CrossMark

SPECIAL ISSUE PAPER

Topology-based image segmentation using LBP pyramids

Martin Cerman[1] · **Ines Janusch**[1] · **Rocio Gonzalez-Diaz**[2] · **Walter G. Kropatsch**[1]

Abstract In this paper, we present a new image segmentation algorithm which is based on local binary patterns (LBPs) and the combinatorial pyramid and which preserves structural correctness and image topology. For this purpose, we define a codification of LBPs using graph pyramids. Since the LBP code characterizes the topological category (local max, min, slope, saddle) of the gray level landscape around the center region, we use it to obtain a "minimal" image representation in terms of the topological characterization of a given 2D grayscale image. Based on this idea, we further describe our hierarchical texture aware image segmentation algorithm and compare its segmentation output and the "minimal" image representation.

Keywords Local binary patterns · Irregular graph pyramid · Primal and dual graph · Topological characterization · Image segmentation

✉ Ines Janusch
ines@prip.tuwien.ac.at

Martin Cerman
mcerman@prip.tuwien.ac.at

Rocio Gonzalez-Diaz
rogodi@us.es

Walter G. Kropatsch
krw@prip.tuwien.ac.at

[1] PRIP group, TU Wien, Vienna, Austria

[2] Applied Math Department, School of Computer Engineering, University of Seville, Seville, Spain

1 Introduction

Given a grayscale digital image I, the local binary pattern LBP(I) [19,21] is again a grayscale digital image which represents the texture element at each pixel in I. This is currently the most frequently used texture descriptor [15] with outstanding results in applications ranging from object detection [18] to segmentation [4,11] and classification [25,27].

Considering image segmentation, the main idea is that a good segmentation can capture perceptually important regions, which reflect local and/or global properties of the image [24]. These regions can then be used for classification and higher level tasks such as image understanding. Existing segmentation algorithms are based on thresholding, histograms, edge detection, split and merge strategies, watershed transformation or graph partitioning [26].

LBPs have already been used in segmentation approaches in the past: first by Ojala et al. who presented an unsupervised three-phase algorithm in [20]. In this paper, we now present an evaluation of our LBP-based texture aware segmentation [2] using the Berkley segmentation dataset [16]. Contrary to existing LBP-based image segmentation approaches, we do not use histograms of LBPs, but keep the spatial information of the LBPs and consider it in the segmentation process. Typically, the LBP operator is applied to all 3×3 image windows of the considered texture (region). Then the histogram provides the characteristic features of the texture. After training the feature space with the textures of interest, new textures can be classified with very good discrimination. Also for segmentation approaches based on LBPs, histograms of the LBPs were used. However, due the histograms, spatial information is lost in these approaches.

In [3], we proposed a new equivalent LBP encoding which transfers the code from the pixels to the neighbor rela-

tions thus only using one bit per edge of the neighborhood graph. We introduced a way to handle non-well-composed images, as well as a repetitive contraction process leading to a hierarchy of successively smaller graphs—a graph pyramid that preserves the basic topological categories (i.e., extrema, saddle points, plateaus and slopes). This approach on its own already provides a topology-based, hierarchical image segmentation. However, by adapting this approach (i.e., the contraction criteria), we now define a texture aware image segmentation using LBPs in this paper. Moreover, we compare the segmentations obtained by our topologically minimal image representation and by our texture aware image segmentation approach. For other non-LBP-based hierarchical topological segmentation approaches, see, for example, [1,22]. The paper is organized as follows: Section 2 recalls the methodological background. In Sect. 2.1, an overview of needed terms and their equivalent meanings as well as a brief introduction to LBPs is given and the topological category of the gray level landscape around a pixel characterized by its LBP code is defined. Section 2.2 recalls irregular graph pyramids. LBP pyramids are introduced in Sect. 3, and an overview of the minimal graph representation approach is given in Sect. 3.4. In Sect. 4, we present our texture aware segmentation approach that is based on LBP pyramids. Section 5 is devoted to experiments, and Sect. 6 to conclusions and future work.

2 Basic concepts and terminology

In this section, we give a definition of terms used within this paper and provide a short recapitulation of the basic concepts of LBPs and graph pyramids, which is needed as a basis for the presented texture aware segmentation approach and the topological minimal representation of the image. Table 1 gives an overview about terms used within this paper and shows equivalences between them.

Table 1 Overview of terms in the image space, the primal graph and the dual graph and their equivalences

Image	Primal graph	Dual graph
Pixel	Vertex	–
4-Neighbors	Edge	–
Region	Connected set of vertices in the base level	–
Superpixel	Vertex at higher pyramid level	–
Adjacent regions	Edges at higher pyramid level	–
Region boundary	–	Dual cycle

2.1 Local binary patterns (LBP)

The intensity of a pixel $p = (x, y)$, denoted by $g(p)$, is expressed within a given range between a minimum and a maximum, inclusive. Without loss of generality, we suppose that the range is [0, 255]. The standard LBP code [19] is computed for a (center) pixel as in Eq. (1), where P is the number of neighbors, R is the distance between the center pixel and the neighbors (we assume 1 here; therefore, the values of P could only be 8 or 4), c is the center pixel of the operator, and p is the local neighbor indexed by p. The basic operator uses the sign function $s(x) = 1$ if $x \geq 0$ and $s(x) = 0$ otherwise.

$$\text{LBP}_{P,R} = \sum_{p=0}^{P-1} s(g(p) - g(c))2^p. \tag{1}$$

The LBP code characterizes the *topological category* of the gray level landscape around the center pixel. A pixel is a *local maximum* if the LBP code is composed just by 0s. A *local minimum* produces an LBP code only with 1s. Notice, however, that an LBP code consisting of 1 only can also be created by a *plateau* (a region composed by neighboring pixels sharing the same gray value) due to the asymmetry of the sign function s. The LBP code describes a *slope* if there are exactly two transitions from 0s to 1s or 1s to 0s in the code, when traversed circularly. More transitions identify a *saddle*.

In this paper, the 4 neighbors (on its top, bottom, right, left) of each pixel are considered for comparison with the center pixel to obtain the LBP code. That is, follow these 4 pixels along a circle (for example, clockwise), whenever the center pixel's gray value is greater than the neighbor's gray value, i.e., when $g(p) - g(c) < 0$, write 0. Otherwise (i.e., when $g(p) - g(c) \geq 0$), write 1. Example:

$$
\begin{array}{|c|c|c|}
\hline
113 & 240 & 23 \\
\hline
20 & 25 & 12 \\
\hline
15 & 30 & 40 \\
\hline
\end{array}
\Rightarrow
\begin{matrix}
 & 1 & \\
0 & 25 & \mathbf{0} \\
 & 1 &
\end{matrix}
\Rightarrow 101\mathbf{0}
$$

$$\Rightarrow 1 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 0 \cdot 2^0 \Rightarrow 10$$

This 4-neighbor LBP codification has been used for solving problems in image processing and analysis, such as face detection and recognition [9], or iris extraction [8].

2.2 Irregular graph pyramids

A *region adjacency graph* (RAG) $G = (V, E)$ encodes the adjacency of regions in a partition. A vertex $v \in V$ is associated with each region r. Vertices of neighboring regions are connected by edges $e \in E$. Classical RAGs do not contain

any self-loops nor parallel edges. An *extended region adjacency graph* (eRAG) contains self-loops and parallel edges used to encode neighborhood relations to a region completely enclosed by one or more other regions [12]. The *dual* graph of an eRAG G is denoted by $\bar{G} = (\bar{V}, \bar{E})$ (G is said to be the *primal* graph of $\bar{G}$). The edges of $\bar{E}$ represent the boundaries (borders) of the regions encoded by G, and the vertices of $\bar{G}$ represent points where boundary segments meet. G and $\bar{G}$ are planar graphs if they represent a 2D decomposition into regions. There is a one-to-one correspondence between the edges of G and the edges of $\bar{G}$, which induces a one-to-one correspondence between the vertices of G and the 2D cells (regions) of $\bar{G}$. The dual of $\bar{G}$ is again G. The following operations are equivalent: edge contraction in G with edge removal in $\bar{G}$, and edge removal in G with edge contraction in $\bar{G}$. Edge contraction preserves the topology (i.e., regions are always homeomorphic to disks) [7]. A (dual) irregular graph pyramid [12,13] is a stack of successively reduced planar graphs $P = \{(G_0, \bar{G}_0), \ldots, (G_n, \bar{G}_n)\}$. Each level $(G_k, \bar{G}_k)$, $0 < k \le n$, is obtained by first contracting edges in selected contraction kernels of G_{k-1}, if the corresponding regions should be merged, and then removing edges in G_{k-1} to simplify the structure.

In this paper, pixels are considered unit square regions, 4-neighborhood is used for constructing the RAG, and each vertex v of the RAG associated with each region r is labeled with the gray value of the region, i.e., $g(v) := g(r)$.

3 LBP pyramids

In this section, we define a codification of local binary patterns (LBP) based on graph pyramids. The notion of well-composed images and a way how to obtain well-composed images by inserting dummy regions is described first, to be able to deduce the topological characterization provided by $\mathrm{LBP}_{8,1}$ from the topological characterization of the LBP code

of the modified configuration. In order to obtain a minimal graph representation of a given image I in terms of topological characterization of the LBP code of I, we first merge regions having the same gray value (plateaus) and finally obtain a minimal graph representation by merging singular slopes (a particular topological category that can be merged without changing the topological category of the remaining regions).

3.1 Creating well-composed images

A 2D image is *well-composed* [14] if it does not contain the following *non-well-composed configuration* (modulo reflection and 90° rotation):

$$\begin{array}{|c|c|} \hline a & b \\ \hline c & d \\ \hline \end{array} \quad \text{where } g(a) < g(b),\, g(a) < g(c),$$
$$g(d) < g(b) \text{ and } g(d) < g(c). \tag{2}$$

Lemma 1 [3] *If the image is well composed, the topological category provided by* $\mathrm{LBP}_{4,1}$ *is the same as* $\mathrm{LBP}_{8,1}$.

$$\begin{array}{|c|c|c|} \hline a & & b \\ \hline & r & \\ \hline c & & d \\ \hline \end{array}$$
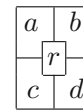
The main problem in a non-well-composed configuration like Eq. (2) is that the relation between a and d (resp. b and c) cannot be deduced from the relation of 4 adjacent regions. To solve this, we insert a new "dummy" region r in the center of the non-well-composed configuration. The new region r with new gray value $g(r)$ reflects the relation between a, b, c, d in a way that the topological category of the 8-connectivity LBP code can be deduced from the LBP code of the modified configuration (Fig. 1a, b). In general, the new value for r is computed as follows:
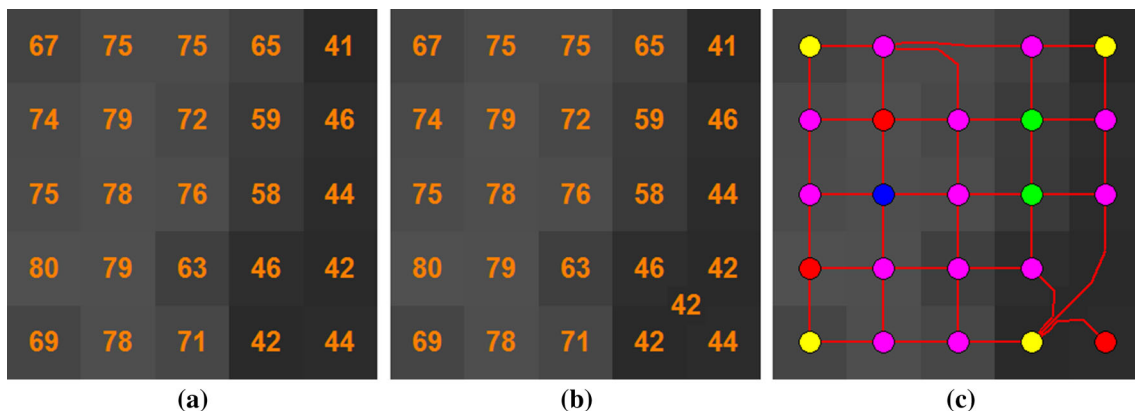

Fig. 1 An 8-bit grayscale image (*highlighted grayscale values*) is **b** made well-composed, and **c** plateaus are merged (for color legend see Fig. 3c)
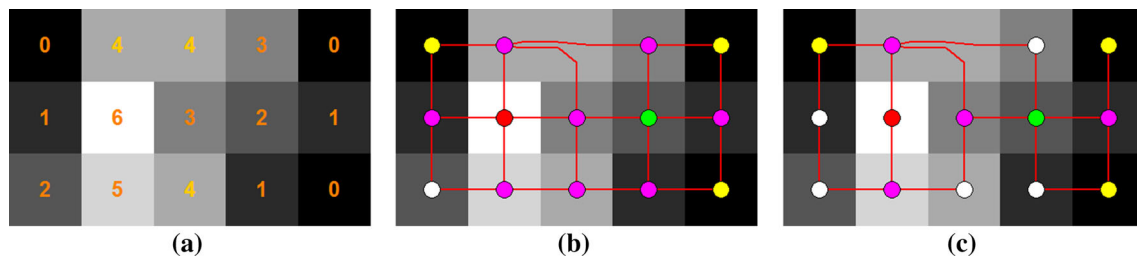
Fig. 2 The plateaus of an image with highlighted values are first merged, and then structurally redundant edges are removed, **a** input, **b** merged plateau, **c** removed edges
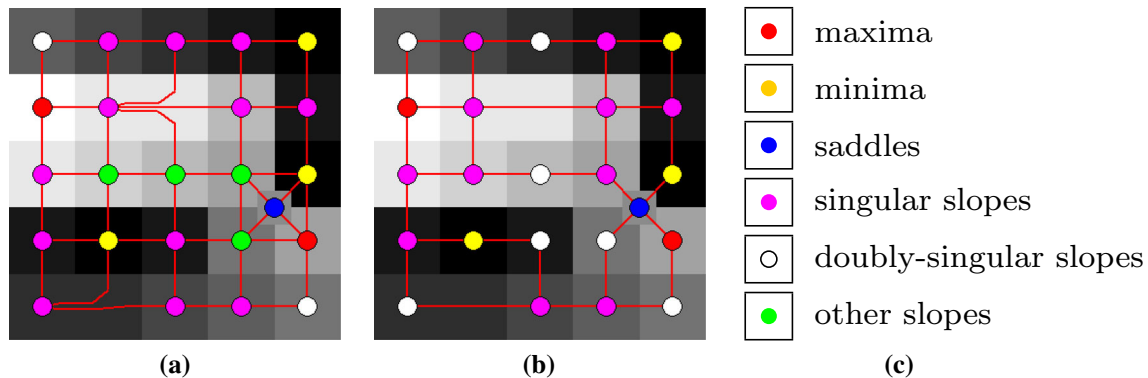

Fig. 3 Removing structurally redundant edges from the primal graph, **a** after merging plateaus, **b** after removing edges, **c** vertex colors

$g(r) = (g(a) + g(b) + g(c) + g(d) - \max\{g(a), g(b), g(c), g(d)\} - \min\{g(a), g(b), g(c), g(d)\})/2$. For some special cases, see [3].

3.2 Merging plateaus and removing edges

The first step in our process to obtain a "minimal" image representation with the same topological information as the original is to merge plateaus. This way, by contracting adjacent vertices with the same gray value in the primal graph (i.e., merging neighboring regions with same value in the dual), we remove the asymmetry caused by the sign function. See Fig. 2b.

After merging all plateaus, a direction can be associated with each edge between different vertices $v \neq u$ of the primal graph G:

$$(u, v) \in E \text{ has direction } u \to v \text{ iff } g(u) > g(v).$$

Proposition 1 [3] *After merging plateaus, $G \setminus \{self\text{-}loops\}$ is a directed graph such that: (a) G does not contain directed cycles. (b) Vertices in $\bar{G}$ do not increase degrees.*

An oriented edge $(u, v) \in E$ is considered *"structurally redundant"* if there exists a dual vertex $\overline{w} \in \overline{V}$ bounded by (u, v) and a directed path $p(u, v)$ from u to v. Structurally redundant edges can be removed in G (see Figs. 2c, 3b). Finally, the notion of well-composed configurations can be extended to regions in $\bar{G}$ due to Prop. 1b.

3.3 Topological category of regions

After merging plateaus, in the same way as for pixels, we can define the *topological category* of a vertex v of G by considering the edges incident to v. See Figs. 1c and 3a. Following the edges incident to v:

- v is a *local minimum* if it is the head of all the edges incident to it.
- v is a *local maximum* if it is the tail of all the edges incident to it.
- v is a *doubly singular slope* if it has degree two and it is the tail of one of the edges incident to it and the head of the other.
- v is a *singular slope* if it has degree greater than two and it is the tail (resp. head) of exactly one edge incident to it and the head (resp. tail) of the others.
- v is a *regular slope* if v is the tail of the first edges incident to it and the head of the others, clockwise or counterclockwise.
- v is a *saddle* otherwise.

Proposition 2 [3] *After removing redundant edges, the topological category of the vertices of G may be simplified: Singular slopes may change to doubly singular slopes, and regular slopes to singular slopes (Figs. 2c, 3b and 4b). Local*
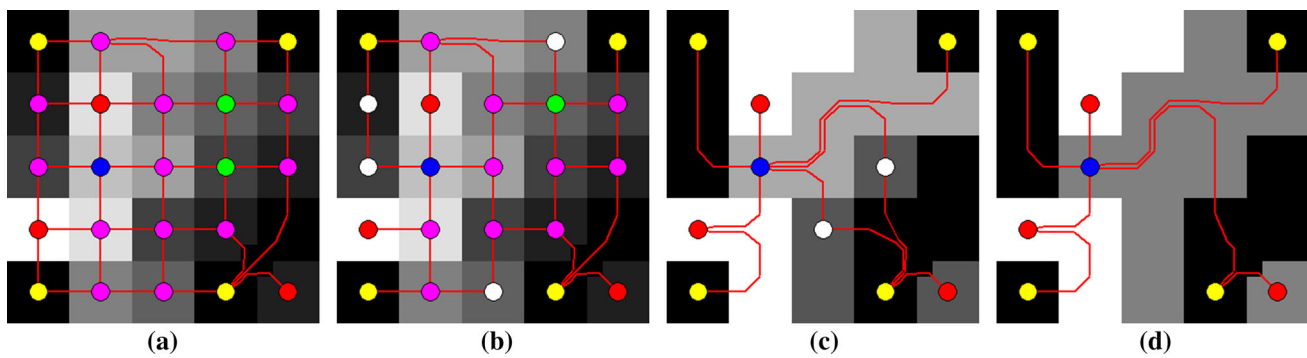
Fig. 4 The primal graph G is successively reduced by merging plateaus and (doubly) singular slopes. Merge operations and removal of structurally redundant edges are applied in an alternating fashion. The last image is considered "minimal," **a** graph plateaus, **b** redundant edges, **c** singular slopes, **d** minimal graph

maxima, minima and saddles are always preserved in the primal graph.

After merging plateaus and removing structural redundant edges, the topological category of the vertices is updated.

3.4 Minimal graph representation

Further steps in our process to obtain a simplified image with same topological information as the original (minimal representative) should remove topologically redundant information by merging regions in the primal graph. In general, a region obtained after merging slopes around a local maximum (resp. minimum) is not a local maximum (resp. minimum) anymore. *Singular slopes* make an exception. Merging singular slopes propagates well around local extrema since a local extremum in the surrounding regions often is or becomes a singular slope. However, non-well-composed configurations (corresponding to saddles in the dual graph) can block propagation. This is why we insert dummy regions in non-well-composed configurations.

Proposition 3 asserts that contracting a singular slope p to a vertex q in the primal graph does not change the topological category of q.

Proposition 3 [3] *Consider a vertex p which is a singular slope in the primal graph. Let E be the set of edges incident to p. Let p be the head (resp. tail) of exactly one edge $e \in E$, and it is the tail of the rest. Let q be the tail (resp. head) of e. Then the vertex obtained after contracting p to q (i.e., after merging the two associated regions in the dual graph) inherits the topological category of the region q, i.e., local max, min, saddle or slope.*

Proposition 4 *The result of merging all singular slopes does not depend on the order we merge.*

Proof Observe that no new singular slope can appear after merging singular slopes, since the topological category of the

rest of the vertices remains invariant. But, new structurally redundant edges can appear. Therefore, the removal of structurally redundant edges and merging singular slopes can be repeated until no more reductions are possible. □

Finally, if regular slopes survive, they may be merged with saddles. On the top of the pyramid local maxima, minima and saddles survive. Regular slopes can remain at the top level of the pyramid. These surviving regular slopes then represent bottlenecks in the image, for example a plateau surrounding several minima and maxima. The number of local maxima, minima and saddles of the original and reduced image coincide (see Fig. 4). Note that after merging a singular slope p, no "corner" adjacent regions with same gray values could become adjacent later since it would mean that p would have at least two 0s in its LBP code. Besides, as a result after contracting plateaus, the reconstructed LBP code is defined for regions thus having as many bits as adjacent regions. Moreover, after merging all plateaus, the LBP codes are symmetric and the operations "image complement" and "LBP code" for regions are commutative. Finally, observe that after merging all singular slopes, each regular slope has at least two 1s and two 0s in its LBP code.

Figure 5 shows the minimal graph representation approach for two grayscale images. After merging plateaus and removing redundant edges, (doubly) singular slopes were merged successively in the primal graph. After each merging, redundant edges were removed. Finally, the surviving regular slopes could be merged with saddles. On the tops of these pyramids only local maxima, minima and saddles survive.

4 Segmentation with LBPs

LBPs have been used in segmentation approaches, for example by Chen et al. [4] and Heikkilä et al. [11]. However, these approaches use LBP histograms, for which the spatial information of LBPs is lost, thus having the drawback that two,
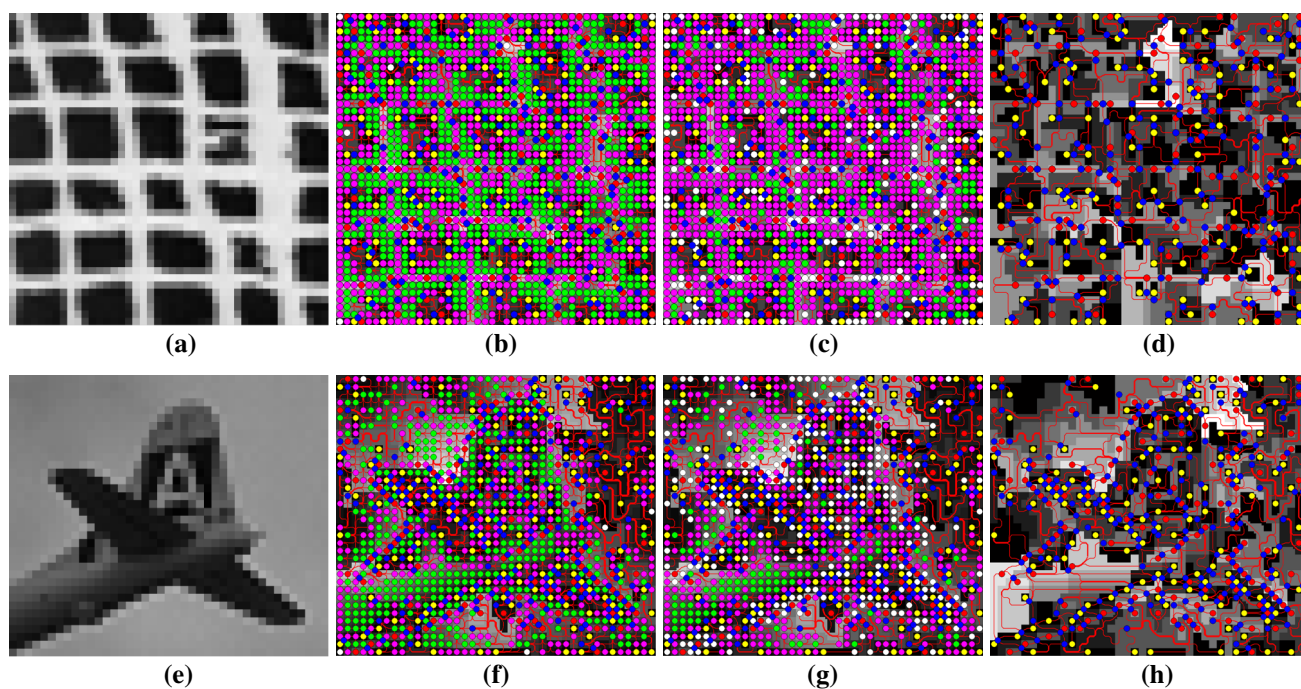
image	maxima	minima	saddles	reg. slopes	sing. slopes	doubly-sing. slopes
(b)	109	103	179	625	693	7
(c)	109	103	179	196	971	158
(d)	109	103	179	0	0	0
(f)	134	131	224	359	361	6
(g)	134	131	224	108	404	214
(h)	134	131	224	0	0	0

Fig. 5 Images are reduced by merging plateaus and then removing structurally redundant edges. After removing all (doubly) singular slopes, regular slopes are merged with saddles, **a** original image1, **b** plateaus, **c** redundant edges, **d** minimal graph, **e** original image2, **f** plateaus, **g** redundant edges, **h** minimal graph

visually completely different, images may have the same histogram.

Modifications made to the so far proposed approach (Sect. 3.4) in form of attributed vertices and contrast-weighted edges allow the definition of texture aware image segmentation algorithms. A first attempt made in this direction was presented in [2]: the "structurally correct image segmentation" (SCIS) algorithm. SCIS is based on LBPs, but does not use histograms. The segmentation builds a combinatorial pyramid (similar as presented in previous sections), while using LBP classes and region contrasts as grouping criteria. This section gives a detailed description of the approach.

Note that the topological categories of regions as defined in Sect. 3.3 refer to vertices in the primal graph. Our segmentation algorithm considers topological categories of the dual graph. In the dual graph, vertices correspond to regions in the primal graph, dual edges encode the boundary of the regions and store the LBP information. In any correctly constructed neighborhood graph (of a well-formed image), it is not possible that local extrema occur in the dual graph. The topological categories are therefore limited to plateaus, slopes and saddles in the dual graph. Dual saddles (i.e., saddles in the dual graph) correspond to non-well-composed configurations as defined in Sect. 3.1 and "are moved" to the primal graph by inserting dummy regions.

4.1 Structurally correct image segmentation

After removing all dual saddles (equals removal of non-well-composed configurations in the primal graph), merging of the primal plateaus (this equals removing dual edges with contrast 0) and merging all dual doubly singular slopes (equals removal of redundant edges in the primal graph) the dual graph (except for the outer face) only contains regular slopes. SCIS is based on this idea (Algorithm 1).

The process of *simplifying the structure* (line 5–8 in Algorithm 1) works as follows. First, notice that only structurally redundant primal edges are determined and removed in the algorithm (corresponding to a contraction of singular slopes in the dual graph). A property of such redundant edges is that neither of the incident vertices is covered by the other.
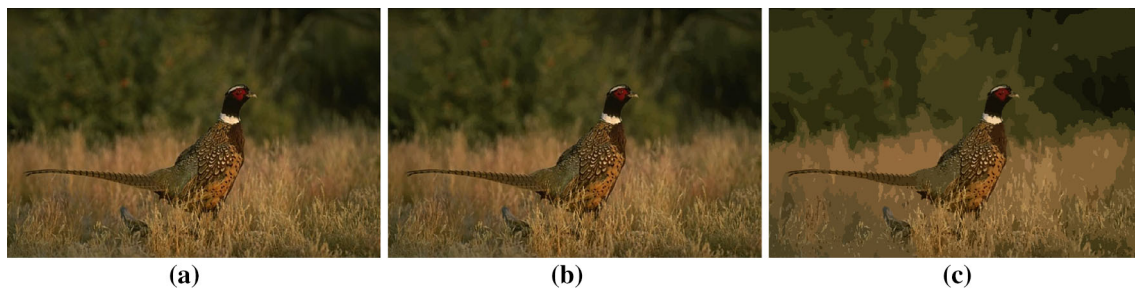
Fig. 6 Original image and reduced version with 70 and 97 % reduction in number of regions, **a** segmentation input, **b** 70 % reduction, **c** 97 % reduction

Therefore, between two vertices u and v, there is always at least one vertex w, such that $g(u) < g(w) < g(v)$ holds. Now, the remaining primal edges are sorted according to their contrast between adjacent regions—note that this contrast is determined in the current SCIS implementation using the CIEDE2000 color distance [28]: In the primal graph, the contrast of an edge (u, v) is given as $|g(u) - g(v)|$. The primal edge with the lowest contrast is checked first if it can be removed: A representative value for the new vertex obtained after merging the primal vertices is computed as the color mean of all included pixels in the region at base level of the pyramid. Because of this re-calculation of the representative values of newly merged regions, the contrast of incident edges needs to be adapted. If the value for the new vertex satisfies the binary relationships stored at the incident dual edges, then the primal edge is contracted. This process is repeated until a suitable primal edge for merging is found. This way, by merging primal edges with the lowest contrast first, also regions with low dissimilarity are merged first.

We denote the graph computed by the SCIS algorithm through removal of redundant edges G_{red}. The SCIS algorithm preserves the structural correctness of the image, since it skips during contraction all edges included in $G_{\text{red}} \backslash G^*$ (where G^* is the minimum equivalent graph [17], which does not contain any redundant edge). Therefore, only edges of G^* are contracted and no cycle is created.

Because of the preservation of structural correctness during the segmentation process, lowly textured regions are merged early in the segmentation hierarchy and highly textured regions are merged late. Figure 6 shows one input image and two reduced images.[1] Note that, even with 97 % reduction in the number of regions, the fine details in the feathers of the bird are still well visible. Therefore, visual information that is important for humans is preserved even at high levels of reduction. This is also well visible in Fig. 7 where the merging history is shown for one image—highly textured regions correspond to bright regions in the merge history as they are merged late in the hierarchy.

[1] More images and segmentation results can be found at: http://prip. tuwien.ac.at/research/scis_results.zip.

Algorithm 1 structurally correct image segmentation (SCIS)

input: 2D image
output: combinatorial pyramid

1: $k := 0$
2: initialize base level C of combinatorial pyramid
3: $C' :=$ remove dual saddles in C
4: $C_0 :=$ merge plateaus in C'
5: **repeat**
6:     $k := k + 1$
7:     simplify structure in current level C_k
8: **until** $C_k = C_{k-1}$

4.2 Internal and external contrast

A segmentation of a digital image is a partition of the pixels into homogeneous regions such that adjacent regions are sufficiently distinct. Observe that a pixel is the smallest region and any connected set of pixels is a region. Criteria for homogeneity can be low contrast and variance, similarity in color or texture. Felzenszwalb [6] uses an "internal contrast" as the largest dissimilarity measurement of such a region, while distinctness is measured by the "external contrast" which is the smallest dissimilarity between adjacent regions. They require that the internal contrast is smaller than the external contrast for the segmentation. In the case of our LBP pyramid, we slightly modify the internal contrast similar to the definition in [10]. We first define low-contrast paths through the neighborhood graph of the image. The contrast of an edge is simply the absolute difference between the values of its two end points. The contrast of a path is the contrast between its endpoints divided by its length. In this way, contrasts of paths of different lengths can be compared.

The contrast of a dual edge is the same as the corresponding edge in the primal graph. The boundary of a simple region (i.e., a region that does not contain any hole) separates the region from its surrounding. It is a path in the dual graph, and its contrast is the smallest contrast of its edges.

We further extend the definition of a low-contrast path to a hierarchical version. In SCIS, a representative value is computed for each region as the color mean of all included
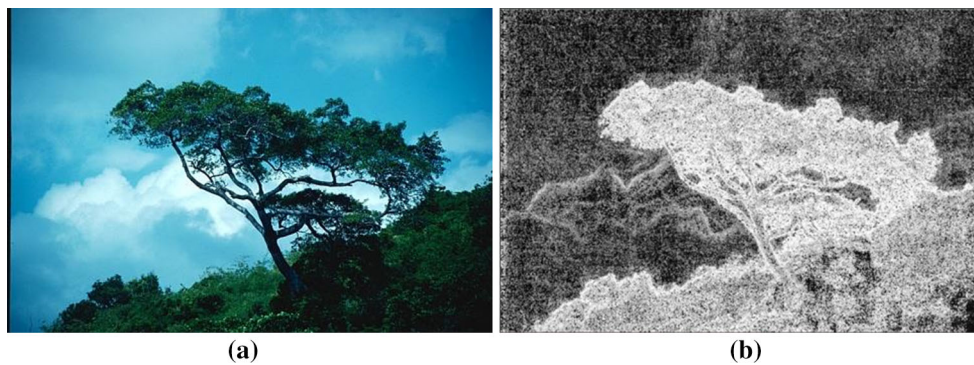
Fig. 7 In the merging history, dark regions were merged early and bright regions were merged late, **a** segmentation input, **b** merge history

pixels at base level of the pyramid. This re-calculation of the representative values of newly merged regions causes that the contrast of incident edges needs to be adapted. This contrast may increase, stay the same, or decrease. A contraction of an edge at a certain level in the pyramid means that this contraction did not create any cycles and that its contrast was the lowest. For this reason, we will consider the hierarchical contrast of a path to consist of edges at base level with the contrast with which they were later merged at higher levels of the pyramid. With these ingredients, we define the internal and external contrasts for our segmentation.

Definition 1 (*Internal contrast*) The internal contrast of a region is the lowest hierarchical contrast of all paths inside the region connecting any pair of its vertices.

Observe that since a region is connected, there is a sequence of neighboring pixels between any pair of pixels of the region. Notice that this definition allows edges with higher contrasts inside the region, they can form branches connected to the boundary, but they cannot form cycles disconnecting the inside of the cycle from the outside. Such a cycle would increase the internal contrast of the region. This definition differs from most other criteria of homogeneity by the fact that it is based on (1D) paths and not on properties of regions.

Definition 2 (*External contrast*) The external contrast of a region is the contrast of the boundary of the region at the current level of the pyramid.

Theorem 1 *All the receptive fields of the SCIS pyramid above plateaus satisfy the condition that the internal contrast of the corresponding region is less than its external contrast. Hence, every level of the pyramid above plateaus forms a segmentation.*

Proof The equivalent contraction kernel of any receptive field (RF) is a tree spanning the receptive field. Any vertex of the receptive field is connected to any other vertex of the RF by edges of the equivalent contraction kernel (ECK).

All edges of the ECK have been contracted before reaching the top of the RF. In SCIS, primal edges are contracted by increasing contrast except in the case when its contraction would reverse the orientation of any adjacent edges. But in this case, the edge is not contracted and cannot be part of the contraction kernel. Edges of a plateau have all zero contrast; hence, they are contracted before any other edges. After all plateaus are reduced to a single vertex, all surviving edges have a higher contrast than the contracted edges. No edge of the boundary has been contracted. They might have been removed by simplification, but this does not disconnect the boundary nor does it decrease the contrast of the boundary. Consequently, the contrast of the boundary of the RF is larger than the internal contrast. Since the above reasoning applies to all levels of the pyramid, any selection of higher level vertices of the RF which partitions the base level forms a segmentation of the underlying image. ☐

5 Experiments

For our image segmentation approach (SCIS) presented in Sect. 4, a quantitative evaluation was done on the 100 test images (in grayscale and color) of the Berkley segmentation database [16] and the segmentation results were compared with the results of five other segmentation algorithms [2]: Greedy, IntExtMST, Eff-Graph, Mean-Shift and Turbopixels. Upon comparison with the ground truth segmentations of the Berkley database, the following observations were made: Around 70 % of the regions in a medium textured image may be contracted, while perceptually observing only a minimal change in the image. A test image from the Berkley database of size $481 \times 321 = 154,401$ pixels may be reduced to around 45,000 regions while keeping the most important information (see also Fig. 6).

Because of the texture preserving property of our segmentation approach, weakly textured regions are undersegmented in favor of preserving highly textured regions. The highly textured regions consist of a high number of very small con-

Fig. 8 Comparative plot of all algorithms for the GCE and the PRI metric for color images for regions in the interval [0, 1000]
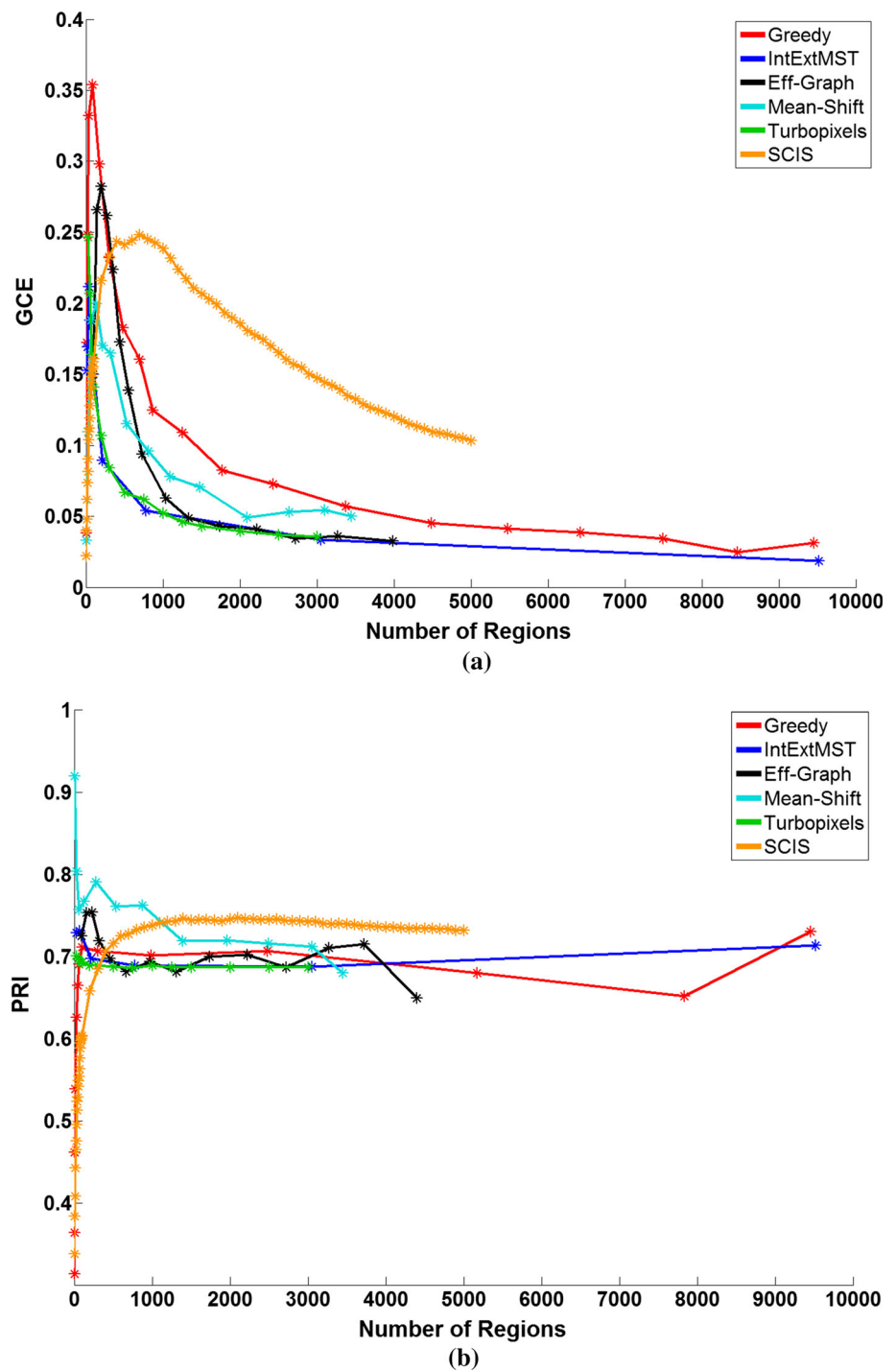
nected components, whereas the weakly textured regions are merged very early and generally consist of a few large connected components. For the comparison with the other segmentation approaches, the global consistency error (GCE) and the probabilistic rank index (PRI) were used as metrics. When comparing the maximum GCE error for color images, the SCIS algorithm provides better results than the Greedy and the Eff-Graph algorithms and is on the same level as

the Turbopixels algorithm. However, in the case of the PRI error metric, the SCIS excels above all algorithm in terms of a steady high PRI score. The PRI curve for the SCIS algorithm drops faster than for the other algorithms, because of the fact that most regions in SCIS are only a couple of pixels large. Consequently, there are a few very large regions that contribute to the bad evaluation. This way the whole curve is shifted a little bit to the right (also for the GCE curve). Com-
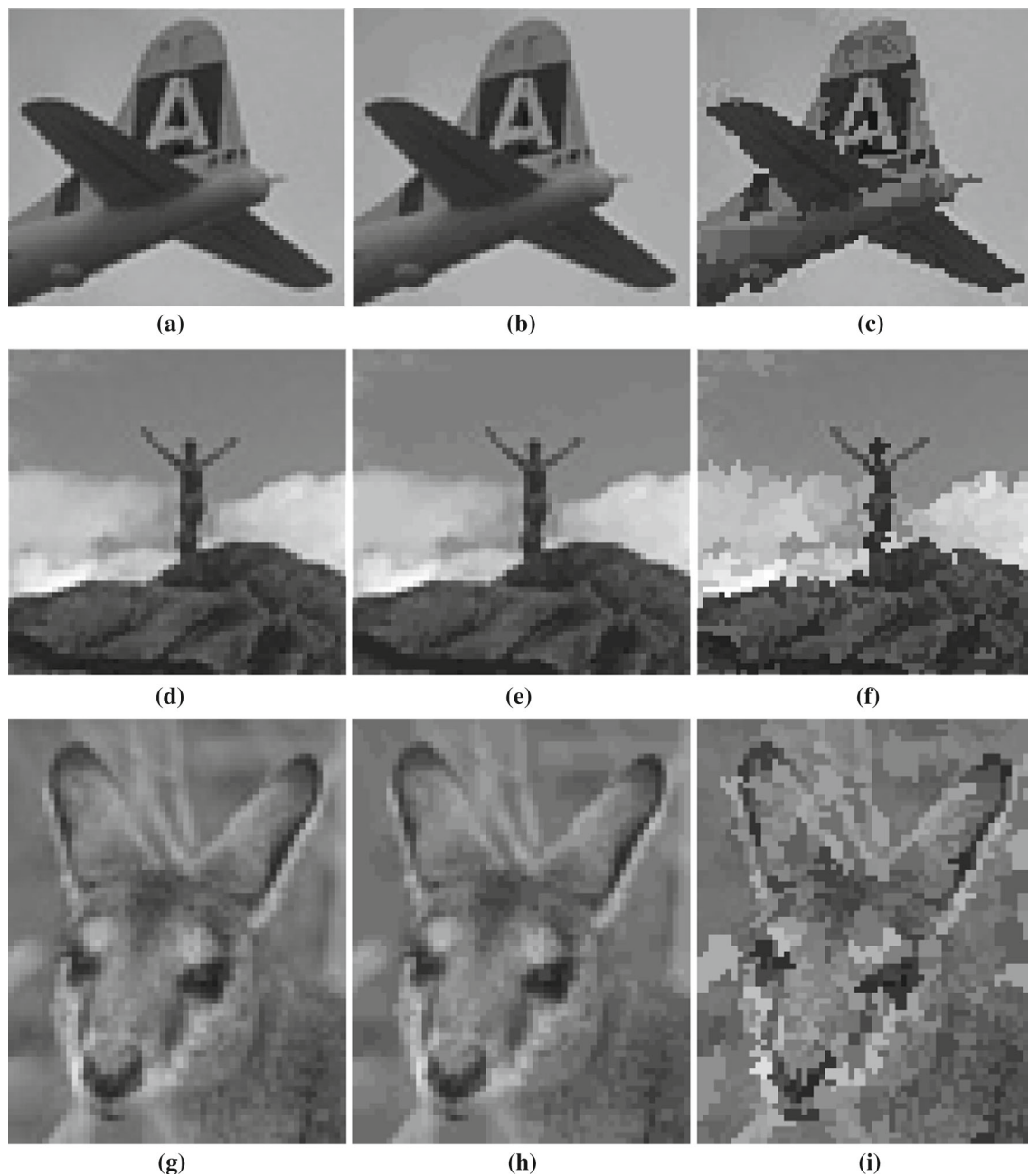
Fig. 9 Segmentation results for the input images, having the same number of regions, obtained using the SCIS algorithm and the minimum representation, **a** airplane (67×72 px), **b** SCIS, **c** minimal graph, **d** statue (53×50 px), **e** SCIS, **f** minimal graph, **g** kangaroo (73×58 px), **h** SCIS, **i** minimal graph

parative plots of all algorithms for the GCE and PRI metric are shown in Fig. 8.

We also compared the results of our segmentation approach with the segmentation obtained by the minimal graph representation. For this comparison, we derive the minimal graph representation of an image as described before (see Fig. 5 for examples). An image segmentation is already given by this minimal graph representation, since each node in the minimal graph representation corresponds to an image region. In

order to compare the segmentations, we assigned the mean color of the pixels of an image region as the color of the region. The minimal graph is given as the apex of the pyramid built when using the topology-based reduction scheme presented in Sect. 3.4. The segmentation approach presented in Sect. 4 proceeds in a similar way, although it primarily considers the contrast of neighboring regions as merging criteria. Therefore, the apex of this segmentation hierarchy is a single node. For the comparison of the two approaches, we
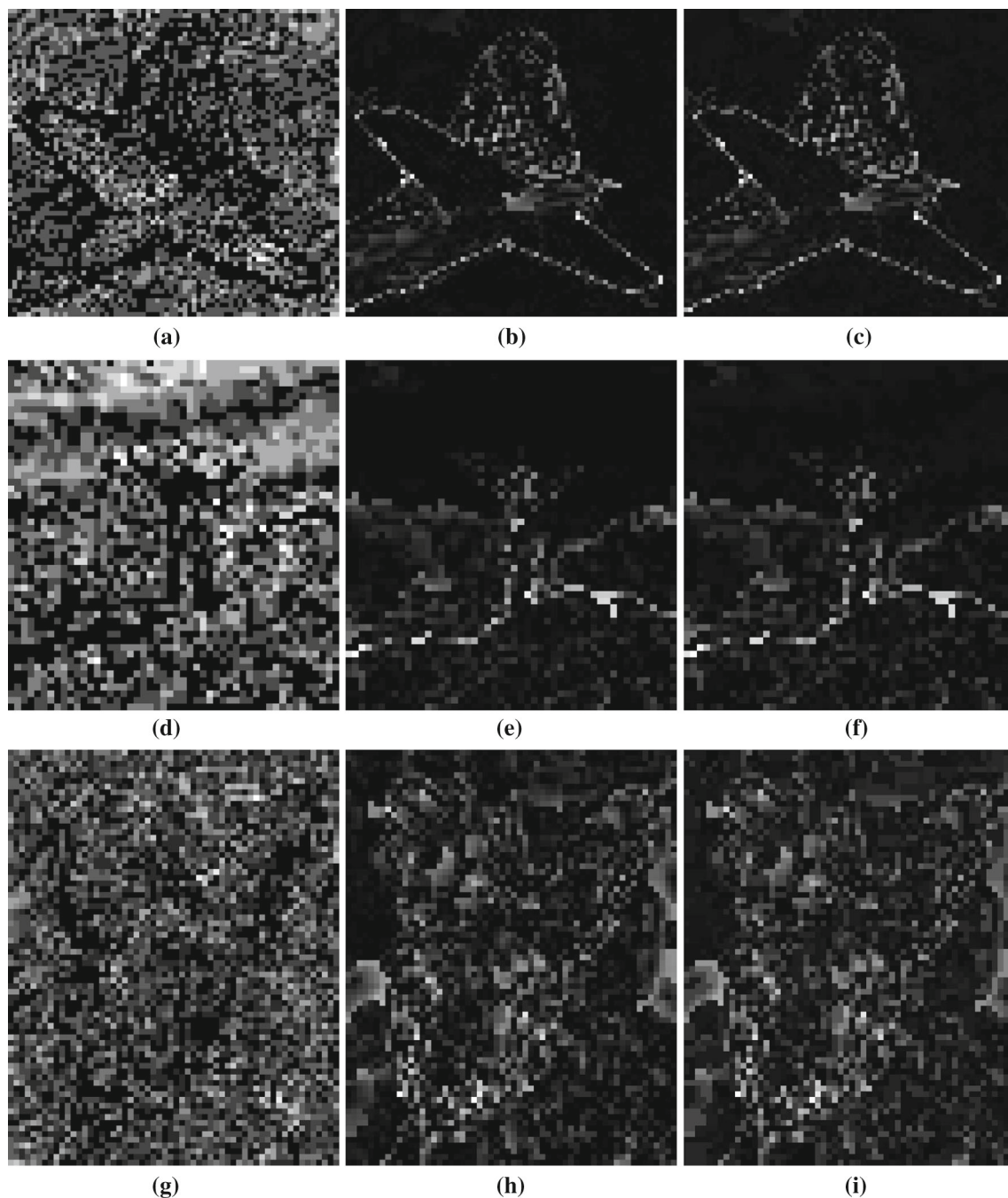
Fig. 10 Pairwise comparison of the pixelwise difference of the images shown in Fig. 9. The differences were stretched to the interval [0, 1] for the visualization, **a** airplane: original—SCIS, **b** airplane: original—min. graph, **c** airplane: SCIS—min. graph, **d** statue: original—SCIS, **e** statue: original—min. graph, **f** statue: SCIS—min. graph, **g** kangaroo: original—SCIS, **h** kangaroo: original—min. graph, **i** kangaroo: SCIS—min. graph

first derived the minimal graph and checked the number of regions $r_{mingraph}$ it preserves. For the segmentation approach, we used the level of the segmentation hierarchy for which the number of regions r_{seg} equals the minimal graph: $r_{mingraph} = r_{seg}$. Region colors are again derived using the mean. Figure 9 shows segmentation results together with the input images.

In Fig. 10, we further computed the pixelwise differences comparing the original image with each of these segmentation results, as well as the two segmented images with each other. Here we can see that the minimal graph segmentation introduces errors predominantly at the edges, whereas the SCIS algorithm introduces errors evenly distributed over

Table 2 Sum of absolute pixelwise differences of the comparisons shown in Fig. 10, based on the three grayscale images (color values in the interval [0, 1]) airplane, statue and kangaroo (Fig. 9)

Image	Original—SCIS	Original—min. graph	SCIS—min. graph
Airplane	13.61	98.63	108.15
Statue	12.68	70.18	78.82
Kangaroo	31.43	170.87	186.95

Table 3 Comparison of the number of individual LBP classes for the two compared segmentation approaches: minimal graph and SCIS

	Maxima	Minima	Saddles	Reg. slopes	(Doubly) sing. slopes
Airplane					
Plateaus merged	378	379	632	2776	1663
Min graph	378	379	632	38	0
SCIS	182	148	183	111	803
Statue					
Plateaus merged	201	198	344	2776	1057
Min graph	201	198	344	19	0
SCIS	88	75	93	34	472
Kangaroo					
Plateaus merged	307	310	563	324	2503
Min graph	307	310	563	27	0
SCIS	111	98	131	88	779

the whole image. However, considering the overall absolute error introduced for one image by SCIS, this error is considerably smaller (see Table 2). Note that this is not visible in Fig. 10 since the differences are stretched to the interval [0, 1] for the visualization. Because of these properties of

the SCIS algorithm, it performs an image segmentation that introduces less artifacts according to human perception, as is well visible in the images in Fig. 9.

However, the number of local minima, maxima and saddles is only preserved by the minimal graph representation, not by the SCIS algorithm (see Table 3). For the SCIS algorithm, a high number of singular and doubly singular slopes remain preserved, while at the same time the number of saddle points decreases. Figure 11 shows a small example for such merged saddle points in SCIS. Here a saddle point is in the middle of two local minima and two local maxima. Since the difference in contrast between the two local minima and the saddle point is the lowest, one of the minima gets merged with the saddle point. This yields a new singular slope, which can be merged again with the remaining local minimum. This way, two neighboring local minima separated by a saddle point (which may, for example, be noise) can get merged into a single local minimum. Such cases are reduced by the SCIS algorithm, while the minimal graph preserves all of the three vertices.

6 Conclusion and future work

We discuss a codification of local binary pattern using graph pyramids. For well-composed images, we demonstrate that from the 4 neighbors' topological information, we can obtain the 8 neighbors' topological information as well. This renders the classical 8-neighbor LBP topologically redundant. By inserting a few dummy regions, every image can be made well-composed. A sequence of merging regions and removal of edges obtains a "minimal" image I derived from the minimal graph representation, with the same topological information as the original. We further adapt this hierarchical representation and present a texture aware image segmentation approach (SCIS) that preserves structure in images
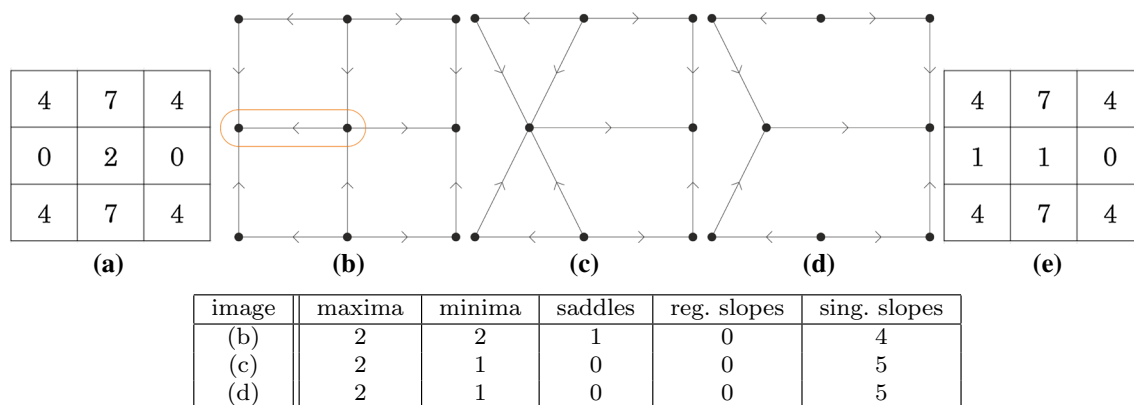

image	maxima	minima	saddles	reg. slopes	sing. slopes
(b)	2	2	1	0	4
(c)	2	1	0	0	5
(d)	2	1	0	0	5

Fig. 11 Example for decreasing number of saddle points and increasing number of slopes in SCIS, **a** input image, **b** input, **c** merge, **d** redundant edges, **e** SCIS

(information that is important for human perception) up to high levels of the segmentation pyramid.

Toward 3D. Several researchers have tried to extend the LBP codification from 2D plane to 3D volume (see, for example, [5,23]); however, it is not so straightforward as it appears at first glance. In our case, the notion of well-composed images and irregular graph pyramids also works for 3D images. Characterizations of pixels in 3D using LBP codes depend not only on the number of connected components but also on the number of holes (1-dimensional homology classes).

Shape LBP. Textures have particular shapes. The idea is to label the edges as 0 or 1 in the dual graph depending on concavity–convexity of the boundary of a region. Computing LBP codification using the concavity–convexity rule could help to recognize shapes.

Image compression. Due to its texture preserving properties the SCIS algorithm may be used for image compression. It allows a high reduction in region numbers, while keeping the perceptively important information of an image, encoded as textured regions, intact. There is, however, a distinction to be made between reduction in region numbers and an overall reduction in terms of storage costs. Currently, the whole merging history of the combinatorial pyramid can be stored at the storage costs of the base combinatorial map. It is possible to store only higher levels of the pyramid—a reduced combinatorial pyramid at lower storage costs. Even though this reduced combinatorial pyramid uniquely stores the structure of the image in terms of adjacency and inclusion relationships of regions, it is still necessary to store the mapping of regions onto the image grid. One open problems is to incorporate a method to also store this mapping efficiently in terms of storage costs.

References

1. Arbelaez, P., Maire, M., Fowlkes, C., Malik, J.: Contour detection and hierarchical image segmentation. IEEE TPAMI **33**(5), 898–916 (2011)
2. Cerman, M.: Structurally correct image segmentation using local binary patterns and the combinatorial pyramid. Technical Report 133(ftp://ftp.prip.tuwien.ac.at/pub/publications/trs/tr133), Vienna University of Technology, Pattern Recognition and Image Processing (PRIP) Group (2015)
3. Cerman, M., Gonzalez-Diaz, R., Kroptasch, W.: LBP and irregular graph pyramids. In: 16th International Conference on Computer Analysis of Images and Patterns (CAIP2015) (2015)
4. Chen, J., Shan, S., He, C., Zhao, G., Pietikäinen, M., Chen, X., Gao, W.: WLD: a robust local image descriptor. IEEE TPAMI **32**(9), 1705–1720 (2010)
5. Fehr, J., Burkhardt, H.: 3D rotation invariant local binary patterns. In: Proceedings of 19th International Conference on Pattern Recognition (ICPR08), pp. 1–4 (2008)
6. Felzenszwalb, P.F., Huttenlocher, D.P.: Image segmentation using local variation. In: Proceedings of Computer Vision and Pattern Recognition, pp. 98–104 (1998)
7. Gonzalez-Diaz, R., Ion, A., Iglesias-Ham, M., Kropatsch, W.G.: Invariant representative cocycles of cohomology generators using irregular graph pyramids. Comput. Vis. Image Underst. **115**(7), 1011–1022 (2011)
8. Guo, G., Jones, M.J.: Iris extraction based on intensity gradient and texture difference. In: Proceedings of the 2008 IEEE Workshop on Applications of Computer Vision (WACV '08), pp. 1–6 (2008)
9. Hadid, A., Pietikinen, M., Ahonen, T.: A discriminative feature space for detecting and recognizing faces. In: 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'04), pp. 797–804 (2004)
10. Haxhimusa, Y.: The Structurally Optimal Dual Graph Pyramid and its Application in Image Partitioning, 1st edn. IOS Press, Amsterdam (2007)
11. Heikkilä, M., Pietikäinen, M.: A texture-based method for modeling the background and detecting moving objects. IEEE TPAMI **28**(4), 657–662 (2006)
12. Kropatsch, W.G.: Building irregular pyramids by dual-graph contraction. IEEE Proc. Vis. Image Signal Process. **142**(6), 366–374 (1995)
13. Kropatsch, W.G., Haxhimusa, Y., Pizlo, Z., Langs, G.: Vision pyramids that do not grow too high. Pattern Recognit. Lett. **26**(3), 319–337 (2005)
14. Latecki, L., Eckhardt, U., Rosenfeld, A.: Well-composed sets. Comput. Vis. Image Underst. **61**, 70–83 (1995)
15. LBP'2014 Workshop on Computer Vision With Local Binary Pattern Variants https://sites.google.com/site/lbp2014ws/
16. Martin, D., Fowlkes, C., Tal, D., Malik, J.: A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In: 8th IEEE International Conference on Computer Vision (ICCV), vol. 2, pp. 416–423 (2001)
17. Moyles, D.M., Thompson, G.L.: An algorithm for finding a minimum equivalent graph of a digraph. J. ACM **16**(3), 455–460 (1969)
18. Nguyen, D.T., Ogunbona, P.O., Li, W.: A novel shape-based nonredundant local binary pattern descriptor for object detection. Pattern Recognit. **46**(5), 1485–1500 (2013)
19. Ojala, T., Pietikainen, M., Harwood, D.: A comparative study of texture measures with classification based on featured distributions. Pattern Recognit. **29**(1), 51–59 (1996)
20. Ojala, T., Pietikinen, M.: Unsupervised texture segmentation using feature distributions. Pattern Recognit. **32**, 477–486 (1999)
21. Ojala, T., Pietikainen, M., Maenaa, T.: Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. IEEE TPAMI **24**(7), 971–987 (2002)
22. Paris, S., Durand, F.: A topological approach to hierarchical segmentation using mean shift. CVPR 2007, IEEE Computer Society 2007, ISBN 1-4244-1179-3
23. Paulhac, L., Makris, P., Ramel, J.-Y.: Comparison between 2D and 3D local binary pattern methods for characterization of three-dimensional textures. In: Proceedings of the International Conference on Image Analysis and Recognition, LNCS, vol. 5112, pp. 670–679 (2008)
24. Peng, B., Zhang, L., Zhang, D.: A survey of graph theoretical approaches to image segmentation. Pattern Recognit. **46**(3), 1020–1038 (2013)
25. Pietikinen, M., Hadid, A., Zhao, G., Ahonen, T.: Computer Vision Using Local Binary Patterns, vol. 40. Springer, New York (2011)
26. Raut, S., Raghuvanshi, M., Dharaskar, R., Raut, A.: Image segmentation—a state-of-art survey for prediction. In: Proceedings of the 2009 International Conference on Advanced Computer Control (ICACC), pp. 420–424 (2009)

27. Shan, C.: Learning local binary patterns for gender classification on real-world face images. Pattern Recognit. Lett. **33**(4), 431–437 (2012)
28. Sharma, G., Wu, W., Dalal, E.N.: The CIEDE2000 color difference formula: implementation notes, supplementary test data, and mathematical observations. Color Res. Appl. **30**(1), 21–30 (2005)

Martin Cerman received the BSc and MSc degrees in Visual Computing from TU Wien in 2012 and 2015, respectively. He is currently employed at Anyline GmbH as a Computer Vision Researcher developing offline OCR solutions for mobile phones and smart glasses. His current research interests include image processing, statistical pattern recognition, machine learning, as well as graph theory and topology, and their implementations in real-time applications.

Ines Janusch is currently a PhD candidate and a University Assistant at the Pattern Recognition and Image Processing group of TU Wien. She received the BSc in Computer Science in 2009 and the MSc in Visual Computing in 2014 both from TU Wien. Her current research interests are in applications of graph theory, topology and persistence to image processing and pattern recognition as well as in shape representations.

Rocio Gonzalez-Diaz is an Associate Professor in the Department of Applied Math (I) at the University of Seville. She is the head of the Computational Image Analysis group since 2010 and Principal Researcher of the Spanish projects on Computational Topology for Combinatorial Image Analysis MTM2012-32706 and Computer Vision MTM2015-67072-P, and vice chair of the technical committee IAPR-TC18: Discrete Geometry. Her main interest is to investigate tools (methods) for Combinatorial Image Analysis and Computer Vision that build on the theory of (persistent-) homology and cohomology.

Walter G. Kropatsch Since 1990, Walter G. Kropatsch is full professor at TU Wien and head of the Pattern Recognition and Image Processing group. His scientific research focused on pyramidal representations of images since his collaboration with Azriel Rosenfeld in 1984/85. The current graph-based pyramids follow similar concepts with the advantage that graphs are a much more flexible data structures than the regular grids as used in deep learning. In his more than 370 scientific contributions, many other concepts and applications have been addressed. He is currently area editor of the journal Computer Vision and Image Understanding and associate editor of the journal of Electronic Imaging, the journal of the Visual Computer and of several special issues in Pattern Recognition, Pattern Recognition Letters and the journal of Mathematical Imaging and Vision.