

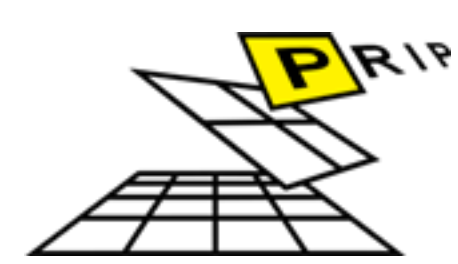
Poster 5



Hierarchical Interactive Image Segmentation using Irregular Pyramids

Michael Gerstmayer, Yll Haxhimusa, Walter G. Kropatsch

Pattern Recognition and Image Processing Group
Institute of Computer Graphics and Algorithms
Vienna University of Technology



Poster 5

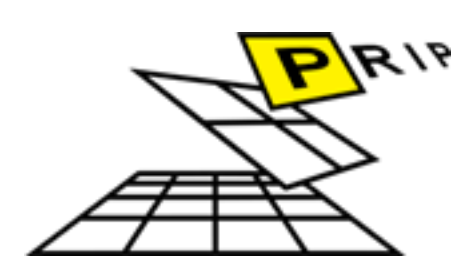


Introduction

- Stack of Segmentations (2D Image)
 - Minimum Spanning Tree based based Method [1]
 - Irregular Graph Pyramid
 - Merging Criteria
- Good but non satisfying result
- Interaction > State-of-the-art [2,3]



[1] Haxhimusa, Y. (2004); [2] Klava, B. (2009); [3] Meine, H. (2004)



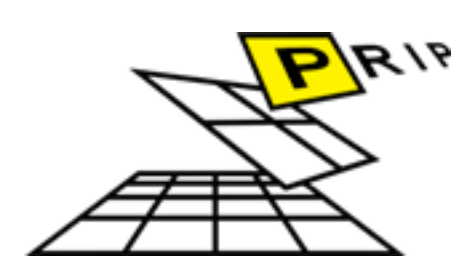
Poster 5



User Interaction

- Set focus on arbitrary regions
- Different levels of granularity
- Merge & inhibit operations
- Guide the process of split & merge
- Can be used for:
 - Annotation
 - Accuracy



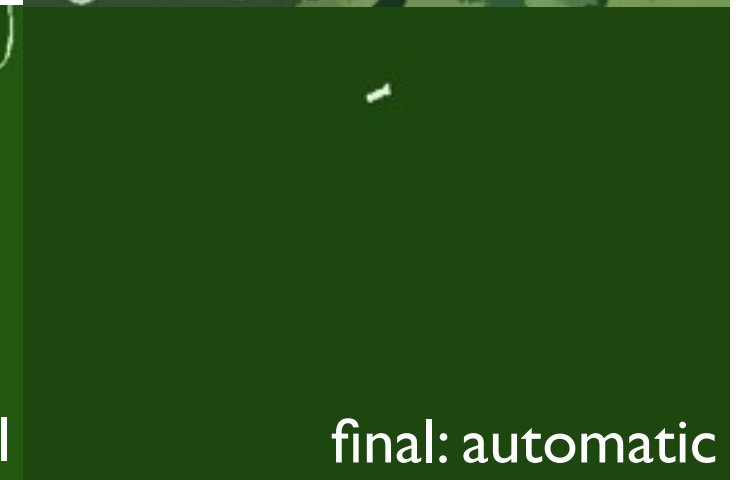


Poster 5



User Interaction

- Set focus on arbitrary regions
- Different levels of granularity
- Merge & inhibit operations
- Guide the process of split & merge
- Can be used for:
 - Annotation
 - Accuracy





Poster 5



Hierarchical Interactive Image Segmentation using Irregular Pyramids

- Interactive Demo
- Key ideas

<http://prip.tuwien.ac.at>

Hierarchical Interactive Image Segmentation using Irregular Pyramids*
Michael Genzmayer, Yi Hachimura, Walter G. Kropatsch
img.prip@tuwien.ac.at

Abstract
In this work modifications of irregular image segmentation pyramids based on user interaction are described. The first full hierarchy of segmentations is the minimum spanning tree based method [1]. For regions from different granularity levels we continue to a final desired segmentation with user specified operations joining the segmentation classes. Based on these operations the user can produce a final image segmentation that best suits their application.

Hierarchy of Partitions
Figure 1: Graph Representations of 2D Image Partitions. The figure shows a grid of pixels, a graph with nodes and edges, and a final segmented image.

Representation of Image/Topological Relations within an IR Image (Figure 1, 2)
• Instance Graphs (IG) or User Graphs (UserEdges, Adjacency & Boundary Graph)
• Octree Water Containment Maps (CWM) and reverse functions (containing IG)
• Irregular Graph Pyramid (IGP) (Figure 1b left)
• Stack of Segmentations
• Image Partitions at different resolution levels
• Hierarchy built by Construction/Removal Operations
• Dual Graph Construction (DG) (reverse function)
• Segmentation Framework
• Automatic process
• Minimum Spanning Tree based Method [1]
• Weighted binary partitioning grouping (containing IG)
• Hierarchical Topological Data Structure
• 2D Construction Map Pyramid

Interactive Operations
The tool developed consists of two parts: a GUI for input and the modified segmentation framework [2]. To guide the process and modify the resulting tree one can apply the following interactive split and merge operations on regions throughout the hierarchy:
Reg: a, b = arbitrary pyramid levels, a, b = Region i adjacent to b_j = Region j
split: $a, b, c, d, e, f, g, h, i, j, k, l, m, n, o, p, q, r, s, t, u, v, w, x, y, z$
merge: $a, b, c, d, e, f, g, h, i, j, k, l, m, n, o, p, q, r, s, t, u, v, w, x, y, z$
removal: $a, b, c, d, e, f, g, h, i, j, k, l, m, n, o, p, q, r, s, t, u, v, w, x, y, z$
addition: $a, b, c, d, e, f, g, h, i, j, k, l, m, n, o, p, q, r, s, t, u, v, w, x, y, z$
The user can be used to locally select regions by dragging the mouse instead of mouse selection by clicking. In the Capture mode one can apply the operations described before within a region (surrounded red rectangle) first inside of the boundary.

Executing Interactive Operations
The approach strongly benefits from the implicit Dual Representation. Operations are described using regions whereas the processing is done in graph-vertices (vertices):
 $R_{i+1} = R_i \cup R_j$
 $R_{i+1} = R_i \setminus R_j$
Base for $R_{i+1} = \{R_i, R_j\}$
Base for $R_{i+1} = \{R_i, R_j, R_k\}$
Figure 3 left: Map of level $i+1$, right: level i , relations between the levels (left: split, right: merge).
If operations are defined in multiple levels, we need a common starting level.
Multi-operative processing steps: Calculation of affected edges using
• Reevaluation with common starting level (a) • Construction Map
• Conflict handling
• Prerequisite condition operations to higher levels
Edge Merging: Edges to which vertices
• Invert
• Invert Edges (Figure 3 left)
• Reevaluate Edges (Reevaluation, Figure 3 right) • Set intersection
Figure 4: Reevaluation of affected elements, from left: image (a), (b) in level $i+1$ to (c).

Sample Results
Four images: without operations (middle) using interaction in split/merge on
Image 1: A landscape with a blue sky and green fields. Image 2: A landscape with a blue sky and green fields. Image 3: A landscape with a blue sky and green fields. Image 4: A landscape with a blue sky and green fields.

| Image | Level | Op. | Time | Size | Time | Size |
|-------|-------|-------|----------|------|----------|------|
| 1 | 1 | split | 0.000000 | 1024 | 0.000000 | 1024 |
| 1 | 2 | split | 0.000000 | 256 | 0.000000 | 256 |
| 1 | 3 | split | 0.000000 | 64 | 0.000000 | 64 |
| 1 | 4 | split | 0.000000 | 16 | 0.000000 | 16 |
| 1 | 5 | split | 0.000000 | 4 | 0.000000 | 4 |
| 1 | 6 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 7 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 8 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 9 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 10 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 11 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 12 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 13 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 14 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 15 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 16 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 17 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 18 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 19 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 20 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 21 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 22 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 23 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 24 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 25 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 26 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 27 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 28 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 29 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 30 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 31 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 32 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 33 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 34 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 35 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 36 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 37 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 38 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 39 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 40 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 41 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 42 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 43 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 44 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 45 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 46 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 47 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 48 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 49 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 50 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 51 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 52 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 53 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 54 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 55 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 56 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 57 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 58 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 59 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 60 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 61 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 62 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 63 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 64 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 65 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 66 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 67 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 68 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 69 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 70 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 71 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 72 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 73 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 74 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 75 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 76 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 77 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 78 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 79 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 80 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 81 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 82 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 83 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 84 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 85 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 86 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 87 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 88 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 89 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 90 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 91 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 92 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 93 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 94 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 95 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 96 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 97 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 98 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 99 | split | 0.000000 | 1 | 0.000000 | 1 |
| 1 | 100 | split | 0.000000 | 1 | 0.000000 | 1 |

Applications
• For hierarchical visualization of images
• Accuracy in image segmentation or creating ground truth
• Easy to create segmentations at different granularities compared to relative [2]
• Performance comparison of existing segmentations to the implemented
• Download framework at <http://prip.tuwien.ac.at>

Conclusion
• Interactive control on information from different granularities levels
• Operations influence but don't modify the framework
• Dual Representation is a strong benefit

References
[1] Felzenszwalb, Y., Krupar, W.G. Segmentation graph hierarchies. In: SPIE/ISIP, pp. 263-267 (2004)
[2] Felzenszwalb, Y., Krupar, W.G. Segmentation graph hierarchies: a comparison with conventional graph-based maps. Computer-Aided Design, 37(11):81-85, 2005
[3] Felzenszwalb, Y., Krupar, W.G. Segmentation graph hierarchies: a comparison with conventional graph-based maps. Pattern Recognition Letters 26(8): 1071-1077 (2005)
[4] Felzenszwalb, Y., Krupar, W.G. Segmentation graph hierarchies: a comparison with conventional graph-based maps. International Journal of Computer Vision 58(2): 107-121 (2004)
[5] Felzenszwalb, Y., Krupar, W.G., Laffont, J. Online image annotation and evaluation. Proceedings of the IEEE 19(9): 1467-1474 (2009)
[6] Felzenszwalb, Y., Krupar, W.G., Laffont, J. A database of human segmented object images and its application to evaluating segmentation algorithms and measuring object statistics. In: Proc. 8th Int'l Conf. Computer Vision, vol. 4, pp. 418-423 July 2001

*This paper has been supported by the Austrian Science Fund FWF (S1101-N13)