is preserved for clean pixels. This targeted approach allows for efficient noise removal, without altering the unaffected regions of the image.

To assess the effectiveness of this algorithm, we used the PSNR/MAE ratio as the primary fitness measure, a common approach in the literature for evaluating image denoising performance (Sahe and Singh, 2024). This ratio balances Peak Signal-to-Noise Ratio (PSNR) and Mean Absolute Error (MAE), offering a quantitative measure of the algorithm's denoising quality. In addition to this, we also explored an alternative evaluation method based on segmentation performance. Here, the segmented images were compared to a reference image, with metrics like adapted Rand error and the Dice score used to quantify the similarity between the predicted segmentation and the reference segmentation. These metrics offer insights into how well the segmentation produced by the denoised image aligns with ground truth data.

Figure 2 illustrates the evolution of the genetic programming algorithm over 30 generations. As the algorithm iterates, there is a noticeable improvement in both the fitness measure and the segmentation performance. The gradual optimization of the filtering process demonstrates the algorithm's ability to adapt and refine its approach to denoising. This refinement ultimately leads to better noise removal and improved segmentation quality, highlighting the effectiveness of the proposed method in handling noisy medical images.

Through these evaluations, it becomes clear that the algorithm not only improves the quality of denoising but also contributes to better segmentation results, particularly when compared to traditional methods. This makes the approach a valuable tool for medical image processing, especially when dealing with noisy datasets.

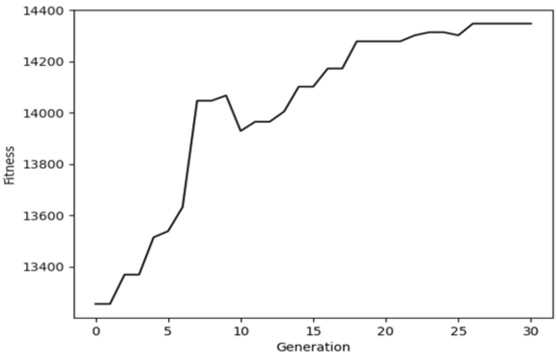

Figure 2. Evolution of the genetic programming algorithm for 30 generations.

For testing the algorithm, we follow a similar procedure to the one used during training. The central part of each test image is stretched, as done in the training phase, to emphasize key regions of the image. Then, we apply the best evolved individual obtained from the genetic programming algorithm to the test images. After the noise is removed, we proceed by applying Watershed segmentation to segment the image into distinct regions.

To evaluate the performance of the denoising and segmentation process, we compare the results of applying CLAHE (Contrast-Limited Adaptive Histogram Equalization) after the denoising step versus not using CLAHE. This comparison helps determine the impact of contrast enhancement on the segmentation accuracy, especially after noise reduction.

All images used for testing are sourced from the CAMUS dataset (Human Heart Project, accessed on 28 April 2025), which includes medical images with varying levels of noise. In addition to the CAMUS dataset, we also plan to extend the algorithm's application to other medical datasets, particularly those with limited data availability, to assess its generalizability and robustness.

The results of these tests are shown in the following images (Figures 3, 4, 5, 6), which illustrate the impact of our proposed method on the quality of segmentation, with and without CLAHE post-denoising. These results help demonstrate the efficacy of the genetic programming algorithm combined with CLAHE and Watershed segmentation in medical image processing.

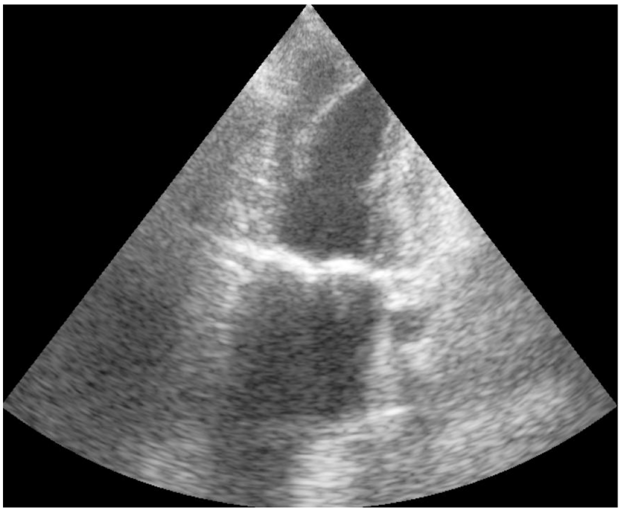

Figure 3. Raw image from CAMUS dataset.

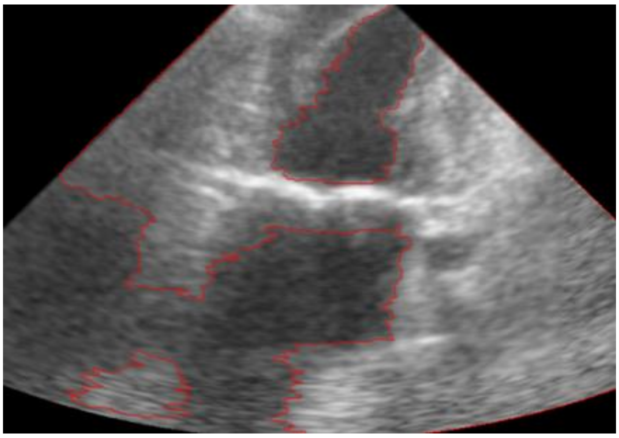

Figure 4. Segmentation performed on the image, with denoising applied on every pixel

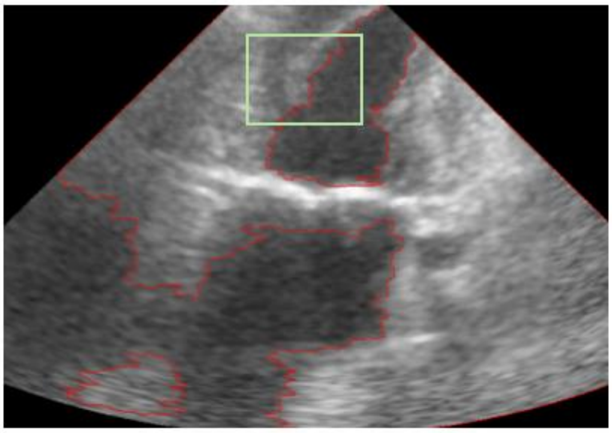

Figure 5. Segmentation performed by applying the genetic programming algorithm on selected pixels.

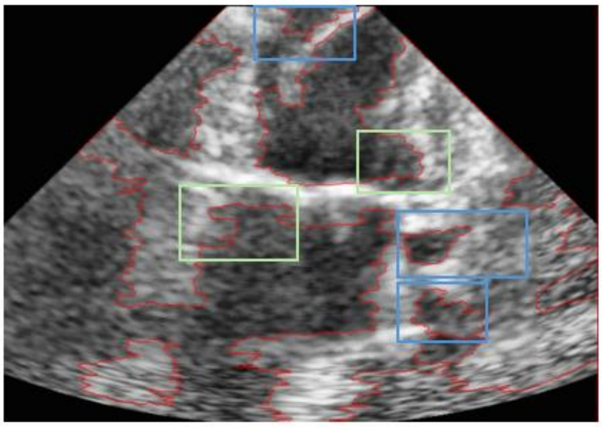

Figure 6. Watershed segmentation applied after denoising on pixel identified as noisy and CLAHE histogram equalisation.

In Figure 6, green rectangles highlight areas where the segmentation aligns closely with the actual edges of the structures in the image, indicating good segmentation accuracy. On the other hand, blue rectangles indicate areas where oversegmentation has occurred, meaning that the segmentation algorithm has split regions incorrectly or too finely. This oversegmentation can be a result of small variations in the image, such as noise or lighting inconsistencies. Upon examining the images, it becomes clear that differences in illumination are often as influential as the noise itself in determining segmentation quality. This suggests that for further improvement, the algorithm needs to account for illumination variations in addition to noise removal. While CLAHE has provided better segmentation in many cases by enhancing contrast, relying solely on CLAHE to address illumination discrepancies does not yield optimal results for real-world applications. Although CLAHE helped improve segmentation in some cases, the method is still limited when dealing with complex lighting conditions. To evaluate the reliability of the proposed methodology, we present the average results from the comparisons in Table 1. These results assess the performance of our method under varying conditions, such as the number of randomly sampled pixels and the choice of classifiers. The segmentation performance of our methodology was compared with that of ITK-SNAP, a widely used software tool for manual and semi-automatic segmentation of medical images. The Intersection over Union (IoU) metric was used for comparison, and the method yielded a value of 0.94, indicating excellent agreement in certain regions of the image when compared to ITK-SNAP’s segmentation. Additionally, in Table 2, we compare the performance of our methodology with other commonly used segmentation and classification techniques, applied across various regions of the images.

Method	**Accuracy**	**Precision**	**Recall**
Thresholding	0.83	0.82	0.77
Convolutional Neural Network (CNN)	0.93	0.90	0.89
Self-Normalising Neural Network (SNN)	0.94	0.92	0.89
Genetic Programming	0.93	0.92	0.94

Table 2. Comparison of the proposed methods with other common segmentation techniques.

The results highlight variations in performance depending on image conditions, such as noise levels and illumination differences. This further emphasizes the need for a more adaptive approach that can handle different image characteristics effectively.

While the proposed methodology shows promising results, incorporating further improvements to handle illumination variations and refine oversegmentation is a key next step for making it more robust and applicable in real-world medical imaging scenarios.

4. Conclusion

In this study, a genetic programming algorithm was applied selectively to pixels that were considered to be affected by speckle noise, with the algorithm being trained on a single image. After training, the images from the test set were processed using a histogram equalization algorithm (CLAHE) to enhance contrast. Although the evolution of the algorithm was modest, with improvements around 10%, some areas of the image showed better segmentation, while other regions exhibited oversegmentation. The oversegmentation issue, however, was primarily observed when CLAHE was applied. Despite this, denoising through the algorithm resulted in smoother boundaries and less jagged segmentation, which contributed positively to the overall segmentation quality. However, no significant differences in terms of segmented area were observed when compared to other methods.

One area for improvement in this approach would be to enhance the brightness and contrast conditions by incorporating additional image enhancement functions like CLAHE, gamma correction, or similar methods within the evolution process. Moreover, using a larger dataset for training, rather than relying on a single image, would likely lead to more robust performance and a better generalization of the algorithm.

Another important observation is that, despite being implemented in Python (rather than a more performance-optimized language like C/C++), the algorithm's computation times were relatively high. This is primarily due to the calculation of more complex terminal functions, such as MAD (Mean Absolute Deviation), quantile absolute deviation, and others. To address this, the algorithm's performance could be improved by using terminals that require faster calculations. For instance, instead of calculating the median directly from the pixel values, it could be computed using histogram values, which would allow for a linear calculation time (O(1)). A similar approach could be taken with the Gaussian filter, optimizing its implementation to reduce processing time.

Additionally, one potential improvement would be to replace computationally expensive functions like Quantile Absolute Deviation and MAD5 (Mean Absolute Deviation for 5 neighbors), which have higher computational complexity, with more efficient alternatives that can be calculated in linear time. This adjustment would help reduce the overall processing time, making the algorithm more efficient without sacrificing its performance.

We can conclude that while the genetic programming algorithm shows promise, further optimization in terms of dataset size, image enhancement integration, and computational efficiency would contribute to improved segmentation quality and faster execution times, making the approach more suitable for real-world applications.

References

Barrile, V., & Genovese, E. (2024). GIS-Like environments and HBIM integration for ancient villages management and dissemination. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 48, 41-47

Barrile, V., Cotroneo, F., Genovese, E., & Bilotta, G. (2023). Using Snn neural networks trained with high resolution data 2 and applied to copernicus SENTINEL-2 data. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 48, 27-31.

Barrile, V., Cotroneo, F., Genovese, E., Barrile, E., and Bilotta, G.: An AI Segmenter On Medical Imaging For Geomatics Applications Consisting Of A Two-State Pipeline, Snns Network And Watershed Algorithm, *Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci*., XLVIII-2/W3-2023, 21–26, https://doi.org/10.5194/isprs-archives-XLVIII-2-W3-2023-21-2023, 2023

Barrile, V., La Foresta, F., Calcagno, S., & Genovese, E. (2024). Innovative System for BIM/GIS Integration in the Context of Urban Sustainability. *Applied Sciences* (2076-3417), 14(19).

Bhardwaj, A., Kaur, S., Shukla, A. P., & Shukla, M. K. (2019, March). An enhanced cellular automata based filter for despeckling of ultrasound images. *In 2019 6th International Conference on Signal Processing and Integrated Networks (SPIN)* (pp. 1095-1099). IEEE.

Bhardwaj, A., Kaur, S., Shukla, M., & Shukla, A. P. (2021). A Despeckling Filter for Ultrasound Images Based on Cellular Automata Approach. *In Intelligent Manufacturing and Energy Sustainability: Proceedings of ICIMES 2021* (pp. 217-226). Singapore: Springer Singapore.. https://doi.org/10.1007/978-981-16-6482-3_22

Cortacero, K., McKenzie, B., Müller, S., Khazen, R., Lafouresse, F., Corsaut, G., ... & Cussat-Blanc, S. (2023). Evolutionary design of explainable algorithms for biomedical image segmentation. *Nature communications*, 14(1), 7112.. https://doi.org/10.1038/s41467-023-42664-x

Dhot, T. S. (2009). GPIS: genetic programming based image segmentation with applications to biomedical object detection (Doctoral dissertation, Concordia University).

Ezhilarasan, K., Somasundaram, K., Kalaiselvi, T., Somasundaram, P., Selvi, S. K., & Jeevarekha, A. (2024). Bio-Inspired Algorithms Used in Medical Image Processing. In *Bio-Inspired Optimization Techniques in Blockchain Systems* (pp. 25-60). IGI Global Scientific Publishing.

Fajardo-Delgado, D., Rodríguez-González, A. Y., Sandoval-Pérez, S., Molinar-Solís, J. E., & Sánchez-Cervantes, M. G. (2023). Genetic Programming to Remove Impulse Noise in Color Images. *Applied Sciences*, 14(1), 126.. https://doi.org/10.3390/app14010126

Intyanto, G. W., Apriyanto, R., Anam, K., Chaidir, A. R., Sarwono, C. S., & Eska, A. C. (2024, November). UGER: Enhancing the Quality of ROUV Image Segmentation with CLAHE and Color Space-Morphology Masking Techniques. *In 2024 IEEE 2nd International Conference on Electrical Engineering, Computer and Information Technology (ICEECIT)* (pp. 1-5). IEEE., doi: 10.1109/ICEECIT63698.2024.10859578

Human Heart Project. https://humanheart-project.creatis.insa-lyon.fr/database/#collection/6373703d73e9f0047faa1bc8 Accessed on 28 April 2025

Javed, S. G., Majid, A., & Lee, Y. S. (2018). Developing a bio-inspired multi-gene genetic programming based intelligent estimator to reduce speckle noise from ultrasound images. *Multimedia Tools and Applications*, 77, 15657-15675. https://doi.org/10.1007/s11042-017-5139-2

Khan, S. U., Ullah, N., Ahmed, I., Chai, W. Y., & Khan, A. (2018). MRI images enhancement using genetic programming based hybrid noise removal filter approach. *Current Medical Imaging Reviews*, 14(6), 867-873.

Korevaar, S., Tennakoon, R., & Bab-Hadiashar, A. (2024). Generalization Capabilities of Neural Cellular Automata for Medical Image Segmentation: A Robust and Lightweight Approach. *arXiv preprint arXiv:2408.15557*.

Poli, R. (1996, April). Genetic programming for feature detection and image segmentation. In AISB Workshop on Evolutionary Computing (pp. 110-125). Berlin, Heidelberg: Springer Berlin Heidelberg. https://doi.org/10.1007/BFb0032777

Kumar Rai, R., Gour, P., & Singh, B. (2012). Underwater image segmentation using clahe enhancement and thresholding. *International Journal of Emerging Technology and Advanced Engineering*, 2(1), 118-123.

Ramesh, N., Ashfaq, T., Kharma, N. (2024). A Hybrid Neuro-evolutionary Approach to the Design of Convolutional Neural Networks for 2D and 3D Medical Image Segmentation. In: Suen, C.Y., Krzyzak, A., Ravanelli, M., Trentin, E., Subakan, C., Nobile, N. (eds) *Artificial Neural Networks in Pattern Recognition. ANNPR 2024*. Lecture Notes in Computer Science(), vol 15154. Springer, Cham. https://doi.org/10.1007/978-3-031-71602-7_8

Sahu, S., & Singh, A. K. (2024). Genetic algorithm based multi-resolution approach for de-speckling OCT image. *Multimedia Tools and Applications*, 83(10), 31081-31102. https://doi.org/10.1007/s11042-023-16575-4.

Vanneschi, L., Poli, R. (2012). Genetic Programming — Introduction, Applications, Theory and Open Issues. In: Rozenberg, G., Bäck, T., Kok, J.N. (eds) Handbook of Natural Computing. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-540-92910-9_24