**Objective**

Develop an image segmentation tool that separates the foreground from the background in grayscale images using the Max Flow / Min Cut Theorem, formulated as a linear programming (LP) problem. The project simulates a real-world scenario where the output will be evaluated as a potential feature for image editing software.

**Tasks**

1. **Network Creation**
   * Convert a grayscale image into a network where:
     + Pixels are represented as nodes.
     + Similarities between neighboring pixels form the connection weights.
   * Extend the network to include a source and sink node connected to a background and foreground pixel, respectively.
   * Use a similarity function to compute weights, experimenting with σ\sigmaσ values to achieve the best results.
2. **Formulate the Linear Program**
   * Define decision variables for all non-zero connections in the network matrix.
   * Set flow constraints to ensure flow conservation at nodes.
   * Maximize the flow from the source to the sink node.
3. **Identify Optimal Cuts**
   * Generate the residual network using the optimal flow solution.
   * Apply a depth-first search to identify accessible and inaccessible nodes.
   * Determine and sum the weights of cut edges, verifying their relation to the LP objective value.
4. **Implementation**
   * Write clear Python code to:
     + Load a CSV file representing pixel intensities.
     + Define background and foreground pixels.
     + Solve the LP problem and visualize cuts on the image using tools like Matplotlib.
5. **Analysis**
   * Provide a detailed report with:
     + Steps taken to solve the problem.
     + Visualizations of segmentation results.
     + Examples of different test images, including the provided and self-created images.