OBJECT-ORIENTED HIERARCHICAL IMAGE VECTORIZATION

A. N. Skurikhin [a, *], P. L. Volegov [b]

[a] MS D436, Space and Remote Sensing Group, Los Alamos National Laboratory, Los Alamos, NM, 87545, USA - alexei@lanl.gov
[b] MS D454, Applied Modern Physics Group, Los Alamos National Laboratory, Los Alamos, NM, 87545, USA - volegov@lanl.gov

KEY WORDS: Segmentation, scale, image vectorization, Delaunay triangulation, Minimum Spanning Tree, Scalable Vector Graphics, visual perception

ABSTRACT:

We present proximity graphs based approach to hierarchical image segmentation and vectorization. Our method produces an irregular pyramid that contains a stack of vectorized images of successively reduced levels of detail. We are jumping off from the over-segmented image represented by polygonal patches, which are attributed with spectral information. We employ constrained Delaunay triangulation combined with the proximity and closure principles known from the visual perception to extract the initial polygonal patches. They are built upon a triangular mesh composed of irregular sized triangles. We then represent the image as a graph with vertices corresponding to the polygons and edges reflecting polygon relations. This is followed by the iterative graph contraction based on Minimum Spanning Tree. The graph contractions merge the polygonal patches based on spectral differences between neighboring polygonal patches. The approach can be generalized to the multi-criteria MST to integrate other factors important for polygon agglomeration, in addition to spectral proximity considered in this investigation. An important characteristic of the approach is that initial and agglomerated polygonal patches are built in a way to retain spatial relationships among spectral discontinuities present in the original image.

1. INTRODUCTION

The concept of “object” plays one of central roles in image interpretation. However, the determination of what constitutes an object is extremely difficult. The major challenge to segmentation of the object-oriented pixel patches, which shape resembles the shape of real-world objects, is the high variability of relationships between the object and image context (background). While there has been considerable effort in the development of image segmentation, this problem remains great challenge for computer vision. This also has impact on the reliability of object recognition, which requires good whole-object segmentation. The presented approach aims at improving the quality of image segmentation via iterative process of polygonal patches agglomeration based on the combination of proximity graphs and ideas inspired by the Gestalt school of psychology.

Early in the 20th century the Gestalt school of psychology has shown the importance of the problem of perceptual organization for image interpretation. Wertheimer approached the problem by postulating principles that affect perceptual grouping and can be used for image segmentation (Wertheimer, 1958). The most known principle is proximity: all else being equal, the closer objects grouped strongly together. The other is similarity: the most similar elements in color, size, and orientation tend to be grouped together. Inspired by the visual psychology considerable progress has been achieved, and many image segmentation methods have been developed (e.g. Boyer, 2000; Sarkar, 2000). Many of them try to partition the image by optimizing a suitable cost function that encodes different perceptual characteristics and relationships between image elements.

Examples of global optimization based approaches include figure-ground separation methods developed by Herault and Horaud (Harault, 1993), Bhandarkar and Zeng (Bhandarkar, 1999). Harault and Horaud researched the use of simulated annealing, mean field annealing, and microcanonical annealing; Bhandarkar and Zeng investigated the use of genetic algorithm. They built cost function constructed out of terms based on cocircularity, proximity and smoothness to balance image element interactions. Recently an important development has been achieved in the area of graph-theoretic approach to the image segmentation and perceptual grouping problems. According to this approach, image structures such as pixels, pixel patches, and edges are described using graph, and the grouping is formulated as a graph-partitioning problem. Grouping is achieved based on spectral graph theory through minimizing some measure of the similarity between the different partitions relative to the similarity within each partition. The state-of-the-art is mostly represented by graph-cuts based approaches (e.g., Shi, 2000; Boykov, 2001; Yu, 2004). However, building the appropriate cost function and affinity matrices, capturing salient relationships among the image elements, and making optimization computationally tractable remains a challenge.

Another category of algorithms seeks optimal image partitioning through a sequence of computations that are done locally and involve elements within a relatively small regions. We emphasize approaches that are based on the use of proximity graphs, specifically Minimum Spanning Tree (MST). One of the initial MST based approaches is represented in the work of Zahn (Zahn, 1971). It is difficult to quantify the performance of the algorithm as it employs number of different heuristics, which can not be generalized. Felzenszwalb and Huttenlocher proposed to use Kruskal’s MST reconstruction algorithm to partition the image (Felzenszwalb, 1998), while

[*] Corresponding author.

Kropatsch and Haxhimusa use Boruvka’s MST reconstruction algorithm (Kropatsch, 2007; Haxhimusa, 2003). These approaches have shown better segmentation results than approaches based on regular pyramids. Besides, they have also provided better computational performance in spite of the fact that they both start from a raw pixel set.

In our view, to achieve better segmentation performance both in terms of computational efficiency and quality of segmentation it is necessary to replace pixels with intermediate level structures. If such structures (chunk knowledge) preserved information about spatial relations of image elements, and avoided excessive grouping of pixels during their reconstruction, they could be used as an initial data set for segmentation algorithms instead of pixels. This would improve overall outcome of the grouping process leading to better image segmentation. This is the problem we are trying to address with our approach.

We present a hierarchical image segmentation framework that derives hierarchy of attributed geometric primitives, such as polygonal patches, from raw pixel sets, and takes steps towards object-oriented image segmentation and high-level analysis. The framework incorporates combination of constrained Delaunay triangulation and the Gestalt principles of visual perception, such as proximity and closure, and exploits structural information on spectrally detected image edges and their spatial relations. This produces an initial set of polygonal patches. A polygonized image is then represented as a graph and initial polygonal patches are iteratively grouped into larger chunks using Boruvka’s MST algorithm. We show our results and discuss opportunities to improve the proposed approach.

2. HIERARCHICAL IMAGE VECTORIZATION

The first step is an extraction of object-oriented pixel patches based on salient image elements which constrain agglomeration of pixels into polygons. Selected salient elements are spectrally detected edges. Sought object-oriented patches are reconstructed by processing of detected edges. We use the image vectorization approach of (Prasad, 2006) to process edges and group pixels into polygons. The image vectorization starts with edge detection, e.g based on Canny edge detection (Canny, 1986) (Fig. 1b). It is followed by a constrained Delaunay triangulation (CDT) (Shewchuk, 1996) where the detected edges are used as constraints for Delaunay triangulation (Fig. 1c). CDT is followed by filtering the CDT generated triangle edge set, where the filtering keeps constraints and filters out the generated triangle edges based on a pre-specified set of rules inspired by principles of visual perception from the Gestalt psychology [Wertheimer, 1958]. The edge filtering is relied on the rules of proximity, closure, and contour completion (Figs. 2a, 2b). Proximity filters out the triangle edges using thresholding based on edge sizes (Fig. 2a). As a result, the spectrally detected edges that are spatially close to each other are linked by the kept triangle edges. Otherwise, spectrally detected edges are disconnected. The closure rule is responsible for filtering out the triangle edges that are bounded by the same spectral edge (Fig. 2b) or the same pair of spectral edges. Contour completion keeps the shortest triangle edge connecting end point of one spectral edge to interior point of another spectral edges, if this triangle edge meets proximity requirement (Fig. 2a). The triangle edge filtering results in a set of preserved edges: kept triangle edges and spectrally detected edges. Finally, a graph traversal algorithm (e.g., depth-first search or breadth-first search) is used to group triangles, which are not separated by the preserved edges. This process groups triangles into polygonal patches bounded by closed contours consisting of the spectral and triangle edges. These polygonal patches are assigned median spectral characteristics based on a sampling of pixels. Pixel sampling is performed by sampling triangles the polygonal patches built from. The result is a segmented image that is represented as a set of spectrally attributed polygonal patches: a vector image (Fig. 1d).

The technique produces visually appealing results (Figs. 3a-b, 4a-b, 5a-b) and reduces the amount of data, number of pixels to number of generated polygons, by 20-80 times depending on the image content. However, it does not produce a triangle grouping that can be directly utilized for object recognition or for interactive image segmentation; the vector image is still over-fragmented. This is due to lack of capability to extract and process really salient edges instead of all the detected ones, and

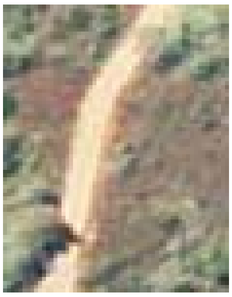
Figure 1a. Original image containing road fragment.

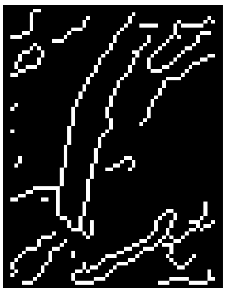
Figure 1b. Result of edge detection; edges are shown in white.

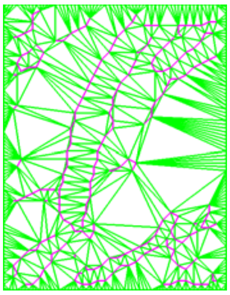
Figure 1c. Result of constrained Delaunay triangulation of the spectrally detected edges (shown in purple).

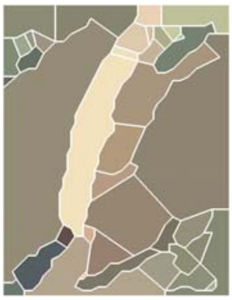
Figure 1d. Result of vectorizing image 1a. Polygons (white boundaries) built by grouping triangles.

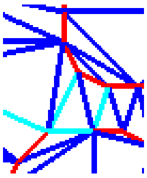
Figure 2a. Light blue triangle edges, linking different spectral edges, are preserved as the shortest links between spectral edges (in red), while blue triangle edges are filtered out.

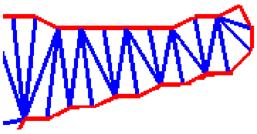
Figure 2b. Blue triangle edges are filtered out as they are bounded by the same spectral edge (in red).

the use of a pre-specified set of edge filtering rules which mostly rely on proximity and closure. An exploitation of all the edges (strong and weak ones) produces too many small polygons. We are currently addressing this problem through the iterative MST-based agglomeration of polygons. The agglomeration reduces number of edges by merging polygonal chunks, and preserves stronger edges if neighbouring chunks are spectrally very different. However, it is necessary to detect salient edges prior to the agglomeration as salient edges may be weak; thus they may be lost by the agglomeration and excessive grouping of polygonal chunks may take a place.

While the produced polygons fall short of representing real-world objects, they can be used as seeds to initiate their grouping into larger polygonal chunks, which shape would better resemble the shape of real-world objects. We use these polygons as seed objects for the hierarchical image segmentation. The advantage is that boundaries of these polygons reflect important discontinuities in image characterization, namely their boundaries are built along the spectrally detected edges. In turn, polygon grouping will be constrained in a sense that boundaries of agglomerated polygonal chunks will also be built along image spectral discontinuities. This is in contrast with other approaches, where selection of good seed pixel locations or good seed pixel patches is quite challenging. The problem is due to the fact that pixel itself, taken without any relationships to the image content, does not carry any object-oriented information. This uncertainty may have detrimental impact on the rest of segmentation process.

Once the over-fragmented vectorized image is created, we have an irregular polygonal grid, which structure is adapted to the image content. We then represent the vectorized image as a graph. Polygons are represented as graph nodes, and graph edges reflect their dissimilarities. We are currently using only polygons pairwise spectral dissimilarities to attribute the edges. We pre-specify a spectral threshold that guides the merging of polygons, and we also pre-specify maximum number of merging (graph contraction) iterations. Number of other schemes are available (e.g., Felzenszwalb, 1998) to characterize differences between image elements, and to control the agglomearion as a function of polygons internal and external variation. Another option is to consider strength of the edges separating the polygons.

Once the graph representation is created, we iteratively group polygons, starting from the fine level of detail (0th level) produced by initial vectorization process, into larger polygonal chunks using Boruvka's algorithm of Minimum Spanning Tree extraction. Boruvka's algorithm proceeds in a sequence of stages, and in each stage it identifies a forest F consisting of the minimum-weight edge incident to each vertex in the graph G, then forms the graph $G_1 = G \backslash F$ as the input to the next stage. $G \backslash F$ denotes the graph derived from G by contracting edges in F. Boruvka's algorithm takes $O(E \log V)$ time, where E is number of edges and V is number of vertices. This MST algorithm successively group polygons into larger chunks until reaching the maximum number of graph contractions or approaching the threshold on dissimilarity between polygonal chunks.

The currently used color similarity of the polygonal chunks is measured in Munsell (HVC) color space. The HVC color space

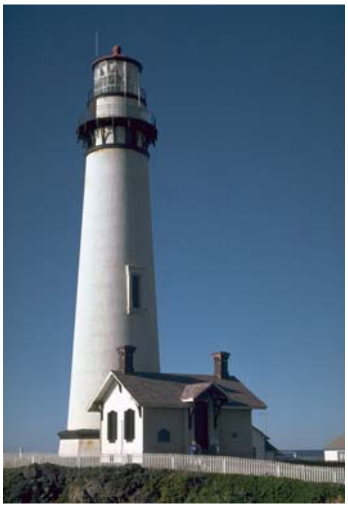

Figure 3a. Original image, 490×727 pixels.

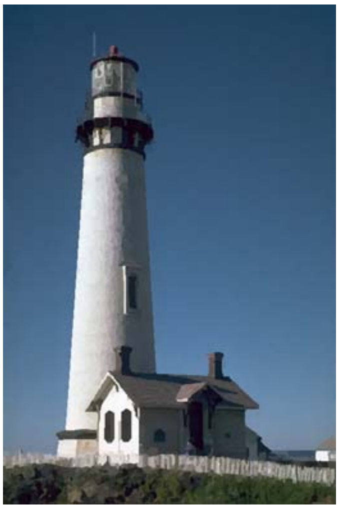

Figure 3b. Fine, 0th, level of detail: 12,518 polygons.

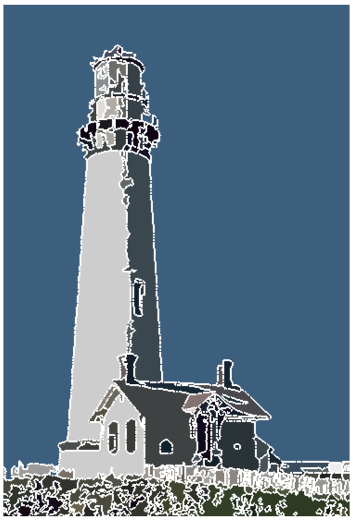

Figure 3c. Result of vectorizing image 3(a).
4th level of detail (result of 3 contraction iterations): 303 polygonal chunks. Contours of polygonal chunks are shown in white.

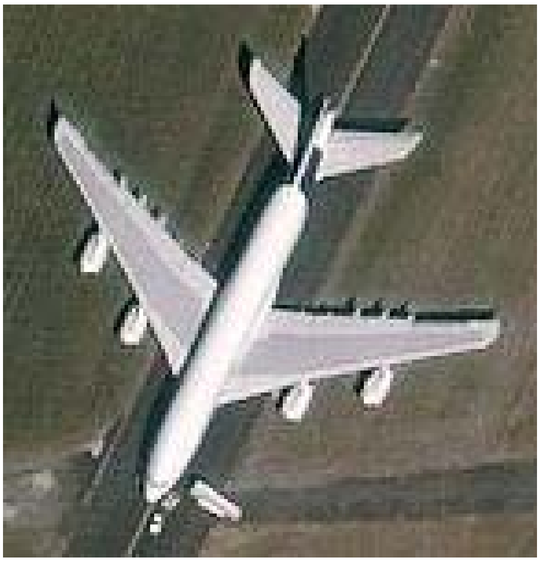

Figure 4a. Original image, 145×141 pixels.
Source: DigitalGlobe.com. The plane was cropped out of the Digitalglobe's image of Le Bourget air show.

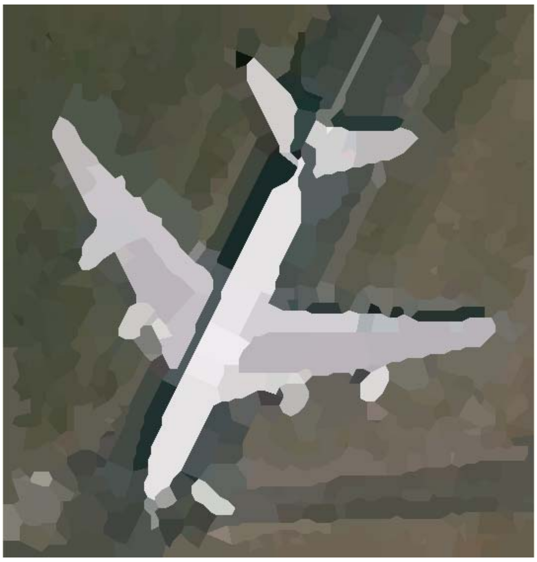

Figure 4b. Result of vectorizing image 4(a).
Fine, 0th, level of detail: 598 polygons.

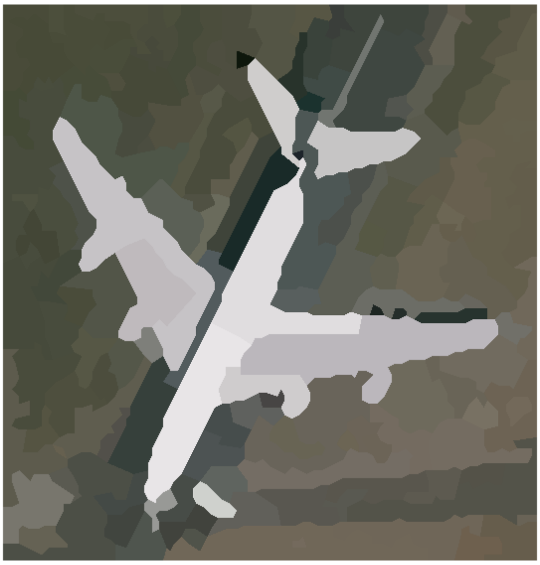

Figure 4c. 1st level of detail: 156 polygonal chunks.

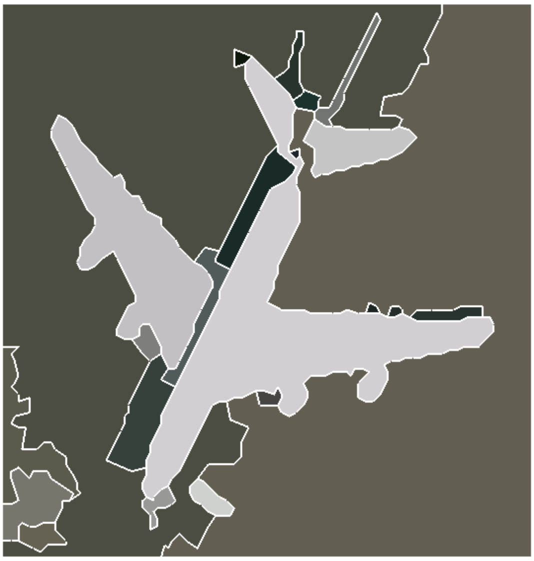

Figure 4d. 4rh level of detail: 23 polygonal chunks.

is perceptually uniform. Given a pair of polygonal chunks of colors (H_1, V_1, C_1) and (H_2, V_2, C_2) their spectral difference is computed using the equation (Miyahara, 1988):

$$\Delta c = 1.2 \cdot \sqrt{(2 \cdot C_1 \cdot C_2)(1-\cos(2\pi\Delta H/100)) + (\Delta C)^2 + (4\Delta V)^2} \quad (1)$$

where $\Delta H = |H_1 - H_2|$,
$\Delta C = |V_1 - V_2|$, and
$\Delta V = |C_1 - C_2|$.

3. EXPERIMENTAL RESULTS

Figures 3 through 5 show some of the results of processing ground-based (Fig. 3) and satellite imagery (Figs. 4, 5) using the presented algorithmic framework. Figs. 4 and 5 are small portions of larger DigitalGlobe images available from the DigitalGlobe sample library. Note that it is difficult to distinguish the original raster images and their vector versions corresponding to the fine level of detail. This is achieved through the use of all the detected edges. This results in good texture representation. At the same time the data size reduction factor (the ratio of the number of pixels to number of polygonal chunks) at fine level of detail is equal to 28, 36, and 39 in Figs. 3b, 4b, 5b correspondingly. First iteration of grouping that results in coarser level of detail, 1st level, continues to preserve visual quality of the original images, while reducing number of polygons about 3 times. Significant visual changes take place after 2nd contraction iteration. Note how a rooftop in Fig. 5d got merged with a road, while at previous level of detail they were different polygonal chunks. The reason is that they became spectrally close as the agglomeration was proceeding. This illustrates a need for more advanced criteria to reconstruct MST. Specifically, it is necessary to take into account structural

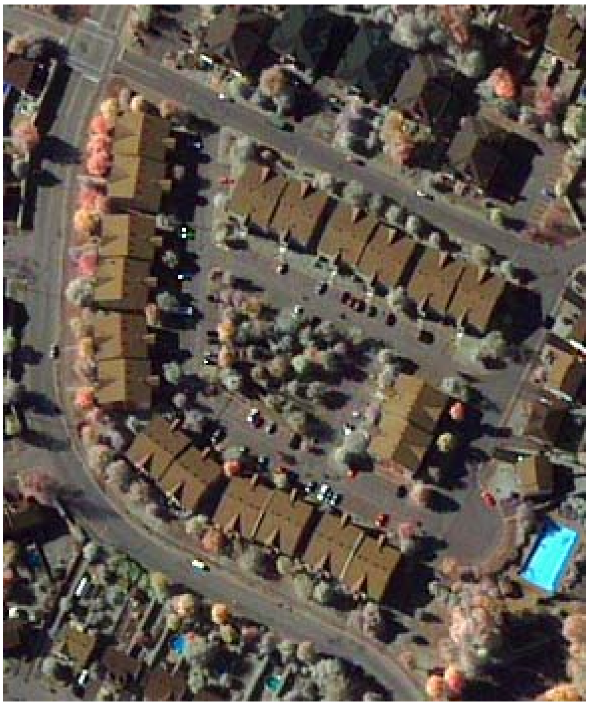

Figure 5a. Original image, 293×350 pixels.
Source: DigitalGlobe.com. This example is part of the Digitalglobe's image of Ottawa, Canada.

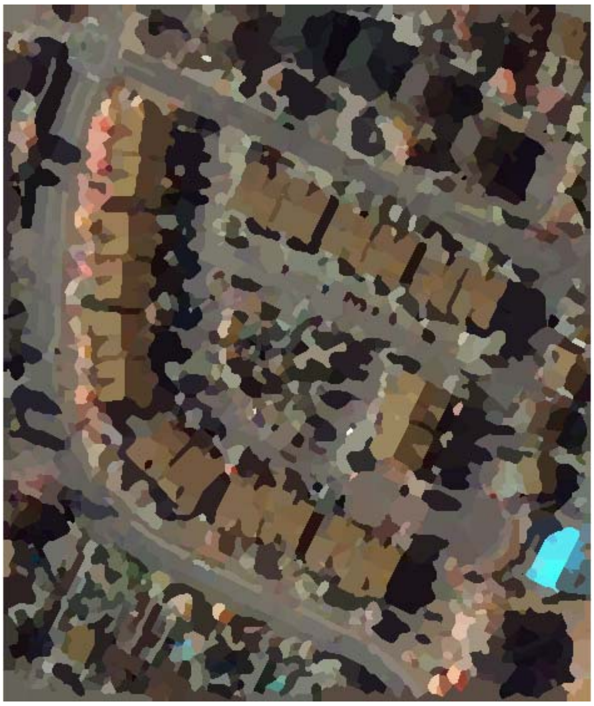

Figure 5b. Result of vectorizing image 5(a).
Fine, 0th, level of detail: 2,612 polygons.

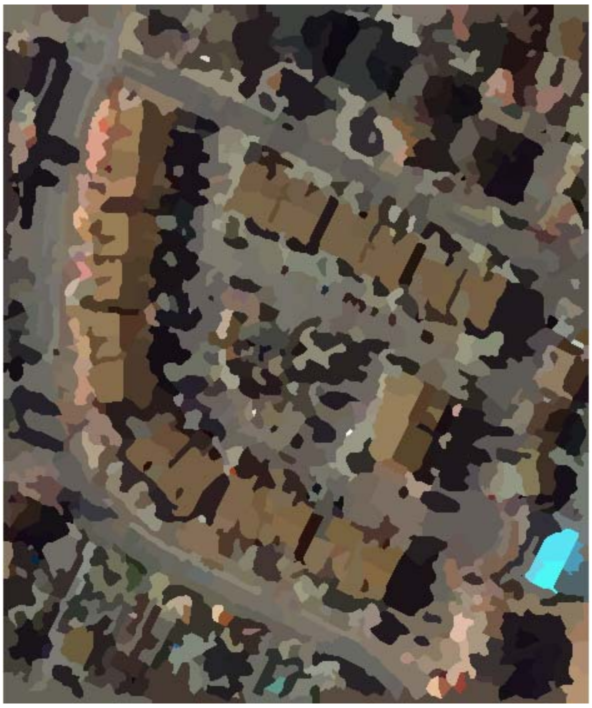

Figure 5c. 1st level of detail: 842 polygons.

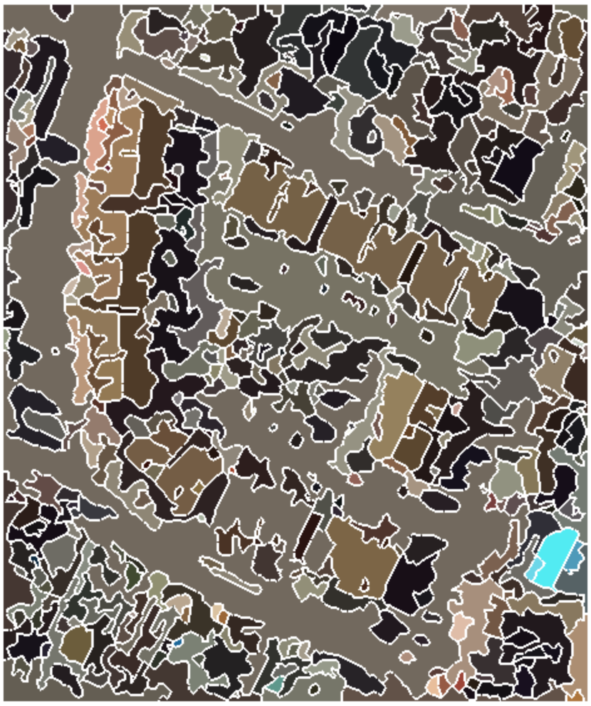

Figure 5d. 4th level of detail: 527 polygons

relationships of the edges separating polygonal patches.

Prior to the MST grouping we detected image edges using Canny edge detection. We used the following Canny parameters: σ = 1., hysteresis low threshold = 2.5, and hysteresis high threshold = 5. Color images were converted into gray-scale image I prior to edge detection by averaging their R, G, and B channels. We set up spectral difference threshold to

3. The initial over-segmented vector images were processed by two-three contractions to produce coarser levels of detail.

If an original image contains a lot of texture, then computationally most expensive step of the presented approach is CDT that has complexity of *O(NlogN)*, where N is the number of points (such as detected edge points). For instance, for an image of about 14000×14000 pixels, taken (by

DigitalGlobe) over urban area, the MST-based extraction of 3 vector levels of detail using an initial set of 4.8×10^6 polygons takes 2 minutes on 2.66 GHz machine. Extraction of this initial polygonal data set (of 4.8×10^6 polygons) takes 2 hours 12 minutes, of which 96% of the time is taken by CDT. Processing smaller image, such as 1000×1000 pixels, takes about 2 seconds, of which CDT takes 12% of the time.

We plan to make software implementing our approach to image vectorization and multi-scale image segmentation available for research purposes in the summer of 2008.

4. CONCLUSIONS AND OUTLOOK

We have demonstrated a proximity graphs based approach to extract hierarchy of image segmentations. This approach uses spectral dissimilarities as criterion to merge polygonal patches and consists of two stages: extraction of initial set of polygonal patches representing an over-segmented image; followed by a sequence of graph contractions based on Boruvka's MST extraction algorithm. Proximity graphs are used at both steps. First, constrained Delaunay triangulation is used to build initial small polygons using structural relations between the image edges. Second, Minimum Spanning Tree is used to merge polygons based on polygons' adjacency relationships. This combination of proximity graphs and polygonal patches distinguishes our approach from other segmentation methods, which start from grouping pixels.

We are currently investigating more advanced schemes to exploit dissimilarities between polygonal chunks. These extensions include both spectral and structural relationships among the polygons, such as structural relations among the edges bounding polygonal chunks. It is also possible to apply object recognition techniques to chunks produced at each level of detail to prevent the chunks recognized as objects of interest from being merged with other polygons by the MST-based agglomeration process. In order to approach near real-time performance for processing large images it is necessary to integrate the presented approach with the detection of salient image edges. This would reduce number of edges that are used as constraints for CDT; thus it would speed up CDT step.

One of interesting application avenues of the presented approach is the interactive image analysis and mapping applications on the Web using vector formats, such as Scalable Vector Graphics (SVG). SVG connects well with other emerging web GIS services such as the Open Geospatial Consortium Web Feature Service and Web Map Service standards.

5. REFERENCES

Bhandarkar, S. M., Zeng, X., 1999. Evolutionary approaches to figure-ground separation. *Applied Intelligence*, 11, pp. 187-212.

Boyer, K.L., Sarkar, S., (Eds.), 2000. *Perceptual Organization for Artificial Vision Systems*, Kluwer Acad. Publ.

Boykov, Y., Veksler, O., Zabin, R., 2001. Fast approximate energy minimization via graph cuts. *IEEE Trans. On Pattern Analysis and Machine Intelligence*, 23(11), 1222-1239.

Canny, J., 1986. A computational approach to edge detection. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 8(6), pp. 679-698.

Felzenszwalb, P. F., Huttenlocher, D. P., 1998. Image segmentation using local variation. In: *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 98-104.

Haxhimusa, Y., Kropatsch, W. G., 2003. Hierarchical image partitioning with dual graph contraction. In: *Proceedings of the DAGM Symposium*, pp. 338-345.

Herault, L., Horaud, R., 1993. Figure-ground discrimination: a combinatorial optimization approach. *IEEE Trans. On Pattern Analysis and Machine Intelligence*, 15(9), pp. 899-914.

Kropatsch, W. G., Haxhimusa, Y., Ion, A., 2007. Multiresolution image segmentation in graph pyramids. In: *Applied Graph Theory in Computer Vision and Pattern Recognition,* Kandel, A., Bunke, H. H., Last, M. (Eds.) Series: Studies in Computational Intelligence, 52, pp. 3-42.

Miyahara, M., Yoshida, Y. 1988. Mathematical transform of (r,g,b) color data to Munsell (h,v,c) color data. In: *SPIE Proceedings in Visual Communication and Image Processing*, vol. 1001.

Prasad, L., Skourikhine, A. N., 2006. Vectorized image segmentation via trixel agglomeration. *Pattern Recognition*, 39(4), pp. 501-514.

Sarkar, S., Soundararajan, P., 2000. Supervised learning of large perceptual organization: graph spectral partitioning and learning automata. *IEEE Trans. On Pattern Analysis and Machine Intelligence*, 22(5), pp. 504-525.

Scalable Vector Graphics (SVG), http://www.w3.org/Graphics /SVG/ (accessed 2 May 2008)

Shewchuk, J. R., 1996. Triangle: engineering a 2D quality mesh generator and Delaunay triangulator. *Lecture Notes in Computer Science*, 1148, pp. 203-222.

Shi, J., Malik, J., 2000. Normalized cuts and image segmentation. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 22(8), 888-905.

Wertheimer, M., 1958. Principles of perceptual organization. In: *Readings in Perception*, Beardslee, D., Wertheimer, M., (Eds.), D. Van Nostrand, Princeton, NJ, pp. 115-135.

Yu, S.X., 2004. Segmentation using multiscale cues. In: *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, 1, pp. 247-254.

Zahn, C. T., 1971. Graph-theoretic methods for detecting and describing gestalt clusters. *IEEE Trans. Comp.*, 20, pp. 68-86.

6. ACKNOWLEDGEMENTS

Triangulation code, Triangle, by J. R. Shewchuk, was used for triangulating image edge datasets. Visit the Triangle website at http://www.cs.cmu.edu/~quake /triangle.html for details.