Image Segmentation With a Multiobjective Evolutionary Algorithm

Deadline: May 3rd, 2022 (Tuesday) at 08:00 AM.

Project goals

There are different project goals depending on the option chosen by the student(s)

- Implement a multiobjective evolutionary algorithm (EA), either:
    - **Option 1**: Implement a multiobjective GA *or*
    - **Option 2**: Implement a multiobjective ACO.
- Compare the performance of your implemented MOEA on several benchmark problems.
- Test and analyze the effects of MOEA(s) in optimizing multiple objectives simultaneously.

Groups Allowed

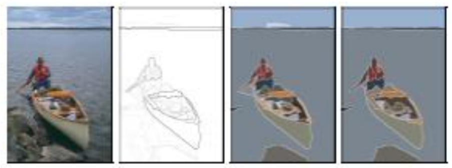
You may work alone or in groups of two. We accommodate both groups of one or two, but recommend groups of two, given the workload of this project.

Assignment Details

Image segmentation is a fundamental process in many image, video and computer vision applications. The main goal is to partition an image into separate regions of pixels, which ideally correspond to different real-world objects. A pixel is defined as the smallest addressable element in an image, while a segment is a set of pixels, where each pixel is a neighbor of at least one other pixel in the set. The union of all segments comprises the full image. Figure 1 illustrates examples of different segmentations on the images given.

(a)

(b)



(c)

Figure 1: Segmentation examples

The concept of **nearest neighbors** is important in image segmentation. In this project we will consider the 8 nearest neighbors to a pixel. Note that the nearest neighbors are numbered according to the 4 nearest neighbors followed by the corner neighbors. Further, the pixels are numbered according to East-West-North- South, as illustrated in Figure 2.

7	3	5
2	P	1
8	4	6

Figure 2: A Pixel's 8-nearest neighbors (Moore neighborhood)

The concept of **similarity** is another important concept in image segmentation. Pixels within a region are similar according to some uniformity predicate and pixels in neighboring regions are dissimilar. Such regions are termed segments.

Segments have two properties:

1. Homogeneity within a segment: the texture, color, or intensity of each pixel in the segment should be ‘similar’ to the other pixels in the same segment.
2. Heterogeneity between neighboring segments: the texture, color or intensity of the pixels in one segment should be distinct from the pixels in the neighboring segment.

There are various types of segmentation. Figure 3 illustrates the two segmentation types that you will be implementing in this project: Type 1 and Type 2.

(a) Original image (b) Type 1 Segmentation (c) Type 2 Segmentation

Figure 3: Image with Type 1 and Type 2 segmentation

Figure 4 presents a further set of examples of image segmentation, each of Type 1. As shown, the same image may be partitioned in many ways resulting in a different number of segments. The image shown and the image examples are taken from the *Berkeley Segmentation Dataset and Benchmark* which contains 1000 original examples. Each original image in the dataset is associated with multiple human-traced segmentation solutions of that image **ground-truth segmentations (GTSs**) for each segmentation type. Each of the GTSs associated with a given image are considered reasonable, i.e., all four solutions shown in Figure 4 are equally good.

(a) 24 segments (b) 106 segments

(c) 54 segments (d) 42 segments

Figure 4: Different "good" segmentations of Type 1

Problem Formulation

The key goal of this project is the application of MOEAs to color image segmentation.
The pixels of an image may be stored by their corresponding color values from a color space [1]: RGB or CIE L*a*b .The first step is to define the genotype (chromosome) for the evolutionary process, together with a conversion from the genotype to the phenotype (image). You are free to choose a representation from existing representations or propose a new one. However, note that care should be given to choose an effective representation. Commonly applied representations include graph-based or tree-based representations, where you can either store all the pixels or only representatives of the segments.

We define the set of all segments C and denote a specific segment as C_k. Hence, $C_k \in C$. Pixels are indexed $i \in \{1, \ldots, N\}$ where N is the number of pixels in the image. All pixels are assigned to a segment, i.e. $\exists C_k : i \in C_k$. Each segment's average pixel color value is denoted the centroid, μ_k for segment C_k. The set of a pixel's neighbors is denoted F for pixel i.

You need to simultaneously optimize several objectives (segmentation criteria) using your chosen MOEA. The objectives are *edge value*, *connectivity* and *overall deviation*. We now define and discuss these objectives.

1. The first objective, the **edge value**, is a measure of how clear the boundaries of the segments in the image are. A clear boundary marks image locations of discontinuity in gray or color levels. It is subject to **maximization** and is defined as:

$$\text{Edge value} := \sum_{i=1}^{N}\left(\sum_{j \in F_i} x_{i,j}\right)$$

$$\text{where} \quad x_{i,j} = \begin{cases} dist(i,j) & if \nexists C_k : i,j \in C_k \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

   The distance function *dist()* should be defined as the **Euclidean distance** by using your choice of RGB or CIE L*a*b* color space. The distance functions using the RGB and CIE L*a*b* color spaces are defined in Equations 2 and 3, respectively.

$$dist_{RGB}(i,j) = \sqrt{\Delta R_{i,j}^2 + \Delta G_{i,j}^2 + \Delta B_{i,j}^2} \tag{2}$$

$$dist_{L^*a^*b^*}(i,j) = \sqrt{\Delta L^{*2}_{i,j} + \Delta a^{*2}_{i,j} + \Delta b^{*2}_{i,j}} \tag{3}$$

   The edge value objective strives to draw the line between two segments such that the pixels differing the most in color belong to different segments.

2. The second objective, the **connectivity measure**, evaluates the degree to which neighboring pixels have been placed in the same segment, as defined in Equation 4. It is subject to **minimization** and is defined as:

$$\text{Connectivity} \coloneqq \sum_{i=1}^{N}\left(\sum_{j \in F_i} x_{i,j}\right),$$

$$\text{where } x_{i,j} = \begin{cases} \frac{1}{F_i(j)} & if \nexists C_k : i, j \in C_k \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

$F_i(j)$ returns which neighbors pixel j is to pixel i, using the number from Figure 2. For example, if pixel A is to the "east" of pixel B, then $F_B(A) = 2$. The weighting of the neighbor pixels will not be equal, but as long as you follow the same weighting pattern for all pixels your solution should not be affected by this difference. It is a common approach in image segmentation. If you want, you can replace *1/F (j)* with *1/8* to get the same weighting for the neighbor pixels.

3. The third objective, **overall deviation, is a** measure of the "similarity" (homogeneity) of pixels in the same segment, as defined in Equation 5. It expresses the compactness of segments, by providing the overall sum for all segments of the distance between every pixel and the center value of the corresponding segment they belong to.

$$\text{Overall-deviation} \coloneqq \sum_{C_k \in C} \sum_{i \in C_k} dist(i, \mu_k) \quad (5)$$

where C is the set of all segments, k is the centroid (the average pixel value) in the segment C_k and *dist()* is the distance function as described in the Equation 2 and 3. Overall deviation should be **minimized** to achieve compact segments.

By comparing the segmentations produced by the weighted-sum simple GA with the Pareto-optimal segmentation produced by the MOEA you will be able to analyze the effects of MOEA in optimizing multiple objectives in the context of image segmentation.

Evolutionary Algorithm (EA)

For this project students can choose *one* among two different options, as outlined below.

Option 1: GA-based multiobjective optimization
Both a MOEA (NSGA-II [recommended], SPEA or PAES) and a simple GA (with a weighted-sum fitness function) will be applied, and compared, on the task of optimizing the objectives given.

All students that choose **option 1** must implement the three objectives in the MOEA. In the weighted GA, the importance of each objective function should be chosen to balance each other's tendency to increase or decrease the number of segments. Edge value and overall-deviation improves with an increasing number of segments (reduced distance) whilst the opposite is the case for connectivity (avoids splitting segments where neighboring pixels would be in different segments).

The interaction of these objectives in your MOEA thus allows for exploration of the search space to look for Pareto-optimal solutions, i.e., a range of solutions, with different objective trade-offs, for each input test image.

Option 2 - ACO-based multiobjective optimization
Here you will study ant colony optimization (ACO) for the purpose of multiobjective optimization.

This option gives the opportunity to implement a new algorithm that has not been encountered in the earlier projects. If you choose option 2, you should be aware that you are more on your own, as the TAs don't have hands-on experience with the ACO approach. However, there are some papers written about it, for instance [4], [5], [6], and [9] which we would recommend you using as your basis.

For weighted sum computation, you can either use single-objective GA (like for option 1, see the discussion there) or single-objective ACO.

Evaluation Criterion

When evaluating the quality of a segmentation solution, the Probabilistic Rand Index (PRI) is applied. The PRI score is calculated based on the proposed segmented solution compared to all GTSs for that image. However, if you have implemented the MOEA, it results in a number of Pareto-optimal segmented solutions for a single test image. To assess the quality of the set of pareto-optimal solutions against the GTS for a given image, a modified PRI is applied. This modified PRI finds the best Pareto-optimal solution amongst your solutions based on its PRI value. If you have implemented the ACO, you can calculate the PRI for the best solution only.

The PRI value provides a score between [0,1], comparing the proposed segmentation solution against all the GTS for that image. The modified PRI provides a similar scoring. A score of 0 means no similarities with the GTS (over or under segmentation), whereas a score of 1 means identical segmentation (excellent segmentation). You can find more about PRI in an article by Unnikrishnan et al. [2].

The coding for the modified PRI is provided together with the test images and the benchmark modified PRI values. The PRI values are calculated based on Type 2 segmentation as shown in Figure 3c. Note that every pixel must belong to a segment, and hence be encapsulated by a black border in your Type 2 segmentation. ***Thus, a square outside border should be present in all your solutions if the image segmentation algorithm is implemented correctly.***

Things To Do

This project accounts for 30 points of the 100 points available for this course.

In this project, you have two options, as discussed above:

1. **Option 1**: You need to implement the segmentation approach for color image, applying all three objectives given and using any one of the Option 1 MOEAs mentioned above. Further, you need to implement a weighted-sum simple GA, applying these same three objectives. The paper "A Multi-Objective Evolutionary Algorithm for Color Image Segmentation " [3] contains some pointers on a possible implementation.
2. **Option 2**: You need to implement the multiobjective segmentation algorithm based on an Ant Colony Optimization approach, according to option 2. The papers "Multiple objective ant colony optimisation. Swarm intelligence" [4], "Use of aggregation pheromone density for image segmentation"[5], "Population-based ant colony optimisation for multi-objective function optimisation"[6], and "MHACO: a multi-objective hypervolume-based ant colony optimizer for space trajectory optimization" [9] contain some pointers on a possible implementation. Further, you need to implement a single-objective EA, either GA or ACO.

In addition, the papers "Evolutionary image segmentation based on multiobjective clustering" [7] and "Image segmentation using a genetic algorithm and hierarchical local search" [8] can prove valuable, as they give some general ideas regarding image segmentation.

Your implementation needs to produce two types of segmentation solutions for each test image given: Type 1 and Type 2, as illustrated in Figure 3. Note that each type needs to be produced for each member of the set of Pareto optimal solutions for that image.

To test your code, we have uploaded a number of benchmark images, their ground-truth segmentation solutions, and the corresponding best PRI values.

All students in a group are required to attend the demo day, i.e., both members of a group of two students.

At the demo day you will show us the running code to verify that it works. Each student needs to be able to describe how you designed and implemented your MOEA and your weighted-sum EA. You will be supplied with 3 test images to test your code during your demo. We will provide you the lowest and highest number of possible segmentations for each test image.

You must run your code and show us all the requirements (including all explanations) **within 30 (thirty) minutes**.

The point distribution for the demo is as follows:

1. **Test images (3x5 = 15p)**
   You need to optimize three objectives simultaneously for three test images.
   For each test image:
   PRI values for a group of two students (single student groups)
   PRI value greater than 0.75 (0.72): 5 points.
   PRI value greater than 0.70 (0.67): 4 points.

PRI value greater than 0.60 (0.57): 3 points.
Otherwise, 0 points.

The above points are for Option 1. For Option 2, PRI values are reduced by 0.05 for each value. That is, 5 points are given for PRI greater than 0.7 (0.67), etc.

2. **Show solutions (2p per image, 6p total)**
   For all the test images, show the segmented solutions along with the respective objective values and segmentation number for all the Pareto optimal individuals of the final population (final segmentations). If the final Pareto optimal solutions are more than 5, you can show any 5 of them. For each of the presented segmented solutions, you need to show both types (Figure 3b and 3c) of segmentation for the same solution.

3. **Explanation (9p)**
   You will be provided with a fourth image that needs to be run using the weighted-sum EA as well as your implemented MOEA. Based on the achieved PRI values, you need to be ready to explain the reason behind the differences in segmentation quality obtained by MOEA and weighted-sum method. You will also be asked to explain the concept of a Pareto-front and -optimal solutions in the context of multi-objective optimization for image segmentation.

Delivery and demo
The demo day is Monday April 26 2021. Every student must deliver a zip file of your joint code on Blackboard. The submission system will be closed at 08:00 AM on the demo day.

Every student must sign up and attend the demo day. No early or late demo will be entertained.

NB:
1. Every question regarding this project will only be handled through the Forum or at the Lab hours.
2. Playing with images can be memory and compute intensive. Therefore, take care when making your choice of programming language and data structures. In the past, the programming languages C++ and Java have been used with success.

References

[1] Gunther Wyszecki and W. S. Stiles. Color Science: Concepts and Methods, Quantitative Data and Formulae, 2nd Edition. Wiley Interscience, 2000.

[2] Unnikrishnan, Ranjith, Caroline Pantofaru, and Martial Hebert. "Toward objective evaluation of image segmentation algorithms." IEEE transactions on pattern analysis and machine intelligence 29.6 (2007): 929-944.

[3] Ripon, Kazi Shah Nawaz & Ali, Lasker & Newaz, Sarfaraz & Ma, Jinwen. (2017). A Multi-objective Evolutionary Algorithm for Color Image Segmentation. 10.1007/978-3-319-71928-3_17.

[4] Angus, D., & Woodward, C. (2009). Multiple objective ant colony optimisation. Swarm intelligence, 3(1), 69-85.

[5] Ghosh, S., Kothari, M., Halder, A., & Ghosh, A. (2009). Use of aggregation pheromone density for image segmentation. Pattern Recognition Letters, 30(10), 939-949.

[6] Angus, D. (2007, December). Population-based ant colony optimisation for multi-objective function optimisation. In Australian Conference on Artificial Life (pp. 232-244). Springer, Berlin, Heidelberg.

[7] Shirakawa, S., & Nagao, T. (2009, May). Evolutionary image segmentation based on multiobjective clustering. In 2009 IEEE congress on evolutionary computation (pp. 2466-2473). IEEE.

[8] Hauschild, M., Bhatia, S., & Pelikan, M. (2012, July). Image segmentation using a genetic algorithm and hierarchical local search. In Proceedings of the 14th annual conference on Genetic and evolutionary computation (pp. 633-640).

[9] Acciarini, G., Izzo, D., & Mooij, E. (2020, July). MHACO: a multi-objective hypervolume-based ant colony optimizer for space trajectory optimization. In 2020 IEEE Congress on Evolutionary Computation (CEC) (pp. 1-8).