Region Growing Algorithm for Binary Images

Introduction

This report presents a comprehensive overview of the development and execution of a region growing algorithm, tailored specifically for the segmentation of binary images using an 8-connected neighborhood approach. Developed in Java, this project leverages the language's robust standard library, object-oriented programming capabilities, and its prevalent use in image processing, to effectively segment images into distinct regions based on pixel intensity values.

Methodology

The algorithm's core functionality revolves around the iterative expansion of regions from selected seed points, based on the intensity values of pixels. This process continues until no adjacent pixel can be added to any region, effectively segmenting the entire image based on the similarity of neighboring pixels.

Algorithm Steps:

Initialization: Begins with reading a PGM format image, extracting its dimensions, and populating a matrix with pixel intensity values.
Seed Point Selection: Iterates over the image matrix to select seed points with the specified target intensity value (either 0 or 255).
Region Growing: Expands a region from each seed point by adding adjacent pixels sharing the same intensity value, utilizing an 8-connected neighborhood.
Termination and Output: Continues until all pixels are evaluated, then outputs the number of detected regions and their sizes in ascending order.

Design and Implementation Decisions

Data Structures and Representation

- **Image and Visited Matrices**: The decision to use two-dimensional arrays for both the image (`imageMatrix`) and a corresponding visited status matrix (`isVisited`)

provides a direct and efficient means to access and modify pixel values and track processed pixels.

Algorithmic Strategy

- **8-Connected Neighborhood**: Opting for an 8-connected neighborhood ensures comprehensive region connectivity, enhancing segmentation accuracy and visual coherence in binary images.
- **Queue-Based Breadth-First Search (BFS)**: The BFS implementation, using a queue, allows for structured pixel exploration, ensuring all adjacent pixels are considered for region inclusion systematically.

Detailed Method Insights

`readImage(String filePath)`

- **Purpose**: Reads and validates the PGM image format, initializing image matrices.
- **Rationale**: Accommodating both standard and non-standard PGM formats ensures the algorithm's broad applicability, enhancing user experience.

`processImage()`

- **Purpose**: Orchestrates the region growing process, iterating over pixels to identify and grow regions.
- **Rationale**: Centralizing control flow enhances readability and efficiency, ensuring comprehensive pixel evaluation.

`exploreRegion(int startY, int startX)`

- **Purpose**: Grows regions using a BFS approach, adding adjacent pixels with matching intensity values.
- **Rationale**: BFS ensures orderly and exhaustive region expansion, leveraging queue mechanics for efficient exploration.

`isValid(int y, int x)`

- **Purpose**: Validates pixel coordinates within image boundaries.
- **Rationale**: Centralizes boundary checking, simplifying modifications and enhancing code maintainability.

`printResults()`

- **Purpose**: Outputs sorted region sizes, providing clear segmentation results.

- **Rationale**: Sorting region sizes before outputting offers a user-friendly summary, aiding in result verification and analysis.

Conclusion

The development of the region growing algorithm for binary images presented a series of challenges, notably in handling edge cases and optimizing performance for larger images. Through iterative testing, research, and refinement, these obstacles were navigated, leading to the creation of a performant, reliable, and user-friendly tool. This journey underscored the delicate balance between theoretical algorithm design and practical application, emphasizing the importance of adaptability and rigorous evaluation. The project's success lies not only in the effective segmentation of binary images but also in the valuable insights gained regarding the complexities of image processing and the nuances of algorithm optimization.

Author’s Information

Name: Visho Malla Oli
Sudent ID: 10915723
Course ID: CSCI 433 Project 1
Submission Date: Feb 8, 2024