

# Hierarchy of Partitions with Dual Graph Contraction <sup>\*</sup>

Yll Haxhimusa and Walter Kropatsch

Pattern Recognition and Image Processing Group 183/2,  
Institute for Computer Aided Automation, Vienna University of Technology, Austria  
{yll,krw}@prip.tuwien.ac.at

**Abstract.** We present a hierarchical partitioning of images using a pairwise similarity function on a graph-based representation of an image. This function measures the difference along the boundary of two components relative to a measure of differences of component's internal differences. This definition attempts to encapsulate the intuitive notion of contrast. Two components are merged if there is a low-cost connection between them. Each component's internal difference is represented by the maximum edge weight of its minimum spanning tree. External differences are the cheapest weight of edges connecting components. We use this idea to find region borders quickly and effortlessly in a bottom-up 'stimulus-driven' way based on local differences in a specific feature, like as in preattentive vision. The components are merged ignoring the details in regions of high-variability, and preserving the details in low-variability ones.

## 1 Introduction

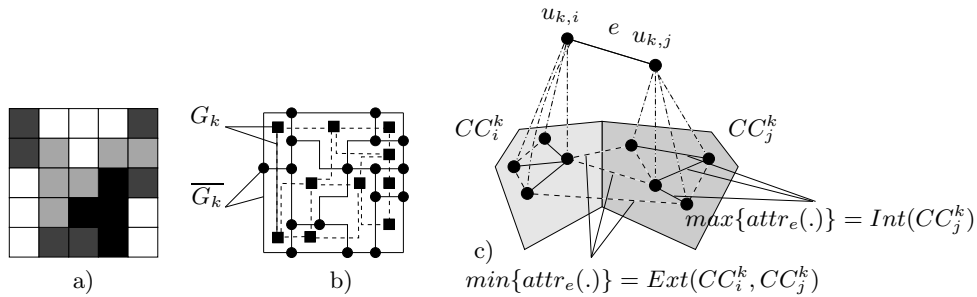
Wertheimer [19] has formulated the importance of wholes (Ganzen) and not of its individual elements, and introduced the importance of perceptual grouping and organization in visual perception. Low-level cue image segmentation cannot and should not produce a complete final "good" segmentation. The low-level coherence of brightness, color, texture or motion attributes should be used to come up sequentially with hierarchical partitions [18]. Mid and high level knowledge can be used to either confirm these groups or to select some for further attention. A wide range of computational vision problems could make use of segmented images, where such segmentation relies on efficient computation. For instance motion estimation requires an appropriate region of support for finding correspondence. Higher-level problems such as recognition and image indexing can also make use of segmentation results in the problem of matching.

It is important that a grouping method has the following properties [3]:

- captures perceptually important groupings or regions, which reflect global aspects of the image,
- is highly efficient, running in time linear in the number of image pixels,

---

<sup>\*</sup> This paper has been supported by the Austrian Science Fund under grants P14445-MAT and P14662-INF



**Fig. 1.** a) Partition of pixel set into cells. b) Representation of the cells and their neighborhood relations by  $(G_k, \overline{G_k})$  of plane graphs. c) Internal and External contrast.

– creates hierarchical partitions [18].

In a regular image pyramid the number of pixels at any level  $k$ , is  $r$  times higher than the number of pixels at the next reduced level  $k + 1$ . The so called reduction factor  $r$  is greater than 1 and it is the same for all levels  $k$ . 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 [17].

However, regular image pyramids are confined to globally defined sampling grids and lack shift invariance. Bister [1] concludes that regular image pyramids have to be rejected as general-purpose segmentation algorithms. In [9] it was shown how these drawbacks can be avoided by irregular adaptive image pyramids, where the hierarchical structure (vertical network) of the pyramid was not a priori known but recursively built based on the data. Moreover in [16, 13, 5] it was shown that irregular pyramids can be used for segmentation and feature detection.

The construction of an irregular pyramid is iteratively local [15]. This means that only local properties build the hierarchy of the pyramid. Each level represents a partition of the pixel set into cells [11], i.e. connected subsets of pixels. On the base level (level 0) of an irregular image pyramid the cells represent single pixels and the neighborhood of the cells is defined by the 4 (8)-connectivity of the pixels. A cell on level  $k + 1$  (parent) is a union of neighboring cells on level  $k$  (children). This union is controlled by so called contraction kernels (decomposition parameters [12]). Every parent computes its values independently of other cells on the same level. This implies that an image pyramid is built in  $O[\log(image\ diameter)]$  time. Neighborhoods on level  $k + 1$ , are derived from neighborhoods on level  $k$ . Two cells  $c_1$  and  $c_2$  are neighbors if there exist pixels  $p_1$  in  $c_1$  and  $p_2$  in  $c_2$  such that  $p_1$  and  $p_2$  are 4-neighbors (Fig. 1a). We assume that on each level  $k + 1$  ( $k \geq 0$ ) there exists at least one cell not contained in level  $k$ . In particular, there exists a highest level  $h$ . We represent the levels as dual pairs  $(G_k, \overline{G_k})$  of plane graphs  $G_k$  and its dual (plane) graph  $\overline{G_k}$  [6] (Fig. 1b). To achieve the planar embedding of graphs we use the 4-connectivity. The sequence  $(G_k, \overline{G_k})$ ,  $0 \leq k \leq h$  is called (dual) graph pyramid. Moreover the graph

is attributed,  $G(V, E, attr_v, attr_e)$ , where  $attr_v : V \rightarrow \mathbb{R}^+$  and  $attr_e : E \rightarrow \mathbb{R}^+$ . We use weights for  $attr_e$  depending on dissimilarity criteria.

The aim of this paper is to build a minimum weight spanning tree (*MST*) of regions of an image combining the advantage of regular pyramids (logarithmic tapering) with the advantages of irregular graph pyramids (their purely local construction and shift invariance). The aim is reached by the selection method for contraction kernels proposed in [6] to achieve logarithmic tapering, local construction and shift invariance. Borůvka’s algorithm [2] with dual graph contraction (DGC) [12] is used for building *MST* of the region and to preserve the graph topology. The topological relation seems to play an even more important role for vision tasks in natural systems than precise geometrical position. We build the *MST* to find region borders based on local differences in a specific feature. See the book of Jolion [10] for an extensive overview of the pyramid framework for early vision.

The plan of the paper is as follows. In Sec. 2 we give the merging decision criteria and we prove that the proposed algorithm builds a nested hierarchy of parititons. Sec. 3 reports on experimental results.

## 2 A Hierarchy of Partitions

Hierarchies are a significant tool for image partitioning as they are naturally combined with homogeneity criteria. Horowitz and Pavlidis [8] define a consistent homogeneity criteria over a set  $V$  as a boolean predicate  $P$  over its parts  $\Phi(V)$  that verifies the consistency property:

$$\forall(x, y) \in \Phi(V) \quad x \subset y \Rightarrow (P(y) \Rightarrow P(x)). \quad (1)$$

In image analysis Eq. 1 states that the subregions of a homogeneous region are also homogeneous. It follows that if *Pyr* is a hierarchy and  $P$  a consistent homogeneity criteria on  $V$  then the set of maximal elements of *Pyr* that satisfy  $P$  defines a unique partition of  $V$ . Thus the combined use of a hierarchy and homogeneity criteria allow one to define partitioning in a natural way.

The goal is to find partitions  $P_k = \{CC_1^k, CC_2^k, \dots, CC_n^k\}$  such that these elements satisfy certain properties. We use the pairwise comparison of neighboring vertices, i.e. partitions to check for similarities [3–5]. A pairwise comparison function,  $Comp(CC_i^k, CC_j^k)$  is true, if there is evidence for a boundary between  $CC_i^k$  and  $CC_j^k$ , and false when there is no boundary. Note that  $Comp(CC_i^k, CC_j^k)$  is a boolean comparison function for pairs of partitions. The definition of  $Comp(CC_i^k, CC_j^k)$  depends on the application.

The pairwise comparison function  $Comp(\cdot, \cdot)$  measures the difference along the boundary of two components relative to the differences of component’s internal differences. This definition tries to encapsulate the intuitive notion of contrast: a contrasted zone is a region containing two components whose inner differences (**internal contrast**) are less then differences between them (**external contrast**). We define an **external contrast** between two components and

an **internal contrast** of each component. These measures are defined in [3–5], analogously.

Every vertex  $u \in G_k$  is a representative of a connected component  $CC^k$  of the partition  $P_k$ . The equivalent contraction kernel [12] of a vertex  $u \in G_k$ ,  $N_{0,k}(u)$  is a set of edges on the base level that are contracted, i.e. applying  $N_{0,k}(u)$  on the base level contracts the subgraph  $G' \subseteq G$  onto the vertex  $u$ . The **internal contrast** of the  $CC^k \in P_k$  is the **largest dissimilarity** inside the component  $CC^k$  i.e. the largest edge weight of the  $N_{0,k}(u_k)$  of vertex  $u_k \in G_k$ , that is

$$Int(CC^k) = \max\{attr_e(e), e \in N_{0,k}(u_k)\}. \quad (2)$$

Let  $u_{k,i}, u_{k,j} \in V_k$  be the end vertices of an edge  $e \in E_k$ . The **external contrast** between two components  $CC_i^k, CC_j^k \in P_k$  is the **smallest dissimilarity** between component  $CC_i^k$  and  $CC_j^k$  i.e. the smallest edge weight connecting  $N_{0,k}(u_{k,i})$  and  $N_{0,k}(u_{k,j})$  of vertices  $u_{k,i}, u_{k,j} \in G_k$ :

$$Ext(CC_i^k, CC_j^k) = \min\{attr_e(e), e = (u_{k,i}, u_{k,j}) : u_{k,i} \in N_{0,k}(u_{k,i}) \wedge w \in N_{0,k}(u_{k,j})\}. \quad (3)$$

This definition is problematic since it uses only the “smallest” edge weight between the two components, making the method very sensitive to noise. But in practice this limitation works well as shown in Sec. 3. In Fig. 1c an example of  $Int(CC^k)$  and  $Ext(CC_i^k, CC_j^k)$  is given. The  $Int(CC_i^k)$  of the component  $CC_i^k$  is the *maximum* of weights of the solid edges (analogously for  $Int(CC_j^k)$ ), whereas  $Ext(CC_i^k, CC_j^k)$  is the *minimum* of weights of the dashed edges connecting component  $CC_i^k$  and  $CC_j^k$ . Vertices  $u_{k,i}$  and  $u_{k,j}$  are representative of the components  $CC_i^k$  and  $CC_j^k$ . By contracting the edges  $N_{0,k}(u_{k,i})$  (see solid edges in Fig. 1c) one arrives to the vertex  $u_{k,i}$ , analogously  $N_{0,k}(u_{k,j})$  for the vertex  $u_{k,j}$ .

The pairwise comparison function  $Comp(\cdot, \cdot)$  between two connected components  $CC_i^k$  and  $CC_j^k$  can now be defined as:

$$Comp(CC_i^k, CC_j^k) = \begin{cases} \text{True} & \text{if } Ext(CC_i^k, CC_j^k) > PInt(CC_i^k, CC_j^k), \\ \text{False} & \text{otherwise,} \end{cases} \quad (4)$$

where  $PInt(CC_i^k, CC_j^k) = \min\{Int(CC_i^k) + \tau(CC_i^k), Int(CC_j^k) + \tau(CC_j^k)\}$  is the minimum internal contrast difference between two components. For the function  $Comp(CC_i^k, CC_j^k)$  to be true i.e. for the border to exist, the external contrast difference must be greater than the internal contrast differences. The reason for using a threshold function  $\tau(CC^k)$  is that for small components  $CC^k$ ,  $Int(CC^k)$  is not a good estimate of the local characteristics of the data, in extreme case when  $|CC^k| = 1$ ,  $Int(CC^k) = 0$ . Any non-negative function of a single component  $CC^k$ , can be used for  $\tau(CC^k)$ . Choosing criteria other than minimum and maximum will lead to an NP-complete algorithm [3].

## 2.1 Building Hierarchy of Partitions

Let  $P_k = CC_1^k, CC_2^k, \dots, CC_n^k$  be the partition on the level  $k$  of the pyramid. The algorithm to build the hierarchy of partitions is as follows:

---

**Algorithm 1 – Hierarchy of Partitions**

---

**Input:** Attributed graph  $G_0$ .

```
1:  $k = 0$ 
2: repeat
3:   for all vertices  $u \in G_k$  do
4:      $E_{min}(u) = \operatorname{argmin}\{attr_e(e) \mid e = (u, v) \in E_k \text{ or } e = (v, u) \in E_k\}$ 
5:   end for
6:   for all  $e = (u_{k,i}, u_{k,j}) \in E_{min}$  with  $Ext(CC_i^k, CC_j^k) \leq PInt(CC_i^k, CC_j^k)$  do
7:     include  $e$  in contraction edges  $N_{k,k+1}$ 
8:   end for
9:   contract graph  $G_k$  with contraction kernels,  $N_{k,k+1}$ :  $G_{k+1} = C[G_k, N_{k,k+1}]$ .
10:  for all  $e_{k+1} \in G_{k+1}$  do
11:    set edge attributes  $attr_e(e_{k+1}) = \min\{attr_e(e_k) \mid e_{k+1} = C(e_k, N_{k,k+1})\}$ 
12:  end for
13:   $k = k + 1$ 
14: until  $G_k = G_{k-1}$ 
```

**Output:** A region adjacency graph (RAG) pyramid.

---

Each vertex  $u_k \in G_k$  i.e.  $CC^k$  represents a connected region on the base level of the pyramid, and since the presented algorithm is based on Borůvka's algorithm [2], it builds a  $MST(u_k)$  of each region, i.e.  $N_{0,k}(u_k) = MST(u_k)$  [7]. The idea is to collect the smallest weighted edges  $e$  (4th step) that could be part of the  $MST$ , and then to check if the edge weight  $attr_e(e)$  is smaller than the internal contrast of both of the components ( $MST$  of end vertices of  $e$ ) (6th step). If these conditions are fulfilled then these two components will be merged (7th step). Two regions will be merged if the internal contrast, which is represented by its  $MST$ , is larger than the external contrast, represented by the weight of the edge,  $attr_e(e)$ . All the edges to be contracted form the contraction kernels  $N_{k,k+1}$ , which are then used to create the graph  $G_{k+1} = C[G_k, N_{k,k+1}]$  [14], so that the topology is preserved. In general  $N_{k,k+1}$  is a forest. We update the attributes of those edges  $e_{k+1} \in G_{k+1}$  with the minimum attribute of the edges  $e_k \in E_k$  that are contracted into  $e_{k+1}$  (11th step). The output of the algorithm is a pyramid where each level represents a RAG, i.e. a partition. Each vertex of these RAGs is the representative of a  $MST$  of a region in the image. The algorithm is greedy since it collects only the nearest neighbor with the minimal edge weights and merges them if Eq. 4 is false.

**Proposition 1.** *For any connected attributed graph  $G(V, E, attr_e, attr_v)$ , Alg. 1 produces a **hierarchy** over  $V$ .*

**Proof.** All individual vertices  $v \in V$  on the base level form a partition. It is only needed to check that partitions are partially ordered by the inclusion relation. Assume this is not the case, i.e.  $\exists(CC_i^k, CC_j^k) \in P_k$  such that  $CC_i^k \cap CC_j^k \neq \phi$  but neither  $C_i^k \subset C_j^k$  nor  $C_j^k \subset C_i^k$ . There are at least two edges,  $e'$  connecting  $CC_i^k$  and  $CC_j^k \setminus CC_i^k$  and the other edge  $e''$  connecting  $CC_j^k$  and  $CC_i^k \setminus CC_j^k$ , from which it follows that  $CC_i^k \in P_k \Rightarrow PInt(CC_i^k, CC_j^k) < Ext(CC_i^k, CC_j^k) = attr_e(e')$ , and for the edge  $e''$  one shows that  $attr_e(e'') = Ext(CC_j^k, CC_i^k) \leq$

$PInt(CC_j^k, CC_i^k)$ , since  $PInt(CC_j^k, CC_i^k) = PInt(CC_i^k, CC_j^k)$  (Eq. 2) and  $e'' \in CC_i^k$  it follows  $Ext(CC_j^k, CC_i^k) \leq PInt(CC_i^k, CC_j^k) < Ext(CC_i^k, CC_j^k) \leq PInt(CC_j^k, CC_i^k) \Rightarrow CC_j^k \notin P_k$ , contradicting the assumption  $CC_j^k \in P_k$ .  $\square$

**Proposition 2.** *For any connected attributed graph  $G(V, E, attr_e, attr_v)$ , Alg. 1 produces the **partitions** which are **invariant** under any monotone transformation of the  $attr_e$  (dissimilarity measure).*

**Proof.** It should be checked that the order by which the edges are contracted is not changed by a monotone transformation. The monotone transformation does not change the total order of edges incident on a vertex. This implies that the edge with the minimum weight is also not changed after this monotone transformation in the 4th step of Alg. 1. Moreover this transformation does not change the total order of the edges in a connected component  $CC_i^k$  and  $CC_j^k$ , implying that the minimum of maximum edge weight of the  $CC_i^k$  and  $CC_j^k$  is on the same edge (7th step). Edges marked in the 4th and 7th step of the Alg. 1 are not changed by the transformation, which results in the invariance of the partitions.  $\square$

**Proposition 3.** *For any connected attributed graph  $G(V, E, attr_e, attr_v)$ , the **hierarchy** over  $V$  is **invariant** under monotone transformation of attributes.*

**Proof.** The proof is straightforward using Prop. 2.  $\square$

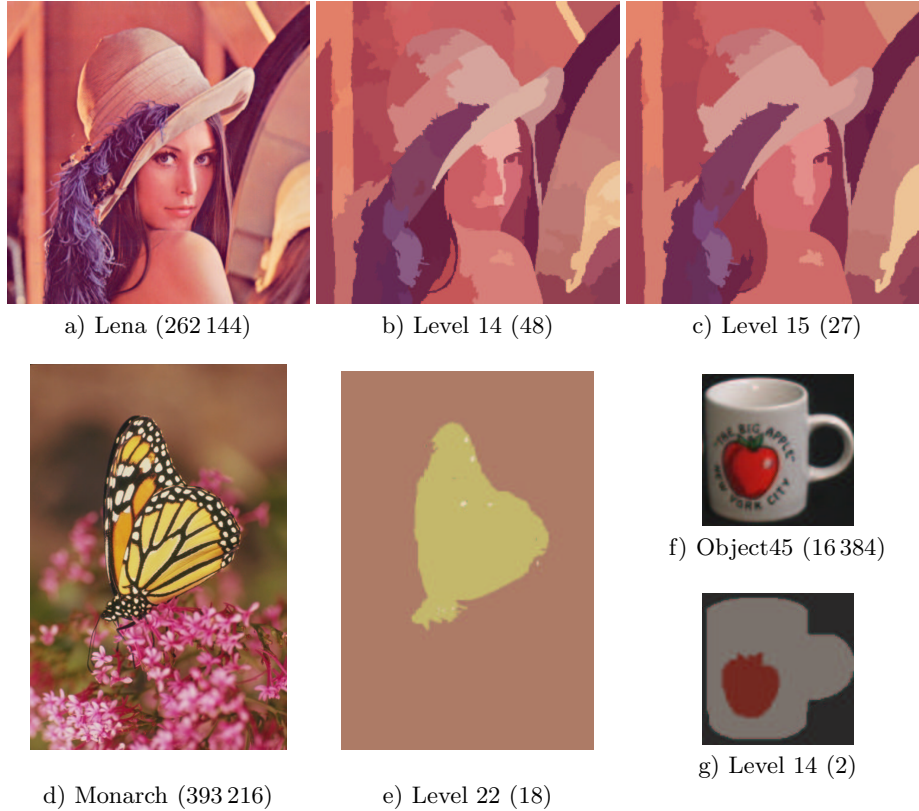
### 3 Experiments on Image Graphs

We attribute edges with the intensity difference  $att_e(u_i, u_j) = |I(u_i) - I(u_j)|$ , where  $I(u_i)$  is the intensity of the pixel  $p_i$ . For color images we run the algorithm by computing the distances in color space. To compute the hierarchy of partitions the function  $\tau(CC^k) = f(CC^k)$  is defined as  $\tau(CC^k) = \alpha/|CC^k|$ , where  $\alpha = const$  and  $|CC^k|$  is the number of elements in  $CC^k$ , i.e. the size of the region. The algorithm has one running parameter  $\alpha$ . A larger constant  $\alpha$  sets the preference for larger components. A more complex definition of  $\tau(CC^k)$ , which is large for certain shapes and small otherwise, would produce a partition which prefers certain shapes. To speed up the computation, vertices are attributed ( $attr_v$ ) with the internal differences, average color and the size of the region it represents. Each of these attributes is computed for each level of the hierarchy. Note that the height of the pyramid depends only on the image content.

We use indoor *RGB* images 'Lena'<sup>1</sup>(512 × 512) and 'Object 45'<sup>2</sup>(128 × 128) and an outdoor image 'Monarch'<sup>1</sup>(768 × 512) for experiments. We found that  $\alpha = 300$  produces the best hierarchy of partitions of the images shown in Fig. 2. Fig. 2b,c,e,h show some of the partitions on different levels of the pyramid and the number of components. In all of the images there are regions of large intensity variability and gradient. This algorithm is capable of grouping perceptually important regions despite of large intensity variability and gradient. Since the algorithm preserves details in low-variability regions, a noisy pixel would survive

<sup>1</sup> Waterloo image database

<sup>2</sup> Coil 100 image database



**Fig. 2.** Some levels of the partitioning produced with  $\alpha = 300$ .

throughout the hierarchy (Fig.2e). Image smoothing in low variability regions would overcome this problem. We do not smooth the images, because that would introduce another parameter in the method. The hierarchy of partitions can also be built from an oversegmented image to overcome the problem of noisy pixels. Note that the influence of  $\tau$  in the decision criterion is smaller as the region gets bigger. For an oversegmented image the algorithm becomes parameterless.

## 4 Conclusion and Outlook

In this paper we have introduced a method to build hierarchical partitions of an image by comparing in a pairwise manner the difference along the boundary of two components relative to the differences of component's internal differences. Even though the algorithm makes simple greedy decisions locally, it produces perceptually important partitions in a bottom-up 'stimulus-driven' way based only on local differences. It was shown that the algorithm can handle large variation and gradient intensity in images. Since our framework is general enough, we

can use RAGs of any oversegmented image and build the hierarchy of partitions. External knowledge can help in a top-down segmentation technique. A drawback is that the maximum and minimum criterion is very sensitive to noise, although in practice it has a small impact. Other criteria, such as median, would lead to an NP-complete algorithm. The algorithm has only one running parameter which controls the sizes of the regions.

## References

1. M. Bister, J. Cornelis, and A. Rosenfeld. A critical view of pyramid segmentation algorithms. *Pattern Recognition Letters*, Vol. 11(No. 9):p.605–617, 1990.
2. O. Borůvka. O jistém problému minimálním. *Práce Mor. Přírodověd. Spol. v Brně (Acta Societ. Scienc. Natur. Moraviae)*, (3):p.37–58, 1926.
3. P. F. Felzenszwalb and D. P. Huttenlocher. Image Segmentation Using Local Variation. In *Proc. of IEEE Conf. on CVPR*, p.98–104, 1998.
4. B. Fischer and J. M. Buhmann. Data Resampling for Path Based Clustering. *Proc. 24th DAGM Symp.*, LNCS 2449 p.206–214, 2002.
5. L. Guigues, L. M. Herve, and J.-P. Cocquerez. The Hierarchy of the Cocoons of a Graph and its Application to Image Segmentation. *Patt. Recog. Lett.*, 24(8) p.1059–1066, 2003.
6. Y. Haxhimusa, R. Glantz, M. Saib, G. Langs, and W. G. Kropatsch. Logarithmic Tapering Graph Pyramid. *Proc. 24th DAGM Symp.*, LNCS 2449, p.117–124, 2002.
7. Y. Haxhimusa and W. G. Kropatsch. Hierarchical Image Partitioning with Dual Graph Contraction. *Technical Report PRIP-TR-81*, TU Wien, Austria 2003, <http://www.prip.tuwien.ac.at>.
8. S. Horowitz and T. Pavlidis. Picture Segmentation by a Tree Traversal Algorithm. *J. Assoc. Compt. Math.*, Vol. 2(23):p.368–388, 1976.
9. J.-M. Jolion and A. Montanvert. The adaptive pyramid, a framework for 2D image analysis. *Comp. Vis., Graph., and Im. Process.*, 55(3):pp.339–348, 1992.
10. J.-M. Jolion and A. Rosenfeld. *A Pyramid Framework for Early Vision*. Kluwer Academic Pub., 1994.
11. V. A. Kovalevsky. Finite topology as applied to image analysis. *Comp. Vis., Graph., and Imag. Process.*, Vol. 46:pp.141–161, 1989.
12. W. G. Kropatsch. Building Irregular Pyramids by Dual Graph Contraction. *IEE-Proc. Vision, Image and Signal Processing*, Vol. 142(No. 6):pp.366–374, 1995.
13. W. G. Kropatsch and S. BenYacoub. A general pyramid segmentation algorithm. *Intl. Symp. on Opt. Scie., Eng., and Instr.*, Vol. 2826:p.216–224. SPIE, 1996.
14. W. G. Kropatsch, A. Leonardis, and H. Bischof. Hierarchical, Adaptive and Robust Methods for Image Understanding. *Surv. on Math. for Ind.*, No.9:p.1–47, 1999.
15. P. Meer. Stochastic image pyramids. *Computer Vision, Graphics, and Image Processing*, Vol. 45(No. 3):pp.269–294, 1989.
16. P. Meer, D. Mintz, A. Montanvert, and A. Rosenfeld. Consensus vision. In *AAAI-90 Workshop on Qualitative Vision*, p. 111–115, 1990.
17. A. Rosenfeld, editor. *Multiresolution Image Processing and Analysis*. Springer, Berlin, 1984.
18. J. Shi and J. Malik. Normalized Cuts and Image Segmentation. In *Proceedings IEEE Conf. Comp. Vis. and Patt.*, pp:731–737, 1997.
19. M. Wertheimer. Über Gestaltheorie. *Philosophische Zeitschrift für Forschung und Aussprache*, 1:30–60, 1925.