Multi-Level Image Segmentation via Particle Swarm Optimization

1. Introduction

Image segmentation, the process of partitioning an image into multiple meaningful regions, is a fundamental task in computer vision and medical image analysis. Multi-level thresholding is a common technique for image segmentation where multiple intensity thresholds are selected to separate an image into several classes. Finding optimal thresholds, especially for multiple levels, can be computationally expensive if an exhaustive search is performed. This project explores the application of Particle Swarm Optimization (PSO), a metaheuristic optimization algorithm, to automatically determine optimal thresholds for multi-level image segmentation of medical MRI images. The primary fitness function employed is Kapur's entropy, which aims to maximize the information content (entropy) of the segmented classes. The project also investigates the use of Otsu's between-class variance as an alternative fitness function and evaluates the segmentation results using the Structural Similarity Index (SSIM) against a provided ground truth.

2. Methodology

The core of this project involved implementing a PSO algorithm in Python to find optimal intensity thresholds for segmenting grayscale medical images.

a. Image Preprocessing

The input medical image (mri-stack.tif) was processed to extract a suitable grayscale representation for thresholding. For the primary experiments, the image was treated as a color image (as per the script's execution flag), and its **L* channel (from the CIELAB color space)** was extracted. This L* channel, representing perceptual lightness, was then normalized to an 8-bit grayscale image (intensity range [0, 255]) to serve as the input for the PSO algorithm. An example of extracting an alternative intensity channel (V channel from HSV) was also implemented (Figure 3).

b. Particle Swarm Optimization (PSO)

A standard PSO algorithm was implemented with the following key components:

i. **Particle Representation:** Each particle in the swarm represented a candidate set of N threshold values. The position of a particle was an array of N sorted integers.

ii. **Search Space:** Particles were initialized and constrained to search for threshold values within the actual content range of the input grayscale image (e.g., [1, 215] if the image content ranged from [0, 216].

iii. **Fitness Evaluation:**

1. **Kapur's Entropy (Primary):** The primary fitness function was Kapur's entropy. The goal was to find a set of thresholds that maximizes the sum of entropies of the individual classes formed by these thresholds. This function was corrected during development to accurately reflect its definition.

2. **Otsu's Between-Class Variance (Alternative):** As an extension, Otsu's method, which aims to maximize the variance between segmented classes, was also implemented as a fitness function.

iv. **PSO Parameters:**
The algorithm utilized standard PSO update rules for particle velocity and position, including an inertia weight (w) that linearly decreased over iterations, cognitive (c1) and social (c2) coefficients. A particle refresh strategy was also employed, where a small percentage of the worst-performing particles were re-initialized in each iteration to promote diversity. The specific parameters used were: Swarm Size = 40, Max Iterations = 100, c1=1.5, c2=1.5, w_start=0.9, w_end=0.4, Refresh Ratio = 10%.

v. **Termination:** The PSO process terminated after a fixed number of iterations.

c. Segmentation Application

Once the optimal thresholds were determined by PSO, they were applied to the input grayscale image to produce a segmented image. Pixels were assigned a new intensity value (typically the mean of the threshold boundaries defining their segment) based on which segment their original intensity fell into.

d. Comparison and Evaluation

i. **Skimage Otsu's Method:** The results from the PSO (using Kapur's entropy) were compared against the established multi-level Otsu's thresholding method provided by skimage.filters.threshold_multiotsu. The thresholds obtained by skimage.filters.threshold_multiotsu were also evaluated using the implemented Kapur's entropy function for a fair comparison of fitness scores.

ii. **SSIM:** The segmentation quality of the PSO output (using Kapur's entropy) was quantitatively evaluated using the Structural Similarity Index (SSIM) against a provided ground truth image.

3. Results and Discussion

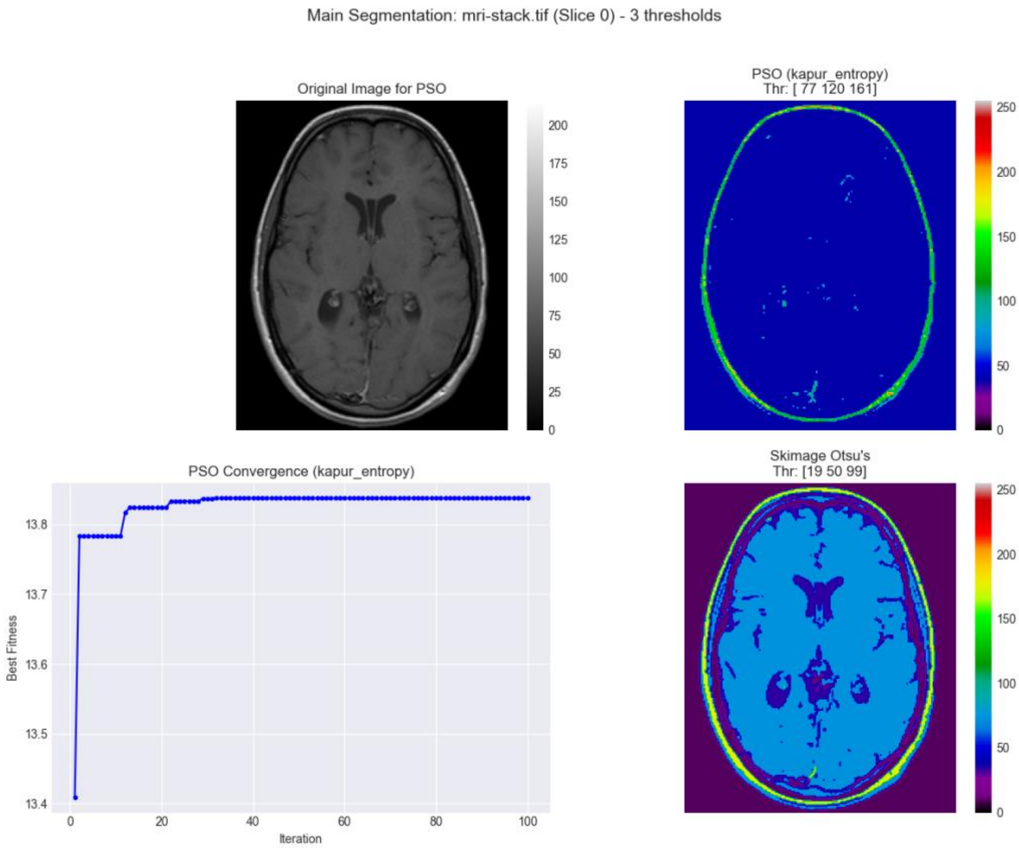
The implemented PSO framework was tested for NUM_THRESHOLDS = 3 on the selected MRI slice (processed L* channel as a grayscale image representing its intensity/lightness).

a. PSO with Kapur's Entropy

i. **Numerical Results:** The PSO algorithm converged to a maximum Kapur's entropy of 13.837618. The optimal thresholds found were [77, 120, 161]. The convergence plot (Figure 1, bottom-left) shows that the fitness improved rapidly within the first ~30-40 iterations and then stabilized. The PSO process took approximately 0.25 seconds.

ii. **Comparison with Skimage Otsu (Kapur's Criterion):** skimage.filters.threshold_multiotsu determined thresholds [19, 50, 99] for the same input image. When these thresholds were evaluated using the implemented Kapur's entropy function, the score was 11.675619. This indicates that the PSO successfully found a set of thresholds yielding a higher Kapur's entropy than the standard Otsu method, suggesting a better segmentation according to the Kapur's entropy criterion.

iii. **Visual Results (Figure 1):** Figure 1 (top-left) shows the original MRI slice (L* channel representation). Figure 1 (top-right) displays the segmentation achieved by PSO with Kapur's entropy. Using the nipy_spectral colormap, this segmentation highlights certain regions. Many pixels are mapped to the lower end of the colormap, corresponding to the first segment defined by the threshold 96. Figure 1 (bottom-right) shows the segmentation by skimage.filters.threshold_multiotsu. Visually, for this particular image and colormap, the Skimage Otsu segmentation appears to provide a more detailed separation of internal brain structures compared to the PSO (Kapur's) result in this figure. This highlights that maximizing a specific quantitative metric (like Kapur's entropy) does not always guarantee superior visual quality for all aspects of an image, or may prioritize different features.



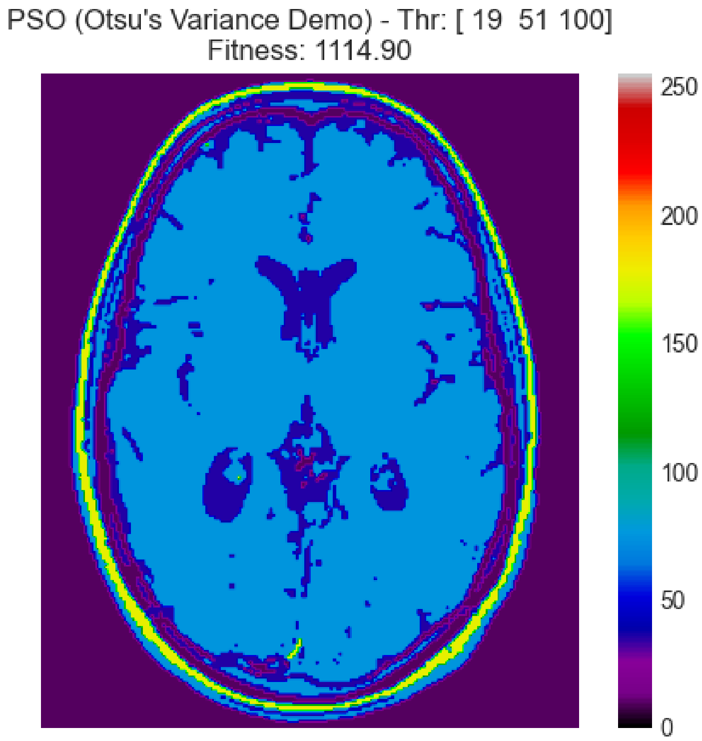
(Figure 1: Main output showing original image, PSO (Kapur's) segmentation, Skimage Otsu segmentation, and PSO convergence for Kapur's.)

b. PSO with Otsu's Between-Class Variance (Extension Demo)

i. **Numerical Results:** When the fitness function was changed to Otsu's between-class variance, the PSO converged to a maximum fitness of 1114.896362. The optimal thresholds found were [19, 51, 100]. This process took approximately 0.41 seconds.

ii. **Visual Results (Figure 2):** Figure 2 shows the segmented image produced by PSO when optimizing for Otsu's variance. These thresholds are remarkably similar to those found by skimage.filters.threshold_multiotsu ([19, 50, 99]). Visually, the segmentation in Figure 2 is almost identical to the Skimage Otsu's output shown in Figure 1 (bottom-right).

iii. **Significance:** This result serves as a strong validation of the implemented PSO framework. It demonstrates that the PSO can effectively optimize different fitness functions and converge to solutions consistent with established algorithms when targeting the same criterion.



(Figure 2: Segmentation result from PSO maximizing Otsu's between-class variance.)

c. Color Image Processing & Segmentation (Extension Demo)

i. **Astronaut Image Processing:** The skimage.data.astronaut() color image was used. Its L* channel was extracted and then segmented using PSO with Kapur's entropy (3 thresholds). The PSO found thresholds [78, 129, 186] for the astronaut's L* channel, achieving a Kapur's entropy of 15.23.

ii. **Significance:** This demonstrates the framework's adaptability. The same PSO core can be applied to different types of images (e.g., natural color images) by first extracting a suitable grayscale channel. Figure 3 specifically shows the V-channel extraction, while the segmentation was performed on the L-channel of the astronaut image.

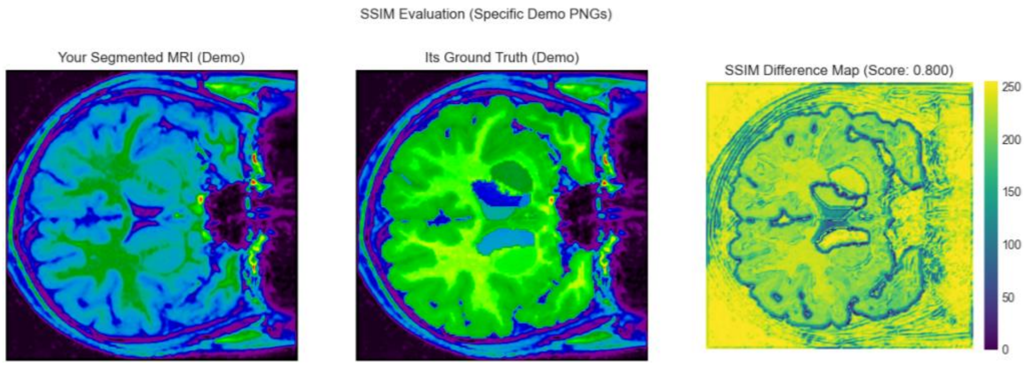
(Figure 3: L channel segmentation of the input color image.)

d. SSIM Evaluation Against Ground Truth (Extension Demo)

i. **Numerical Result:** For this demonstration, a user-provided segmented MRI PNG (my_segmented_mri_example.png) was compared against its corresponding ground truth PNG (gt_for_my_segmented_mri_example.png). The evaluation yielded an SSIM score of 0.7998.

ii. **Visual Results (Figure 4):** Figure 4 (left) shows the PSO-segmented image. Figure 4 (center) shows the Ground Truth image. Figure 4 (right) displays the SSIM difference map, highlighting areas of structural dissimilarity.

iii. **Discussion:** An SSIM score of ~0.8 indicates a good structural similarity. The difference map shows variations primarily within the internal brain structures. This score suggests that while the PSO (Kapur's) segmentation captures some broad features, its detailed structural representation (influenced by the Kapur's entropy objective and the chosen number of thresholds) differs to a fair extent from this specific ground truth. The way pixel values are assigned in the segmented image (mean intensity of segments) versus how the ground truth is labeled can also impact SSIM.



(Figure 4: SSIM evaluation of PSO (Kapur's) segmentation against a ground truth.)

4. Conclusion and Future Work

This project successfully demonstrated the application of Particle Swarm Optimization for multi-level thresholding of medical MRI images.

- The implemented PSO, using Kapur's entropy as a fitness function, was able to find thresholds that achieved a higher Kapur's entropy score than the standard Skimage Otsu's method.
- When Otsu's between-class variance was used as the fitness function, the PSO converged to thresholds remarkably similar to those produced by the established Skimage Otsu's algorithm, validating the PSO implementation's ability to optimize different criteria.
- The project also showcased extensions for handling color image channels and for quantitative evaluation using SSIM against a ground truth, yielding a moderate similarity score of 0.800.

The visual results indicate that while Kapur's entropy provides high scores, the resulting segmentation might not always align perfectly with visual expectations for delineating all anatomical structures, especially with a limited number of thresholds. The choice of fitness function is critical and application-dependent.

Future Work could include:

- **Exploring other fitness functions:** Investigating minimum cross-entropy or fuzzy entropy methods.
- **Implementing CIWP-PSO features:** Incorporating advanced PSO variants such as adaptive inertia weights or mutation operators to potentially improve convergence and escape local optima.
- **Spatial Considerations:** Extending the thresholding to include spatial information (e.g., 2D entropy or spatial constraints) rather than relying solely on the histogram.
- **Application to a larger dataset:** Testing the robustness and performance on a diverse set of medical images and comparing with clinical segmentations if available.
- **Adaptive determination of the number of thresholds:** Developing methods to automatically find the optimal number of thresholds rather than predefining it.

Overall, this project provides a solid foundation and implementation of PSO-based multi-level image segmentation, offering promising results and clear avenues for further development and refinement.