# REPRESENTATION OF IMAGE STRUCTURE BY A PAIR OF DUAL GRAPHS[1)]

R. Glantz and R. Englert and W. Kropatsch

Institute for Computer Aided Automation

Pattern Recognition and Image Processing Group 183/2

Vienna University of Technology, Austria

Phone: +43 1 58801 1835-{8,6,0} Fax: +43 1 58801 18391

e-mail: {glz,englert,krw}@prip.tuwien.ac.at

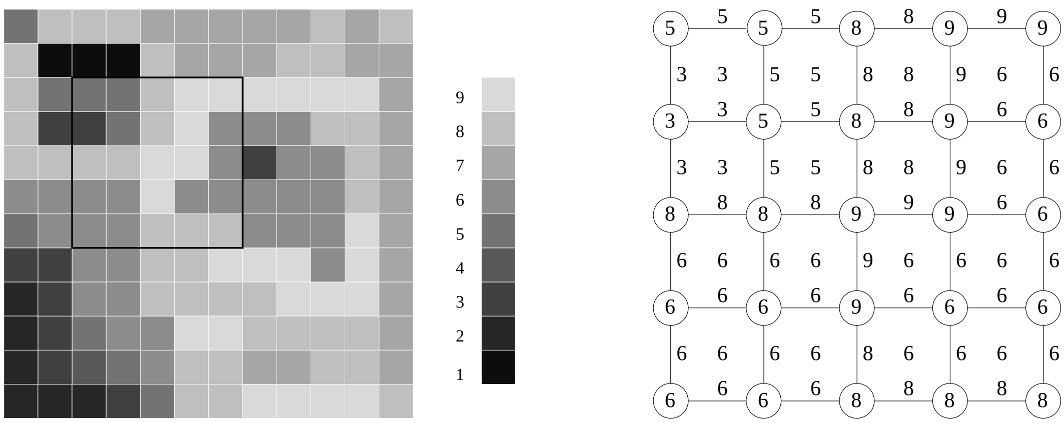
*Abstract*

*Human image understanding copes with even strong gray level transformations, as long as the ordering of the gray levels with respect to neighboring pixels is nowhere disturbed. In this paper, we study dual graph contractions, which do not depend on these gray level transformations either. The result of the dual graph contractions is a pair of dual attributed graphs. One of them describes neighborhood relations of the local maxima in the image, the other one does the same for the local minima. In contrast to the well known watersheds, the two graphs also express hierarchies of nested bright and dark blobs in the image. The concept of duality for graphs is coupled with the reversal of the attribute order in the contraction rules. The contractions are characterized by the preservation of a connectivity property taking into account the gray levels.*

## 1 Introduction

In this paper the rigid arrangement of the pixels in a two-dimensional digital gray level image is replaced by a pair of dual graphs adapted to the gray levels. Throughout the paper we refer to the intuitive interpretation of a gray level image as a digital elevation model (see Figure 3(a)), where the heights are given by the gray levels [KvD94]. The task is as follows: Construct a plane graph, where each summit is represented by exactly one vertex and each edge expresses a neighborhood of hills. In addition, each basin is represented by exactly one face of the graph, i.e vertex of the dual graph. The edges of the dual graph stand for neighborhood relations of the basins. For the gray level image in Figure 1(a) such a graph is depicted in Figure 3(b). The first step of constructing the dual pair consists in the generation of a so called image graph, where each vertex represents exactly one pixel and the edges connect the 4-connected pixels (see Figure 1(b)).

[1)]This work is supported by the Austrian Science Foundation (FWF) under grant S7002-MAT.




(a) (b)

**Figure 1:** (a) The gray levels of the pixels. (b) The image graph of the marked sub-image (inside the black border), where the vertices are represented as circles. The numbers in the vertices and at the edges indicate the vertex values and the edge values of the image graph. The numbers in the middle of the square faces refer to the vertices of the dual graph.

The plan of the paper is as follows: In order to obtain the final pair from the image graph and its dual by contractions, we first provide the image graph and its dual with attribute values for the vertices and the edges. The choice of these initial values is motivated by a sketch of the contractions we want to apply. This is done in Section 2. The framework for the contractions proposed is the concept of dual graph contraction presented in Section 3. In Section 4 we then specify the rules for the contractions leading to the final pair. These contractions turn out to preserve strong properties of the graphs, which are formulated and whose preservation is proven in Section 5. In Section 6 we conclude and give an outlook.

## 2 Initialization of the Image graph and its Dual

Initially an image graph $G$ is created, which reflects the rigid arrangement of the pixels. Each pixel of the graph is represented by a vertex $v$, the attribute value $val(v)$ of which (short: vertex value) indicates the gray level of the pixel. In Figure 1(b) the vertices are represented by circles. The corresponding vertex value is written in the circle. Each pair of 4-connected pixels gives rise to an undirected edge $e = (v_1, v_2)$, where $v_1$ and $v_2$ are the vertices representing the pixels. In the following, we sketch the main idea of obtaining the final pair by dual graph contraction [Kro95a]. This will also explain the choice of the values associated with the edges and faces of Figure 1(b)). Since, at the end, each vertex of $G$ is to represent exactly one summit, it is intuitive to contract edges, such that the end vertex with the maximal value survives. However, the contraction must not lead to the unification of separate hills. We now think of an edge as representing a path in the landscape. If the height of the lowest point along this path is between the heights of the end points, we may contract the edges without the risk of unifying two separate hills. Hence, we want each edge to be associated with the minimal height of the path that is represented by the edge: the attribute value $val(e)$, short *edge value*, of edge $e = (v_1, v_2)$ is thus initialized to the minimum of the values $val(v_1)$ and $val(v_2)$.

We now focus on the contraction of the dual of $G$ denoted by $\overline{G}$. Recall, that the contraction of an edge $\overline{e}$ in $\overline{G}$ goes along with the removal of the corresponding dual edge in $G$ (see [Kro99]). In terms of watersheds, (see [MR98]) it is intuitive to fuse the faces of $G$ until each of the resulting faces corresponds to exactly one basin of the landscape. Two faces sharing an edge may be fused, if the edge does not represent a barrier. Now the term *barrier* has to be specified: Again we think of the edge as representing a path in the landscape. If the lowest height along the path (i.e. the edge value) is between the lowest heights in each of two faces, the edge does not represent a barrier. Thus, for each face we have to know the height of its deepest point.

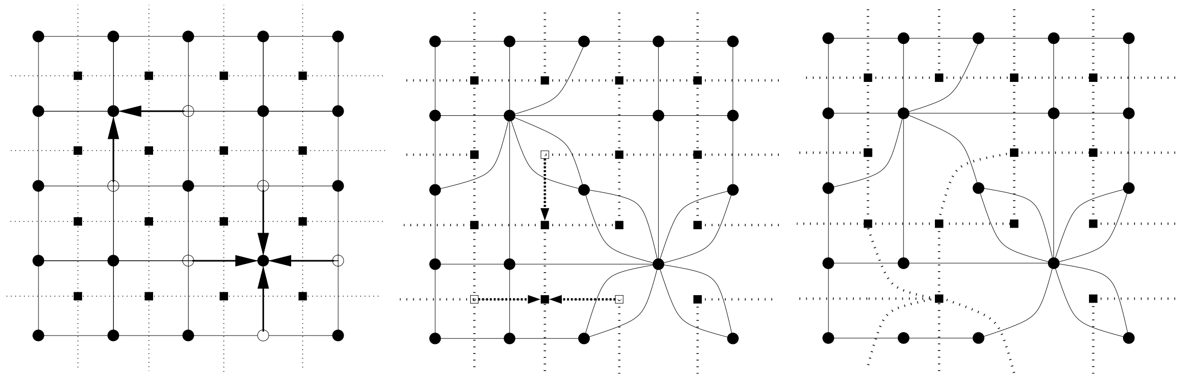
Each face of $G$ is represented by exactly one vertex of the dual graph of $G$, i.e. $\overline{G}$. The depths of the faces may thus be stored as attribute values of the vertices of $\overline{G}$. The attribute value $val(\overline{v})$, short *vertex value*, of the vertex $\overline{v}$ is initialized to the minimum of all values $val(u)$, where $u$ is a vertex on the boundary of the corresponding face. This initialization includes the infinite background face.

To complete the initialization, we assign values to the edges $\overline{e} = (\overline{v_1}, \overline{v_2})$ of the graph $\overline{G}$. The goal is to express the fusion of two adjacent faces by a contraction in $\overline{G}$ similar to the contractions in $G$. This is achieved by setting the value of $\overline{e}$ to the value of the corresponding dual edge $e$ in $G$. Like in the graph $G$ an edge of $\overline{G}$ may be contracted only if the edge value is between the values of the end vertices. In $\overline{G}$, however, the value of the surviving vertex has to be smaller or equal to the value of the non-surviving vertex.

This initialization of the pair $(G, \overline{G})$ yields the following properties. The value of a vertex in $G$ is always greater or equal to the values of the adjacent edges. In $\overline{G}$ the value of a vertex is always smaller or equal to the values of the adjacent edges. We suggest the names *max-graph* and *min-graph* for graphs with these properties. The sketched contractions always transform a dual pair consisting of a max-graph and a min-graph into another dual pair to consist of a max-graph and a min-graph.

## 3 Dual Graph Contraction

The contractions sketched in the previous section were contractions of single edges. This might lead to the false impression of a sequential process. In this section we present a terminology and a data structure that allow for the specification of contractions performed in *parallel*. Consider a subtree of edges to be contracted, such that the surviving vertex is always the same.



(a) (b) (c)

**Figure 2:** Part (a) shows a plane graph (vertices = '•', edges = '–') and its dual graph (vertices = '▪', edges = '···'), where the vertex representing the background face is omitted for sake of simplicity. The contraction kernels are marked ('∘' are non-surviving vertices and '→' point at the surviving vertices). Part (b) shows the result of the dual edge contraction and two contraction kernel of the dual graph ('□' are non-surviving vertices and '··>' point at the surviving vertices). Part (c) shows the result of the dual face contraction.

Obviously these edges may be contracted in parallel. The same applies to a collection of such trees, where the trees are disjoint (no vertex is contained in more than one tree). Including trees to consist of a single vertex each, we may specify a parallel contraction by a spanning forest, the trees of which have depths $\leq 1$. This concept can be extended to specify a whole sequence of parallel contractions (see [Kro95b]). The depths of the trees may then be greater than one:

**Definition 3.1 (Decimation Parameter, Contraction Kernel)**
*Given a graph $G$. A subgraph $D_e$ of $G$ is a* DECIMATION PARAMETER *of $G$, if and only if $D_e$ is a spanning forest of $G$. The connected components (trees) of $D_e$ are called* CONTRACTION KERNELS. *The roots of the trees are called* SURVIVING VERTICES. *All other vertices of the trees are called* NON-SURVIVING VERTICES.

As sketched in the previous section, we wish to contract also the dual $\overline{G}$ of $G$. Therefore we require $G$ to be *plane*. The decimation parameter for the contraction of $\overline{G}$ is denoted by $D_f$. The vertex representing the background face must always survive. The indices $e$ and $f$ of the decimation parameters refer to the two phases, i.e dual edge contraction and dual face contraction, of the dual graph contraction (see again [Kro95b]):

**Definition 3.2 (Dual Graph Contraction)**
*Given an embedded pair of dual graphs $(G, \overline{G})$ and decimation parameters $D_e$ and $D_f$ for contractions in $G$ and in $\overline{G}$ respectively.* DUAL GRAPH CONTRACTION *consists of the two following phases:*

1. DUAL EDGE CONTRACTION *described by a function* $C_e : (G, \overline{G}) \rightarrow C_e[(G, \overline{G}), D_e] = (G', \overline{G'})$.

2. DUAL FACE CONTRACTION $C_f : (\overline{G'}, G') \rightarrow C_f[(\overline{G'}, G'), D_f] = (\overline{G''}, G'')$.

Figures 2(a) and 2(b) demonstrate an example of the dual edge contraction. The contraction kernels for the dual edge contraction are shown in Figure 2(a). The result of the contraction is shown in Figure 2(b). During the contraction a non-surviving vertex is *identified* with a surviving vertex which is the root of the contraction kernel. Note, that the contraction of an edge goes along with the deletion of the corresponding dual edge in $\overline{G}$ (see Figure 2(b)). Afterwards the dual face contraction is executed (Figures 2(b) and 2(c)). Note, that the contraction of an edge in $\overline{G}$ goes along with the deletion of the corresponding dual edge in $G$ (see Figure 2(c)). Applications for the dual graph contraction are described in [Kro95a, GEK99].

## 4 Monotonic Dual Graph Contraction

The dual graph contractions defined in Section 3 provide a very large class of graph transformations. One way to specify a subclass is to formulate conditions for edges, which may be contracted. In the following, the conditions are formulated with respect to the values of the edges and the values of the end vertices of the edges. The resulting dual graph contractions are called *monotonic*.

Let $e = (v_1, v_2)$ denote an edge of the image graph $G$. As sketched in Section 2 the edge $e$ may be contracted only if

$$min\{val(v_1), val(v_2)\} \leq val(e) \leq max\{val(v_1), val(v_2)\}. \tag{1}$$

If the condition is fulfilled, the edge is called *contractible*. As motivated in Section 2 the value of the surviving vertex has to be greater or equal to the value of the non-surviving vertex. If $val(v_1) = val(v_2)$, the surviving vertex is chosen randomly. Without loss of generality let $v_2$ denote the non-surviving vertex of a contractible edge $e = (v_1, v_2)$. Let $e_{old} \neq e$, $e_{old} = (v_2, v_3)$ be another edge adjacent to the vertex $v_2$. After the contraction of $e$, there will be a new edge $e_{new} = (v_1, v_3)$, which needs a new value. As motivated in Section 2, we set the value of $e_{new}$ to

$$val(e_{new}) := min\{val(e), val(e_{old})\} = val(e_{old}). \tag{2}$$

The dual face contractions are defined symmetrically: The edge $\overline{e} = (\overline{v_1}, \overline{v_2})$ may be contracted, if

$$min\{val(\overline{v_1}), val(\overline{v_2})\} \leq val(\overline{e}) \leq max\{val(\overline{v_1}), val(\overline{v_2})\}. \tag{3}$$

If the condition is fulfilled, the edge is called *contractible*. The value of the surviving vertex has to be smaller or equal to the value of the non-surviving vertex. Without loss of generality let

$\overline{v_2}$ denote the non-surviving vertex. Let $\overline{e_{old}} \neq \overline{e}$, $\overline{e_{old}} = (\overline{v_2}, \overline{v_3})$ be another edge adjacent to the vertex $\overline{v_2}$. After the contraction of $\overline{e}$, there will be a new edge $\overline{e_{new}} = (\overline{v_1}, \overline{v_3})$, which receives the value

$$val(\overline{e_{new}}) := max\{val(\overline{e}), val(\overline{e_{old}})\} = val(\overline{e_{old}}). \tag{4}$$

Recall, that the vertex representing the background must always survive. In the initial dual graph the values of the edges bounding a face $\overline{v}$ are all greater or equal to $val(\overline{v})$. Obviously, this property is preserved by the monotonic dual graph contraction:

$$val(e_b) \geq val(\overline{v}) \text{ for all edges } e_b \text{ on the boundary of } \overline{v}. \tag{5}$$

Applying the monotonic dual graph contraction to the image graph of Figure 1(a), a part of which is shown in Figure 1(b), yields the graph shown in Figure 3(b). None of the edges of the graph and its dual are contractible anymore. The vertices of the corresponding dual are drawn as boxes. The boxes contain the values of the corresponding vertices. Comparing the graph (Figure 3(b)) with the digital elevation model of Figure 3(a), we emphasize that the nested craters on the upper left are represented as nested cycles in the graph.

A drawback of our concept goes back to the fact, that the saddles in the digital elevation model are not represented as vertices. We cannot properly describe saddles which lead to more than two summits, if one follows the ascending crest lines (see Figure 4). Figure 4 also makes clear, that the final graph is not unique. The bent edge might as well be situated at the left side. Our future work will aim at the proper representation of the saddles. For this purpose we will have a closer look at the contraction kernels.

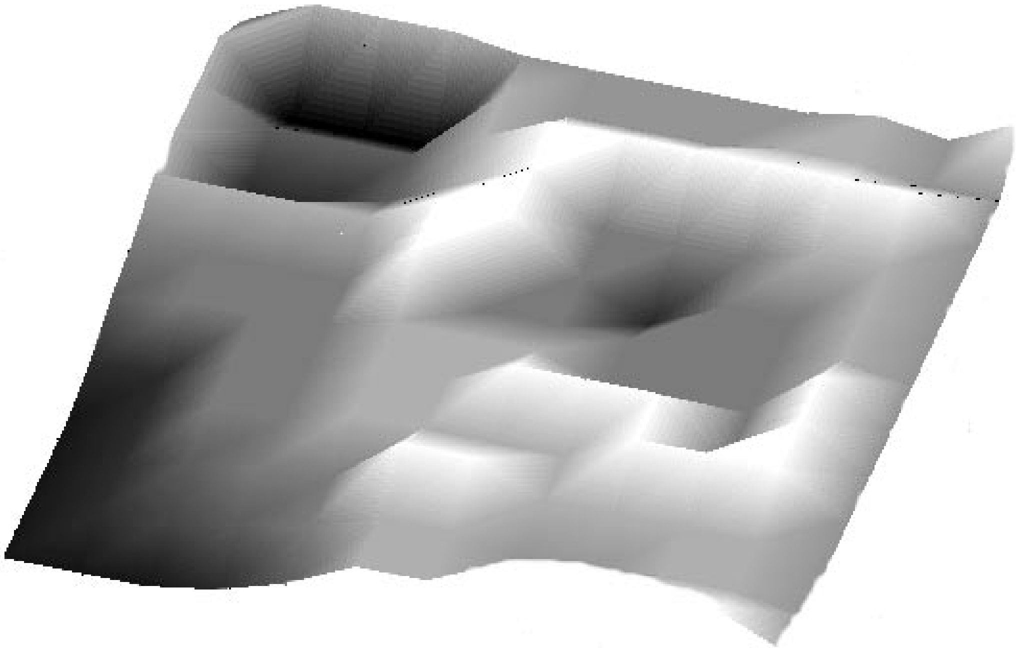
# 5 Connectivity Levels

Consider two points in a landscape. A map of an area containing the two points should at least roughly contain the information about the highest altitude, which one needs to descend to, if going from one point to the other. This critical height is now defined for the image graph $G$:

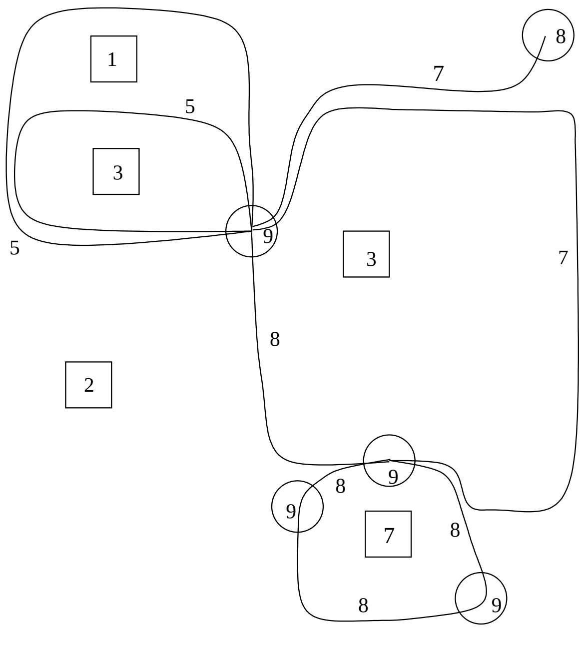
**Definition 5.1 (Connectivity Levels, *minval*)**
*Let $\Pi_G(v, w)$ denote the set of all paths in $G$, which connect the vertices $v$ and $w$. For each path $P \in \Pi_G(v, w)$ set $minval(P) := min\{val(e) \mid e \text{ is an edge on } P\}$. The connectivity level $C_G(v, w)$ between $v$ and $w$ in $G$ is defined as: $C_G(v, w) := max\{minval(P) \mid P \in \Pi_G(v, w)\}$.*

We prove that the monotonic dual graph contraction preserves the connectivity levels:

**Theorem 5.2 (Preservation of Connectivity Levels)**
*Let $G' = (V', E')$ denote a graph, which was obtained from $G = (V, E)$ by monotonic dual graph contractions. Let $v'$ and $w'$ denote two vertices of $V'$ (i.e. vertices of $G$ that survived the monotonic dual contractions). The following equation holds: $C_{G'}(v', w') = C_G(v', w')$.*



(a)




(b)

**Figure 3:** (a) Digital elevation model of the gray levels from Figure 1(a) (interpolated). (b) Corresponding final graph (vertices = '∘', undirected (curved) edges = '—') and the surviving vertices '□' of the dual (displayed without edges).

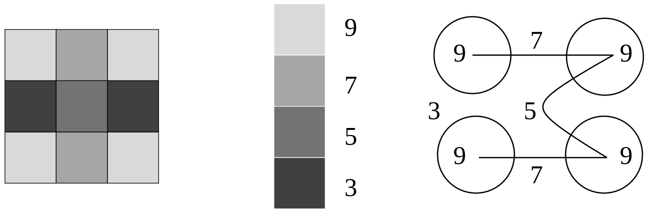



**Figure 4:** The gray levels of the pixels (left) and the final graph representing the image (right).

**Proof of 5.2:** We first introduce the following notations: Let $e_{con} = (v_1, v_2)$ denote a contractible edge of $E$ (see condition 1 in Section 4). Without loss of generality we assume that $val(v_1) \geq val(v_2)$. Let $G/\{e_{con}\}$ denote the graph, which is obtained from $G$ by contracting $e_{con}$, such that $v_1$ is the surviving vertex.
Let $\overline{e_{del}} = (\overline{v_1}, \overline{v_2})$ denote a contractible edge of $\overline{E}$ (see condition 3 in Section 4). The contraction of $\overline{e_{del}}$ in $\overline{G}$ goes along with the deletion of the corresponding edge $e_{del}$ in $G$. Let $G \backslash \{e_{del}\}$ denote the graph, which is obtained from $G$ by deleting $e_{del}$.

The proof is organized as follows:

- (a): prove the theorem for $G' = G/\{e_{con}\}$,
- (b): prove the theorem for $G' = G \backslash \{e_{del}\}$,
- (c): prove the theorem for all $G'$ using (a) and (b).

In order to prove (a) we consider the set $E_{old} \subset E$, $E_{old} := \{e_{old} = (v_2, v_3)$ for some vertex $v_3 \in V, e_{old} \neq e_{con}\}$ and the set of paths $\Pi_{old} := \{(e_{con}, e_{old}) \mid e_{old} \in E_{old}\}$. Each path $\pi = (e_{con}, e_{old})$ of $\Pi_{old}$ corresponds to exactly one edge $e_{new}$ in $E' \backslash E$ after the contraction of $e_{con}$. From 2 in Section 4 follows $val(e_{new}) = val(e_{old})$. Since $val(e_{con}) \geq val(v_2) \geq val(e_{old})$, we have $minval(\pi) = val(e_{old})$. Thus, there is a one-to-one correspondence between all paths in $G$ with end points other than $v_2$ and all paths in $G' = G/\{e_{con}\}$, such that corresponding paths have the same minimal edge value. This completes the proof of (a).

We now prove (b): Since the removal of an edge does not increase any connectivity level, it suffices to show that no connectivity level is decreased by the removal of $e_{del}$.
Let $(v', w')$ denote a pair of surviving vertices in $G' = G \backslash \{e_{del}\}$. Then there exists a path $P$ between $v'$ and $w'$ in $G'$, such that $minval(P) = C_G(v', w')$. We only have to consider the case that $e_{del}$ is an edge of $P$ and may assume that $\overline{v_1}$ is the surviving vertex in $\overline{G'}$. It suffices to show that $val(e_{del})$ is minimal with respect to all edges which bound the face represented by $\overline{v_2}$ (see Figure 5). In this case there exists a detour of $e_{del}$ with a minimal edge value greater or equal to $C_G(v', w')$. This would complete the proof of (b). Hence, to prove (b) we only have to show that $val(e_{del})$ is minimal with respect to all edges, which bound the face represented by $\overline{v_2}$: From 5

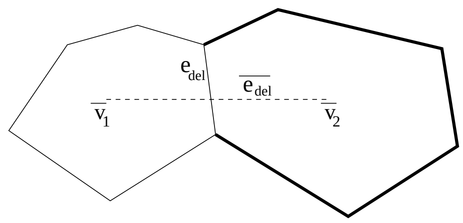



**Figure 5:** The faces of $\overline{v_1}$ and $\overline{v_2}$.

in Section 4 we know that each of the edges bounding the face represented by $\overline{v_2}$ has an edge value greater or equal to $val(\overline{v_2})$. On the other hand, we have $val(e_{del}) = val(\overline{e_{del}}) \leq val(\overline{v_2})$, since $\overline{v_2}$ is the non-surviving vertex. This finishes the proof of (b).

The proof of (c) follows from the fact that each monotonic dual graph contraction can be performed by a sequence of contractions treated in (a) or in (b). $\square$

A theorem analogous to Theorem 5.2 holds for the dual graph.

## 6 Conclusions and Outlook

In this paper we have proposed a new approach to the computation of image structure for gray level images. The structure is represented by a pair of attributed dual graphs. This pair is constructed by parallel dual graph contractions respecting the monotonicity of the gray levels. The contractions preserve the connectivity levels of the surviving vertices. Our representation of gray level images can express structures like nested gray level basins. Future work will focus on a representation of image structure, where the saddles are represented as vertices.

## References

[GEK99] R. Glantz, R. Englert, and W. G. Kropatsch. Contracting Distance Maps of Pores to Pore Networks. In *Proc. 4th Computer Vision Winter Workshop*, pages 112 – 121, 1999.

[Kro95a] W. G. Kropatsch. Building Irregular Pyramids by Dual Graph Contraction. *IEE-Proc. Vision, Image and Signal Processing*, 142(6):366 – 374, 1995.

[Kro95b] W.G. Kropatsch. Equivalent Contraction Kernels and The Domain of Dual Irregular Pyramids. Technical Report PRIP-TR-42, Institute f. Automation 183/2, Dept. for Pattern Recognition and Image Processing, TU Wien, Austria, 1995.

[Kro99] W.G. Kropatsch. Dual Contraction of Combinatorial Maps. Technical Report PRIP-TR-54, Institute f. Automation 183/2, Dept. for Pattern Recognition and Image Processing, TU Wien, Austria, 1999.

[KvD94] J. Koenderink and A. van Doorn. Image structure. In E. Paulus and F. Wahl, editors, *Mustererkennung 1997*, pages 401–408. Springer, 1994.

[MR98] A. Meijsler and J. Roerdink. A Disjoint Set Algorithm for the Watershed Transform. In *Proc. of EUSIPCO'98, IX European Signal Processing Conference*, pages 1665 – 1668, Rhodes, Greece, 1998.