Image Segmentation using a Genetic Algorithm and Hierarchical Local Search

Mark W. Hauschild, Sanjiv Bhatia, and Martin Pelikan

Missouri Estimation of Distribution Algorithms Laboratory (MEDAL)
Department of Mathematics and Computer Science
University of Missouri, St. Louis, MO

martin@martinpelikan.net
http://medal-lab.org

Motivation

- Image segmentation important in image processing.
  - Often used as a preliminary step.
  - Can emphasize boundaries between objects.
  - Locate distinct objects in images.
- Image segmentation is challenging
  - Difficult to succeed across diverse images.
  - Even a relatively small image has a large number of points.
  - Large number of locally optimal segmentations.
  - Ideally want it done automatically.
- The purpose of this paper
  - Use Potts spin glass model to define image segmentation as optimization problem.
  - Use a steady-state genetic algorithm with hierarchical local search to solve the problem.

Outline

- Image segmentation as Potts spin glass.
- Algorithm
  - Steady-state genetic algorithm.
  - Crossover operator.
  - Hierarchical local search.
- Experiments.
- Conclusions and future work.

q-state Potts Model

- Origin in physics.
- Spins arranged on a 2D or 3D lattice.
- Each spin s_j can have up to q states.
- Edges have weights that define relationships between neighboring spins.
- Energy defined as

$$E = -\sum_{\langle i,j \rangle} J_{i,j}\delta(s_i, s_j) \ ,$$

  where $\delta(s_i, s_j) = 1$ if $s_i = s_j$, and $\delta(s_i, s_j) = 0$ if $s_i \neq s_j$.
- Usual task is to minimize the total energy of the system.
- To minimize energy, neighboring spins s_i and s_j should be equal if $J_{i,j} > 0$; otherwise the spins should be different (ideally). Larger $J_{i,j}$ values are more important than smaller.

Mapping Image Segmentation to Potts model

- Each pixel corresponds to one spin.
- The value of the spin defines the pixel's segment.
- Map grayscale image to a set of couplings $\{J_{i,j}\}$ between neighboring pixels:

$$J_{i,j} = 1 - \frac{\Delta_{i,j}}{\theta \Delta_{avg}},$$

where

  - $\Delta_{i,j}$ is absolute difference between neighbors,
  - Δ_{avg} is average difference between all neighbors in the image,
  - $\theta = 1$ controls sensitivity to changes in intensity.

- Small differences between neighbors imply a large coupling (meaning that the neighbors should be in the same segment).
- Large differences between neighbors imply a small coupling (meaning that the neighbors should be in different segments).

Mapping Image Segmentation to Potts model

- Image segmentation using Potts model
  - Number q of states in Potts model = number of segments.
  - Each pixel's state defines its segment.
  - Minimizing energy corresponds to segmentation so that similar neighbors belong to the same segment (and dissimilar neighbors belong to different segments).
- Can apply any optimizer (e.g. simple genetic algorithm).
- But naive implementations won't work too well
  - High dimensionality.
  - Large number of local optima.

Genetic Algorithm

- Representation
  - Segmentations represented by strings.
  - One character for each pixel in the image (long strings).
  - Each character can take q values (q=number of segments).
- Steady-state hybrid GA with small population.
  - Initial population of strings represents random segmentations.
  - Each iteration selects two random parents from population.
  - Parents are combined using crossover to create new string.
  - Local search used to improve the resulting string.
  - Candidate replaces the worst parent if it's better.

Algorithm

- Two details left to discuss:
  - Crossover operator.
  - Local search and hierarchical local search.

Crossover Operator

- Creates new segmentation c from segmentations a and b.
- Select randomly rectangular region.
- c takes content of rectangular region from a.
- c takes the rest from b.
- Selecting rectangular region
  - Picks a random pixel for center of region to swap.
  - Size of swap region for image of size $x \times y$ is $\sqrt{x/2} \times \sqrt{y/2}$.
  - Other scenarios possible.

Transforming Solutions for Crossover

- Similar segmentations have often different strings (renaming segments produces extremely different strings).

Image a

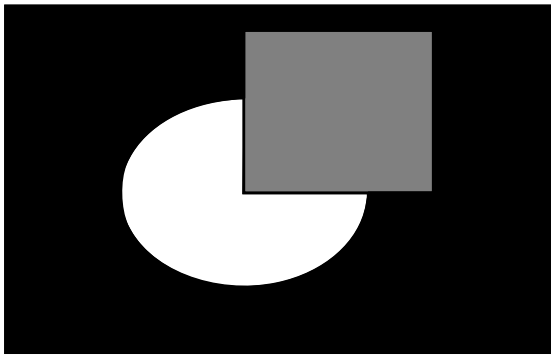
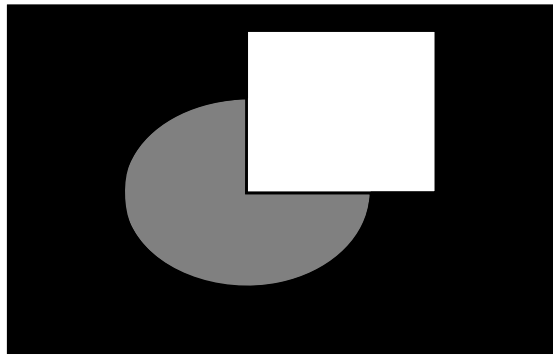


Image b



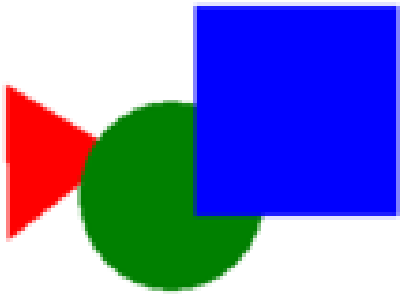
- Crossover without transforming parents leads to poor results because of excessive disruption and ineffective juxtaposition.
- Resolution: Transform parents by renumbering segments
  - Calculate conditional probabilities that a segment i in parent a is represented by segment j in parent b for all i and j.
  - Use greedy algorithm to renumber each segment in parent a to the most likely segment it represents in b.

Difficulties with Simple Local Search

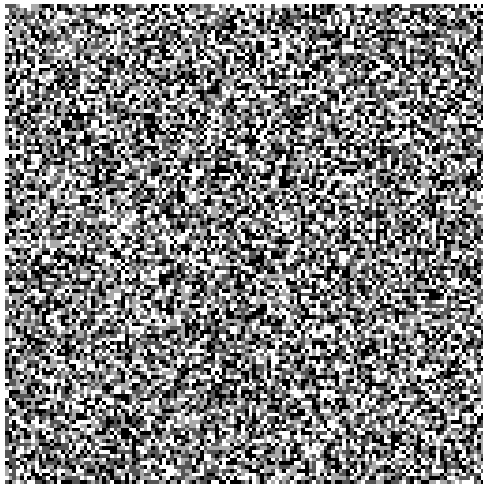
- Problems have large number of variables in each solution.
- Advantageous to use local search to speed optimization up.
- First attempted to use bit-flip hill climbing (HC)
  - Would get stuck at poor local optimum, even for images where each segment consisted of a single color (easy to segment).
  - Led to actual degradation of performance.

Difficulties with Simple Local Search: Illustration

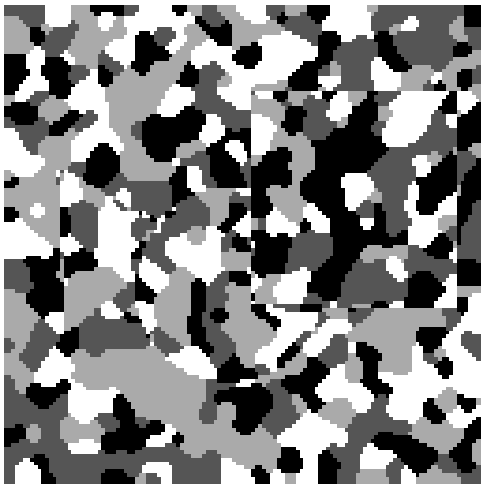
Initial image



Random segmentation



Segmentation after bit-flip HC

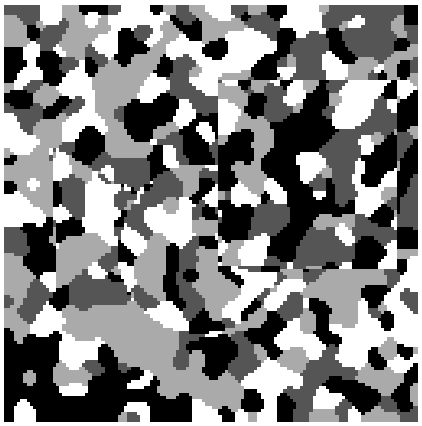


Hierarchical Local Search

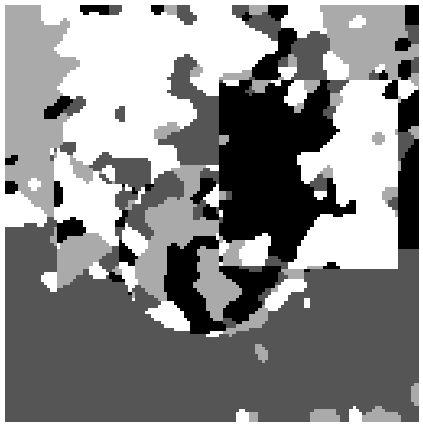
- Developed hierarchical local search to tackle this challenge.
- Start with simple bit-flip hill climbing.
- Treat each connected segment as one variable (character).
- Each iteration
  - Try to change each connected segment (connected region of pixels assigned 1 segment number) to another color.
  - Accept the best change (if improvement).
  - Terminate when no more improvement possible.
- Regions increase in size (regions merge over time).
- Implementation can be done efficiently.
- Important: Deals with the problem of simple HC, leading to a much more efficient search for good segmentations.

Steps of Hierarchical Local Search

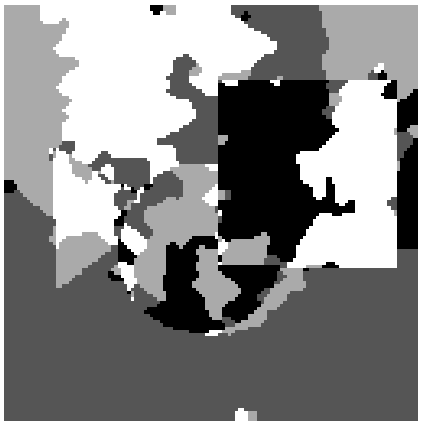
$Step = 1$



$Step = 50$



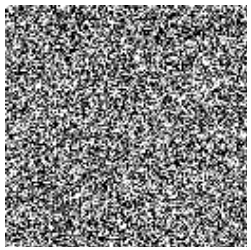
$Step = 100$



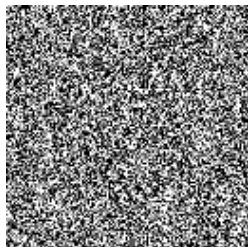
$Step = 143$

Demonstration of Importance of Hierarchical LS

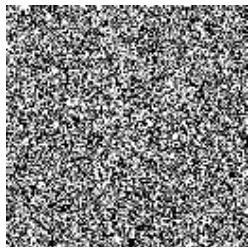
$g = 1$, no LS



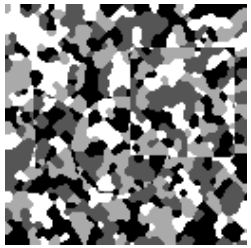
$g = 100$, no LS



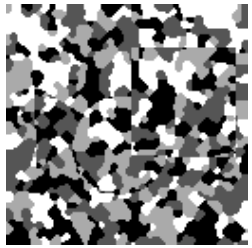
$g = 1000$, no LS



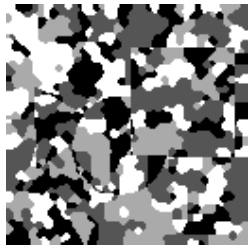
$g = 1$, DHC



$g = 100$, DHC



$g = 1000$, DHC



$g = 1$, HLS

$g = 100$, HLS

$g = 1000$, HLS

Experimental Setup

- Number of segments set to $q = 4$.
- Population size set to $N = 50$.
- Input images converted to 8-bit grayscale for 256 distinct gray levels before generating the weight matrix.
- Two different digital images were examined
  - House image of size 150×100 pixels.
  - Dog image of size 160×128 pixels.
- For each image:
  - Find final resulting segmented image using hybrid GA.
  - Examine image after coloring regions based on average color of that region in original image.
  - Compare the results to meanshift segmentation.

House Image using Hybrid GA

Initial image

Segmented image

Average color

House Image using Mean-Shift Segmentation

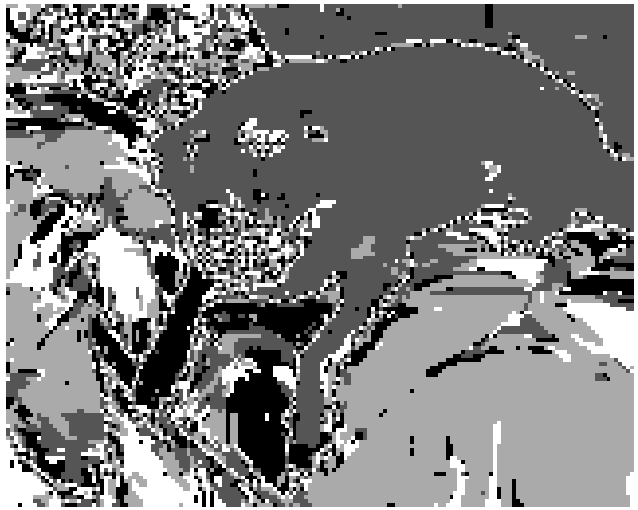
Intermediate

Final, 17 colors

Dog Image using Hybrid GA

Initial image

Segmented image



Average color

Dog Image using Mean-Shift Segmentation

Intermediate

Final, 26 colors

Conclusions

- Described a hybrid GA to perform image segmentation.
- Hierarchical local search proposed to improve efficiency.
- Transformation used to lower disruption due to crossover.
- Resulting hybrid GA was able to efficiently segment images.
- Results also show the necessity of hierarchical local search where simple bit-flip local search actually led to performance degradation.

Acknowledgments

- Support was provided by
  - NSF grants ECS-0547013 and IIS-1115352.
  - ITS at the University of Missouri in St. Louis.
  - University of Missouri Bioinformatics Consortium.