



HAL
open science

Hierarchical image partitioning using combinatorial maps

Yll Haxhimusa, Adrian Ion, Walter G. Kropatsch, Luc Brun

► **To cite this version:**

Yll Haxhimusa, Adrian Ion, Walter G. Kropatsch, Luc Brun. Hierarchical image partitioning using combinatorial maps. Joint Hungarian-Austrian Conference on Image Processing and Pattern Recognition, 2005, Prague, Czech Republic. pp.179-186. hal-00815662

HAL Id: hal-00815662

<https://hal.science/hal-00815662>

Submitted on 19 Apr 2013

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Hierarchical Image Partitioning using Combinatorial Maps

Yll Haxhimusa ^a, Adrian Ion ^a, Walter G. Kropatsch ^a, and Luc Brun ^b

¹⁾ ^a Pattern Recognition and Image Processing Group,

Institute of Computer Aided Automation, Vienna University of Technology, Austria

^b GREY CNRS UMR 6072,

Ecole Nationale Supérieure d'Ingénieurs de Caen 6, France

^a{yll, ion, krw}@prip.tuwien.ac.at, ^bluc.brun@greyc.ensicaen.fr

Abstract:

We present a hierarchical partitioning of images using a pairwise similarity function on a combinatorial map based representation. We used the idea of minimal spanning tree to find region borders quickly in a bottom-up way, based on local differences. The result is a hierarchy of image partitions with multiple resolutions suitable for further goal driven analysis. The algorithm can handle large variation and gradient intensity in images. Dual graph representations lack an explicit encoding of the orientation of planes, existing in combinatorial maps.

1 Introduction

The authors in [10] suggested to bridge and not to eliminate the representational gap, and to focus efforts on *region segmentation*, *perceptual grouping*, and *image abstraction*. They employ a region-adjacency-graph based technique in order to produce *meta-regions*²⁾ of a particular view of a generic model. They start from a single vertex, representing a single region (silhouette), which is derived from the input region adjacency graph (produced by the graph-based segmentation method in [5]), by merging in a pairwise manner the regions of an example image. After having the apexes (of two or more images) they proceed in a top-down manner to find the decomposition of each apex region into two subregions by comparing corresponding shapes and *relations* among the corresponding regions.

The union of regions forming the group is again a region with both internal and external properties and relations. Low-level cue image segmentation cannot and should not produce a complete final *good* segmentation, because there is *an intrinsic ambiguity* in the exact location of region boundaries in digital images. Problems emerge because homogeneity of low-level cues will not map to the semantics [10], and the degree of homogeneity of a region is in general quantified

¹⁾This paper has been supported by the Austrian Science Fund under grant FSP-S9103-N04.

²⁾An output region adjacency graph as abstraction of the particular view.

by threshold(s) for a given measure [6]. A *grouping method* should have the following properties [5]: capture perceptually important groupings or regions which reflect global aspects of the image, be highly efficient, running in time *linear* in the number of image pixels (e.g. *minimal spanning tree*), and creating *hierarchical partitions* [17].

In a regular image pyramid the number of pixels at any level l , is r times higher than the number of pixels at the next reduced level $l + 1$. The so called reduction factor r is greater than one and it is the same for all levels l . If s denotes the number of pixels in an image I , the number of new levels on top of I amounts to $\log_r(s)$. Thus, the regular image pyramid may be an efficient structure for fast grouping and access to image objects in top-down and bottom-up processes. However, the authors in [1] conclude that regular image pyramids have to be rejected as general-purpose segmentation algorithms, because they lack shift invariance. In [15, 9, 14] it was shown how these drawbacks can be avoided by irregular image pyramids, the so called adaptive pyramids, in image segmentation and feature detection.

Region adjacency graphs (*RAG*), dual graphs [8] and combinatorial maps have been used before [3] to represent the partitioning of $2D$ space. From these three structures, we use the combinatorial maps because, *RAGs* cannot correctly encode multiple boundaries and inclusions, and dual graphs lack the explicit encoding of edge orientation around vertices (see Section 2.1 for a problem in $2D$), present in a combinatorial map [3]. Moreover with combinatorial maps, its dual must not be explicitly represented because one combinatorial map is enough to fully characterize the partition, and its dual can be easily deduced anytime.

In this paper we present a hierarchical method, introduced in [8], which, in a bottom-up manner produces a stack of region adjacency combinatorial maps (called *irregular combinatorial pyramid*³⁾), and at the same time preserves the proper *topology* among regions during the merging processes, suitable for the top-down decomposition (e.g. as in [10]). Combinatorial maps and combinatorial pyramids are shortly presented in Section 2. Borůvka's algorithm [2] was used to build a minimal spanning tree (*MST*), since it was easily integrated in our combinatorial pyramid concept. Combinatorial map contraction is presented in Section 3. In Section 4 we present some experimental results.

2 Combinatorial Maps

Combinatorial maps and generalized combinatorial maps define a general framework which allows to encode any subdivision on nD topological spaces orientable or non-orientable with

³⁾called lattice in [10].

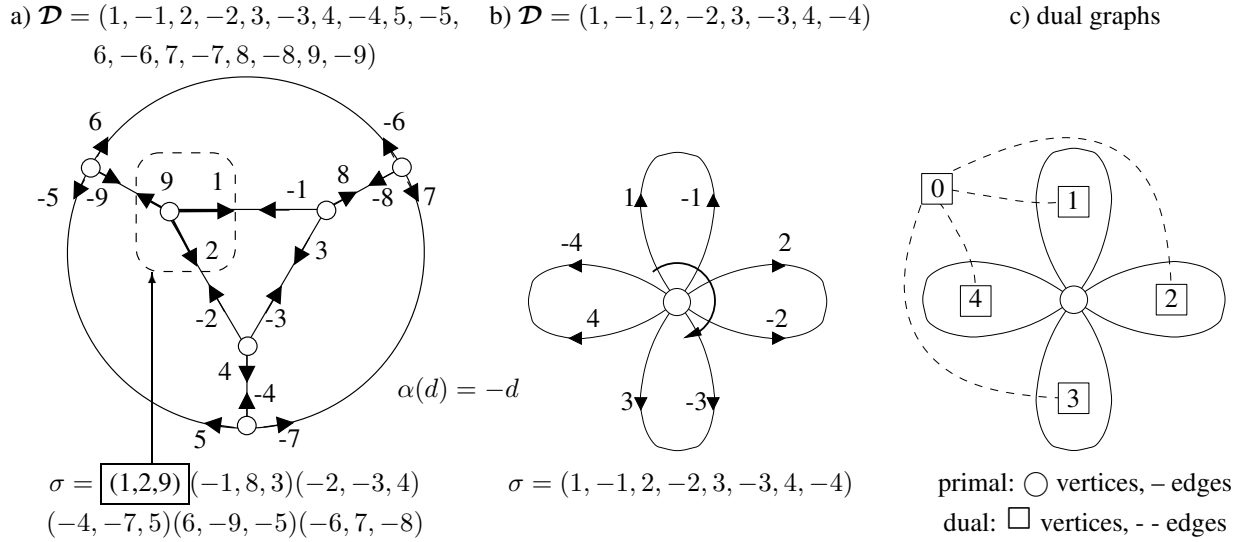


Figure 1: a), b) Combinatorial maps. b), c) Topological case handled correctly only by combinatorial map.

or without boundaries. Using $2D$ images, combinatorial maps may be understood as a particular encoding of a planar graph, where each edge is split into two half-edges called darts. Since each edge connects two vertices, each dart belongs to only one vertex. A $2D$ combinatorial map is formally defined by the triplet $G = (\mathcal{D}, \sigma, \alpha)$ [4] where \mathcal{D} represent the set of darts and $\sigma(d)$ is a permutation on \mathcal{D} encountered when turning clockwise around each vertex. Finally $\alpha(d)$ is an involution on \mathcal{D} which maps each of the two darts of one edge to the other one. Given a combinatorial map $G = (\mathcal{D}, \sigma, \alpha)$, its dual is defined by $\overline{G} = (\mathcal{D}, \varphi, \alpha)$, with $\varphi = \sigma \circ \alpha$. The cycles of permutation φ encode the faces of the combinatorial map. In what it follows, the cycles of α , $\sigma(d)$ and φ contain a dart d will be respectively denoted by $\alpha^*(d)$, $\sigma^*(d)$ and $\varphi^*(d)$. Thus all graph definitions used in irregular pyramids [11] are analogously defined. A *combinatorial pyramid* is a stack of combinatorial maps successively reduced by the set of contraction and removal operations, i.e. (G_0, \dots, G_k) , where k represent the levels of the pyramid. Each map $k+1$ is build from the one below, k , by selecting a set of contraction kernels $K_{k,k+1}$ and applying it to a given combinatorial map G_k to get the reduced $G_{k+1} = \mathcal{C}[G_k, K_{k,k+1}] = G_k \setminus K_{k,k+1}$. More on removal of the redundant edges can be found in [3].

2.1 Combinatorial Maps versus Dual Graphs

Advantages of combinatorial maps over dual graphs come from the embedding, that is inherently present at the former ones. Let us analyze the 'flower' example given in Figure 1b,c w.r.t uniqueness of topological representation. The combinatorial map of this 'flower' is shown and

defined in Figure 1b) by $G = (\mathcal{D}, \sigma, \alpha)$ ⁴⁾. If the leafs of the 'flower' exchange position for e.g. leafs 1 and 3, a different $\sigma = (3, -3, 2, -2, 1, -1, 4, -4)$ will be defined, hence uniquely encoding the topology. The dual graphs are encoded by a pair of graphs, the (planar) primal graph vertices) and its dual. For each edge in the primal graph there is a corresponding one in the dual, that crosses it (Figure 1c). Since there is no ordering of the edges around the vertices, the dual graph representation does not uniquely encode the topology of the 'flower', as can be easily seen if we exchange the position, for e.g. of leafs 1 and 3, the dual graph describing this configuration is identical to the previous one (the one without exchanging the position of leafs.)

3 Image Partitioning

The authors in [5] define a function, which measures the difference along the boundary of two components relative to a measure of the differences of components' internal differences. This definition tries to encapsulate the intuitive notion of contrast: a contrasted zone is a region containing two connected components whose inner differences (*internal contrast*) are less than differences within its context (*external contrast*). We define an external contrast measure between two components and an internal contrast measure of each component, analogously to [5, 7].

Let $P_k = \{CC_i^k, CC_j^k, \dots, CC_n^k\}$ be the partitions on the level k of the pyramid i.e P_k is the attributed combinatorial map $G_k(\mathcal{D}_k, \sigma_k, \alpha_k, a_k)$, its vertex set $\sigma_k^*(D_k)$ by V_k , edge set $E_k = \alpha^*(\mathcal{D})$ and $a_k : \mathcal{D} \rightarrow \mathbb{R}^+$ a weight function. One way to attribute the darts is given in Section 4. Every vertex $u \in V_k$ is a representative of a component $CC_i = RF(u_i)$ ⁵⁾ of the partition P_k . The equivalent contraction kernel of a vertex $u \in V_k$, $K_{0,k}(u)$ is a set of darts (a subtree) of the base level $G_0(\mathcal{D}_0, \sigma_0, \alpha_0)$ that are contracted; i.e. applying the equivalent contraction kernel on the base level, one contracts the sub combinatorial map $G'_0 \subseteq G_0$ onto the vertex u .

The internal contrast of a connected component $CC_i \in P_k$ is defined as the largest dissimilarity between its vertices. Such a dissimilarity is defined as the largest weight of the darts in $K_{0,k}(u_i)$:

$$Int(CC_i) = \max\{a(d), d \in \alpha^*(K_{0,k}(u_i))\}. \quad (1)$$

Let $u_i, u_j \in V_k$ be the end vertices of an edge $\alpha_k^*(d) \subset D_k$. The *external contrast* between two components $CC_i, CC_j \in P_k$ is the *smallest dissimilarity* between the components CC_i and CC_j i.e. the smallest dart weight connecting $K_{0,k}(u_i)$ and $K_{0,k}(u_j)$ of vertices $u_i \in CC_i$ and $u_j \in CC_j$:

$$Ext(CC_i, CC_j) = \min\{a(d), d \in K_{0,k}(u_i) \wedge \alpha_k(d) \in K_{0,k}(u_j)\}. \quad (2)$$

⁴⁾ σ is encoded clockwise, shown with the arrow in Figure 1b).

⁵⁾connected components CC , and receptive field (RF)

Algorithm 1 – Construct Hierarchy of Partitions

Input: Attributed combinatorial map G_0 .

- 1: $k = 0$
- 2: **repeat**
- 3: $\forall u \in V_k = \sigma_k^*(\mathcal{D}_k)$
- 4: $D_{min}(u) = \{d \in \mathcal{D} \mid a(d) = \min\{a(d') \mid d \in \sigma^*(d')\}\}$
- 5: $\forall d \in D_{min}, u_i^k = \sigma_k^*(d), u_j^k = \sigma_k^*(\alpha(d))$ with $Ext(CC_i^k, CC_j^k) \leq PInt(CC_i^k, CC_j^k)$
- 6: include d and $\alpha(d)$ in contraction kernel $K_{k,k+1}$
- 7: contract combinatorial map G_k with contraction kernel, $K_{k,k+1}$: $G_{k+1} = \mathcal{C}[G_k, K_{k,k+1}]$.
- 8: set $a(d_{k+1}) = \min\{a(d_k) \mid d_{k+1} = \mathcal{C}[d_k, K_{k,k+1}]\}$
- 9: $k = k + 1$
- 10: **until** $G_k = G_{k-1}$

Output: An attributed combinatorial map at each level of the pyramid (G_0, G_1, \dots, G_k) .

The pairwise comparison function $Comp(\cdot, \cdot)$ between two connected components is defined:

$$Comp(CC_i, CC_j) = \begin{cases} \text{True} & \text{if } Ext(CC_i, CC_j) > PInt(CC_i, CC_j), \\ \text{False} & \text{otherwise,} \end{cases} \quad (3)$$

where $PInt(CC_i, CC_j)$ is the minimum internal contrast difference between two components:

$$PInt(CC_i, CC_j) = \min(I(CC_i) + \tau(CC_i), I(CC_j) + \tau(CC_j)). \quad (4)$$

For the function $Comp(CC_i, CC_j)$ to be true i.e. for the border to exist, the external contrast difference must be greater than the internal contrast difference. The reason for using a threshold function $\tau(CC)$ in Eq. (4) is that for small components CC , $Int(CC)$ is not a good estimate of the local characteristics of the data, in extreme case when $|CC| = 1$, $Int(CC) = 0$. Any non-negative function of a single component CC , can be used for $\tau(CC)$ [5]. One can define τ to be function of the size of CC : $\tau(CC) = \beta/|CC|$, where $|CC|$ denotes the size of the component CC and β is a constant. More complex definition of $\tau(CC)$, which is large for certain shapes and small otherwise would produce a partitioning which prefers these shapes. Algorithm 1 shows how to build the hierarchy of partitions. Basically the algorithms starts by collecting darts with the smallest weight around vertices $D_{min}(u)$, and then checks if the weights of these darts fulfill the condition for merging the regions $Ext(CC_i^k, CC_j^k) \leq PInt(CC_i^k, CC_j^k)$.

4 Experiments on Image Maps

We start with the trivial partition, where each pixel is a homogeneous region. The attributes of edges can be defined as $d(F(u_i), F(v_j))$, where d is some distance function and $F(u_i)$ denotes



Figure 2: Some levels of the partitioning of ‘Tulips’: level (number of components).

the attributes of u_i . Using gray level images, d and F may be respectively defined as the distances deduced from the L_1 norm from the gray level intensity of pixels. Using color images, one may think to valuate each edge by the Euclidean distance between the vertice’s colors using a perceptual color space such as the *CIE* Luv or Lab. However, the use of *CIE* color spaces requires the knowledge of the illuminants defining *RGB* components which is often not available. Therefore, for the sake of simplicity and in order to valuate our method we choose in our experiments a simple Euclidean distance in *RGB* space. However the choice of the definition of the weights and the features to be used is in general a hard problem, since the grouping cues could conflict each other [13].

For the experiments shown in Figure 2, 3, 4, we used as attributes of edges the euclidean distance between pixel *RGB* values, $a(u_i, u_j) = |rgb(u_i) - rgb(u_j)|$. To compute the hierarchy of partitions, we also need to define $\tau(CC) = \beta/|CC|$, where $\beta = const$ and $|CC|$ is the number of elements in CC , i.e. the size of the region. The algorithm has one running parameter β , which is used to compute the function τ . A larger constant β sets the preference for larger components. Note that as size of $|CC|$ gets larger, which happens as the algorithms proceeds toward the top of the pyramid, the function $\tau \rightarrow 0$, which means that the influence of the parameter β decreases. We found that $\beta = 300$ produces the best hierarchy of partitions of the images shown in *Tulips* Figure 2, *Obj18_355* Figure 3, and *Obj59_0* ⁶⁾Figure 4. Figures 2, 3 and 4 show some of the partitions on different levels of the pyramid and the number of components. In general the top of the pyramid will consist of one vertex, an apex, which represents the whole image. Note that in all images there are regions of large intensity variability and gradient. This algorithm copes with this kind of gradient and variability. In contrast to [5]⁷⁾ the result is a hierarchy of partitions with multiple resolutions, suitable for further goal driven, domain specific analysis ⁸⁾. On the lower level of the pyramid the image is over segmented (partitioned) whereas

⁶⁾Waterloo image database and Coil 100 image database

⁷⁾In [16] results of different segmentation methods, including the ones in [5] and [13], are shown and compared.

⁸⁾Please note that a whole class of partitions is created, where a partition is not limited to a certain level of the pyramid, but can be constructed of components from different levels (the receptive fields of the vertices of a



Figure 3: Some levels of the partitioning of ‘Obj18_355’: level (number of components).



Figure 4: Some levels of the partitioning of ‘Obj59_0’: level (number of components).

in the upper it is under segmented (partitioned), the help of mid and high level knowledge would select the proper partitioning. An approach based on some statistical measures to decide which level is the most appropriate could be used as well [12]. Since the algorithm preserves details in low-variability regions, a noisy pixel would survive through the hierarchy. Of course, image smoothing in low variability regions would overcome this problem. We, however do not smooth the images, as this would introduce another parameter into the method. The hierarchy of partitions can also be built from an over segmented image to overcome the problem of noisy pixels. The constant β is used to handle an over segmented image at the lower levels of the pyramid. For an over segmented image, where the size of regions is large, there is no need to define the function τ , thus the algorithm becomes parameterless.

5 Conclusion

In this paper we presented a method for building hierarchical image partitions using Borůvka’s minimal spanning tree algorithm. The hierarchy is presented as a combinatorial pyramid, where each level is a 2D combinatorial map. Combinatorial maps are defined in any dimension, thus the current work should lead the way to segmentation of digital video streams using contraction in 3D combinatorial maps/pyramids. It was shown that the algorithm can handle large variation and gradient intensity in images. Even though the algorithm makes greedy decisions locally, it produces perceptually important partitions in a bottom-up way based only on local differences.

multilevel partition occupy the whole image, and do not overlap)

A drawback is that the maximum and the minimum criterion is very sensitive to noise, although in practice it has a small impact. To overcome the problem of noise, one could start with an over segmented image produced by a robust method e.g. robust watershed method. A comparison between the classes of partitions produced by the presented method, and some of the well known methods (e.g. [5, 17]) is planned.

References

- [1] M. Bister, J. Cornelis, and A. Rosenfeld. A critical view of pyramid segmentation algorithms. *Pattern Recognition Letters*, 11(9):605–617, September 1990.
- [2] O. Borůvka. O jistém problému minimálním. *Práce Mor. Přírodověd. Spol. v Brně (Acta Societ. Scienc. Natur. Moravicae)*, 3(3):37–58, 1926.
- [3] L. Brun and W. Kropatsch. Contraction kernels and combinatorial maps. *Pattern Recognition Letters*, 24(8):1051–1057, 2003.
- [4] L. Brun and W. G. Kropatsch. Dual contraction of combinatorial maps. Technical Report PRIP-TR-54, Institute f. Computer Aided Automation 183/2, PRIP Group, TU Wien, Austria, 1999.
- [5] P. F. Felzenszwalb and D. P. Huttenlocher. Image segmentation using local variation. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 98–104, June 1998.
- [6] C.-S. Fu, W. Cho, S, and K. Essig. Hierarchical color image region segmentation for content-based image retrieval system. *IEEE Transaction on Image Processing*, 9(1):156–162, 2000.
- [7] L. Guigues, L. M. Herve, and J.-P. Cocquerez. The hierarchy of the cocoons of a graph and its application to image segmentation. *Pattern Recognition Letters*, 24(8):1059–1066, 2003.
- [8] Y. Haxhimusa and W. G. Kropatsch. Segmentation graph hierarchies. In A. Fred, T. Caelli, R.P.W. Duin, A. Campilho, and D. de Ridder, eds., *Joint Int. Worksh. on S+SSPR*, LNCS 3138:343–351, 2004.
- [9] J.-M. Jolion and A. Montanvert. The adaptive pyramid, a framework for 2D image analysis. *Computer Vision, Graphics, and Image Processing: Image Understanding*, 55(3):339–348, May 1992.
- [10] Y. Keselman and S. Dickinson. Generic model abstraction from examples. *IEEE Transactions on PAMI, special issue on Syntactic and Structural Pattern Recognition*, 2005. to appear.
- [11] W. G. Kropatsch. Parallel graph contraction for dual irregular pyramids. Technical Report PRIP-TR-28, Institute f. Automation 183/2, Dept. for Pattern Recognition and Image Processing, TU Wien, Austria, 1990.
- [12] S. Lallich, F. Muhlenbach, and J.-M. Jolion, A test to controll a region growing process within a hierarchical Graph. In *Pattern Recognition*, 36(10):2201-2211, 2003.
- [13] J. Malik, S. Belongie, J. Shi, and T. Leung. Textons, contours and regions: Cue integration in image segmentation. In *The International Conference on Computer Vision*, pages 918–925, 1990.
- [14] P. Meer, D. Mintz, A. Montanvert, and A. Rosenfeld. Consensus vision. In *AAAI-90 Workshop on Qualitative Vision*, pages 111–115, Boston, Massachusetts, USA, July 29 1990.
- [15] A. Montanvert, P. Meer, and A. Rosenfeld. Hierarchical image analysis using irregular tessellations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(4):307–316, April 1991.
- [16] M. Pavan and M. Pelillo. Dominiant sets and hierarchical clustering. In *ICCV03*, 2003.
- [17] J. Shi and J. Malik. Normalized cuts and image segmentation. In *Proceedings IEEE Conference Computer Vision and Pattern*, pages 731–737, 1997.