Pyramids of n-Dimensional Generalized Maps

Carine Grasset-Simon, Guillaume Damiand, Pascal Lienhardt

HAL Id: hal-01511770
https://hal.science/hal-01511770

Pyramids of n-dimensional generalized maps

Carine Grasset-Simon, Guillaume Damiand, and Pascal Lienhardt

SIC, FRE-CNRS 2731 - Université de Poitiers
bât. SP2MI, Bvd M. et P. Curie
BP 30179, 86962 Futuroscope Chasseneuil Cedex - France
{simon,damiand,lienhardt}@sic.univ-poitiers.fr

Abstract. Graph pyramids are often used for representing irregular pyramids. Combinatorial pyramids have been recently defined for this purpose. We define here pyramids of n-dimensional generalized maps. This is the main contribution of this work: a generic definition in any dimension which extend and generalize the previous works. Moreover, such pyramids explicitly represent more topological information than graph pyramids. A pyramid can be implemented in several ways, and three representations are discussed in this paper.

Keywords. Hierarchical data structure, irregular pyramids, generalized map, n-G-map pyramids, multi-resolution.

1   Introduction

Hierarchical structures are often used for representing a same object with different resolutions. The first level of such a structure generally represents the object with a very fine precision, then the precision decreases as the level increases. For image processing, pyramids are used for representing different segmentation levels of an image. It is thus possible to get immediately the segmentation level suited for a particular process. It is also possible to modify a pyramid level, for instance when changing the segmentation criteria.

Irregular image pyramids have been studied by many authors, and used for several applications ([1,2,3,4,5,6,7,8]). An irregular pyramid is defined as a stack of reduce graphs where each graph is built from the precedent level by a sampling or decimation process. Brun and Kropatsch have extended this notion by defining pyramids in which each level is a 2-dimensional combinatorial map ([9,10,11]). Combinatorial maps represent the topology of any subdivisions of any orientable surfaces without boundary.

We extend these works for any dimension by defining pyramids of n-dimensional generalized maps (or n-G-maps). This is the main result of this paper. Generalized maps represent the topology of any subdivisions of any orientable or not orientable n-dimensional manifolds with or without boundary (see [12,13]). So, pyramids of n-G-maps can be used in order to process images in 2D, 3D, as well as 4D (for instance for tracking objects in 3D image sequences). An other interest of generalized maps is that their definition is homogeneous for any dimensions. This facilitates the definition of generic operations and algorithms.

Our work follows that of Damiand and Lienhardt who have defined for n-G-maps a general operation for removing and contracting cells of any dimensions [14]. More precisely, each level of a pyramid is a simplification of the previous one, obtained by applying this operation.

We recall in section 2 the notion of generalized map as well as the operation of cell contraction and removal. Generalized map pyramids are defined in section 3. We study in section 4 three different representations of generalized map pyramids. We discuss in section 5 the construction of a pyramid in the context of image segmentation. Generalized map pyramids and graph pyramids are compared in section 6. Further issues are discussed in section 7.

2 Recalls: n-G-map, cell removal and contraction

n-dimensional generalized maps (or n-G-maps) represent the topology of n-dimensional subdivided objects, and more precisely the topology of quasi-manifolds (see [12,15,13]). An n-G-map is a set of abstract elements (darts), together with applications defined on these darts. Cells are implicitly represented as sets of darts. (see [13] for more details).

Definition 1 (n-G-map and i-cell (cf. figure 1)). *Let $n \geq 0$.*
An n-dimensional generalized map G is defined by $G = (D, \alpha_0, \ldots, \alpha_n)$ where:

1. *D is a finite set of darts;*
2. *$\forall k,\ 0 \leq k \leq n,\ \alpha_k$ is an* involution[1] *on D;*
3. *$\forall k, j,\ 0 \leq k < k + 2 \leq j \leq n,\ \alpha_k \alpha_j$ is an involution.*

Let $d \in D$, $N = \{0, 1, \ldots, n\}$ and let i be such that $0 \leq i \leq n$.
The i-cell incident to d is the orbit [2]
$$<>_{N-\{i\}} (d) = < \alpha_0, \ldots, \alpha_{i-1}, \alpha_{i+1}, \ldots, \alpha_n > (d).$$

In order to define n-G-map pyramids, Damiand and Lienhardt have proposed an operation for simultaneously removing and contracting cells of any dimensions [14](cf. figure 2). These cells have to satisfy two preconditions: they are disjoint two by two[3], and their degree is two.

For $i \in N$, R_i is the set of removed i-cells, and C_i is the set of contracted i-cells. Note that $R_n = \emptyset$ and $C_0 = \emptyset$ since it is not possible to remove n-cells nor to contract 0-cells (see [14] for more details).

---

[1] An involution f on a finite set S is a one to one mapping from S onto S such that $f = f^{-1}$.

[2] Let $\{\Pi_0, \ldots, \Pi_n\}$ be a set of permutations on D. The orbit of an element d related to this set of permutations is $< \Pi_0, \ldots, \Pi_n > (d) = \{\Phi(d), \Phi \in < \Pi_0, \ldots, \Pi_n >\}$, where $< \Pi_0, \ldots, \Pi_n >$ denotes the group of permutations generated by $\Pi_0, \ldots, \Pi_n$.

[3] The set of i-cells is a partition of the set of darts of the n-G-map, for each i between 0 and n. Two cells are disjoint if their intersection is empty, i.e. when no dart is shared by the cells.
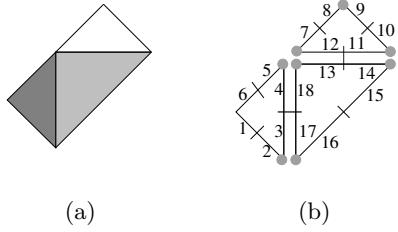
(a)　　　　　　　　　(b)

Fig. 1. (a) A subdivision of a surface. (b) The corresponding 2-G-map. Darts are represented by numbered black segments. Two darts in relation by α_0 share a little vertical segment (ex. darts 1 and 2). Two darts in relation by α_1 share a same point (ex. darts 2 and 3). Two distinct darts in relation by α_2 are parallel and close to each other (ex. darts 3 and 17); otherwise, the dart is its own image by α_2 (ex. dart 2). The vertex incident to dart 2 is $< \alpha_1, \alpha_2 > (2) = \{2, 3, 16, 17\}$, the edge incident to dart 3 is $< \alpha_0, \alpha_2 > (3) = \{3, 4, 17, 18\}$, and the face incident to dart 9 is $< \alpha_0, \alpha_1 > (9) = \{7, 8, 9, 10, 11, 12\}$.
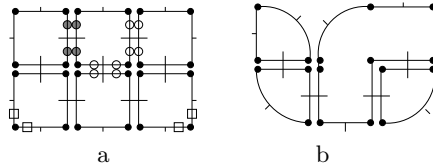

a　　　　　　　　b

Fig. 2. An example of simultaneous removal and contraction of cells of different dimensions. (a) A 2-G-map where the darts of removed 0-cells, removed 1-cells and contracted 1-cells are respectively marked by empty squares, circles and gray disks. (b) The resulting 2-G-map.

3　Definition of n-G-map pyramids

An n-G-map pyramid is a hierarchical structure. Each level is an n-G-map: the first level represents the initial data; the other levels represent successive simplifications. A possible bottom-up construction is the following one. Level 0 is an n-G-map which represents the initial data. According to the application, a process will define two sets for any dimension i ($i \in N$): R_i^0 (resp. C_i^0) is the set of i-cells which will be removed (resp. contracted). The cells of these sets have to satisfy the preconditions of the removal and contraction operation. The application of this operation gives the level 1 map. This process is reapplied in order to get the other levels of the pyramid.

The formal definition of an n-G-map pyramid is the following:

Definition 2 (n-G-map pyramid). *Let n, $m \geq 0$. A $m + 1$ level pyramid $\mathcal{P}$ of n-dimensional generalized maps is the set $\mathcal{P} = \{G^k\}_{0 \leq k \leq m}$ where:*

1. $\forall k$, $0 \leq k \leq m$, G^k is the n-G-map $(D^k, \alpha_0^k, \ldots, \alpha_n^k)$,

2. *For each k, $0 \leq k \leq m$, for each i, $0 \leq i < n$, let R_i^k (resp. C_i^k) be sets of i-cells such that: cells are disjoint two by two and the degree of each cell is equal to 2, i.e.:*
   - *$\forall C$, $C' \in \bigcup_{i=0}^{n}(R_i^k \cup C_i^k)$, $C \cap C' = \emptyset$,*
   - *$\forall i$, $0 \leq i \leq n-2$, $\forall d \in R_i^k$, $d\alpha_{i+1}^k\alpha_{i+2}^k = d\alpha_{i+2}^k\alpha_{i+1}^k$,*
   - *$\forall i$, $2 \leq i \leq n$, $\forall d \in C_i^k$, $d\alpha_{i-1}^k\alpha_{i-2}^k = d\alpha_{i-2}^k\alpha_{i-1}^k$,*
3. *$\forall k$, $0 < k \leq m$, G^k is obtained from G^{k-1} by removing the cells of $\cup_{i=0}^{n}R_i^{k-1}$ and contracting the cells of $\cup_{i=0}^{n}C_i^{k-1}$.*

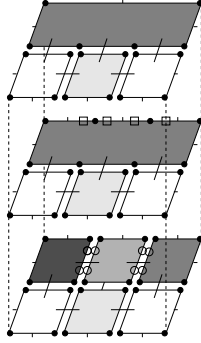Examples of 2D and 3D pyramids are provided in figures 3 and 4.

Fig. 3. A 2-G-map pyramid composed of three levels. The darts of removed 0-cells (resp. 1-cells) are marked by empty squares (resp. circles).
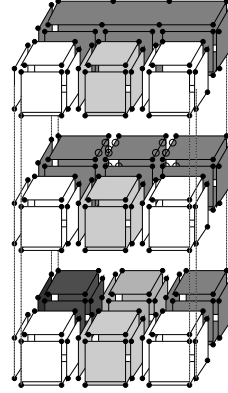

Fig. 4. A 3-G-map pyramid composed of three levels. The second level is obtained by removing 2 faces, and the third level is obtained by removing 4 edges (the two upper edges and the two edges in the background which are adjacent to the first ones).

Two major properties of n-G-map pyramids are:

- each dart which belongs to a removed or a contracted cell of level k does not belong to another removed or contracted cell in the same level or in another level.
- Let k, $0 \leq k < m$. Note that a one to one mapping φ^k exists between the surviving darts of G^k (i.e. the darts which are not removed nor contracted), and the darts of G^{k+1} ($\varphi^k : D^k - \bigcup_{i=0}^{n}(R_i^k \cup C_i^k) \longrightarrow D^{k+1}$). In order to simplify, we will denote a dart of G^k and its image in G^{k+1} by the same name. So, we have: $D^{k+1} = D^k - \bigcup_{i=0}^{n}(R_i^k \cup C_i^k)$.

More formally:

Proposition 1.

1. $\forall i, j \in N, \forall k, l \in [0..m-1]$ we have: $\begin{cases} R_i^k \cap C_j^l = \emptyset, \\ R_i^k \cap R_j^l = \emptyset, \ \ with \ i \neq j \ or \ k \neq l, \\ C_i^k \cap C_j^l = \emptyset, \ \ with \ i \neq j \ or \ k \neq l. \end{cases}$

2. $\forall k, \ 0 \leq k < m, \ D^{k+1} \subseteq D^k$.

These properties can be easily deduced from the definition of the removing and contracting operation [14], and from the definition of n-G-map pyramids. Moreover these properties are useful for the definition of representations of n-G-map pyramids, and more precisely for the *implicit* and *hierarchical* representations.

4 Different representations of n-G-map pyramids

An n-G-map pyramid can be represented more or less explicitly according to the performance expected in space and in time. We present in this section three possible representations: *explicit*, *hierarchical* and *implicit* (see figure 5). Note that each representation has to satisfy the constraints of the definition of n-G-map pyramids, which can be easily expressed for each representation. Note also that these three representations contain the same information, but each of them has different advantages and drawbacks (explain in the following).

Explicit n-G-map pyramid: each level k (i.e. G^k) and each bijection φ^k are explicitly represented (see figure 5-a). The representation contains thus $m + 1$ n-G-maps, and each dart is linked with its predecessor (except the darts of level 0) and with its successor[4] (except the darts of the last level and the darts which belong to a removed or contracted cell). Moreover, for each level, a dart which belongs to a removed or contracted cell in this level is marked by the type of the operation (contraction or removal) and the dimension of the cell.

Hierarchical n-G-map pyramid: $D^k - \bigcup_{i=0}^{n}(R_i^k \cup C_i^k)$ is identified with D^{k+1} for each k ($0 \leq k < m$) but each involution α_i is explicitly represented ($\forall i$, $0 \leq i \leq n$). This representation (see figure 5-b) contains a single set of darts, and for each dart one table which represents involutions α for all levels (the size of this table is the product of the number of involutions by the number of levels for each dart). If a dart disappears at level k, its images by involutions α are only defined from level 0 to level k. A possible optimization is the following: when $d\alpha_i^k = d\alpha_i^{k+1}$, the image of d is represented only once. More precisely, given i ($0 \leq i \leq n$) and a dart d, we only memorize distinct images of d by α_i, and for each dart, the last level at which it belongs.

---

[4] The predecessor relation corresponds to $(\varphi^{k-1})^{-1}$ and the successor relation φ^k.

Implicit n-G-map pyramid: this representation contains only the first level (i.e. G^0) and three marks are associated with each dart. The first one corresponds to the type of operation (removal or contraction) which suppresses the dart (if the case arises). The second one corresponds to the dimension of the removed or contracted incident cell. The third one indicates the level at which the cell disappears.
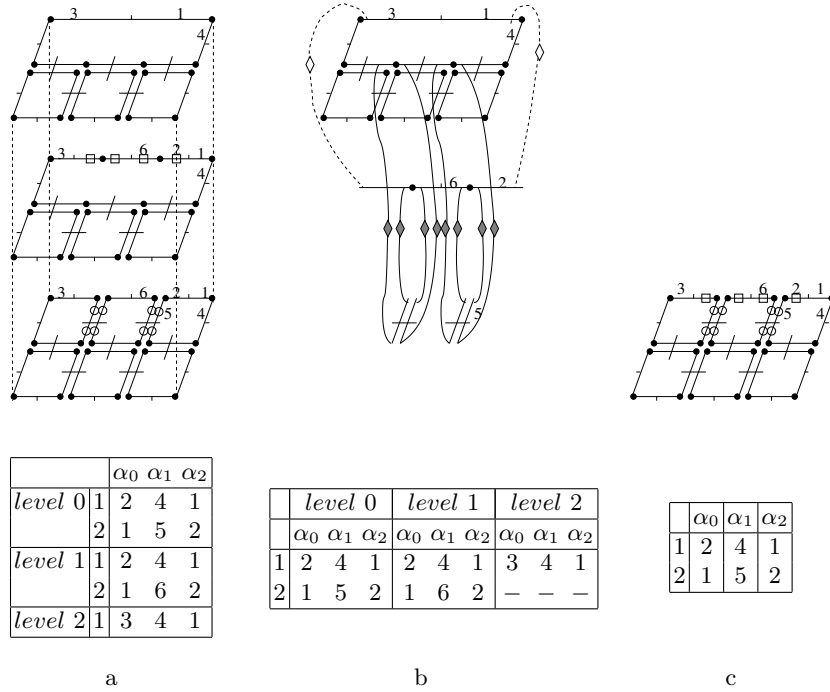

		α_0	α_1	α_2
level 0	1	2	4	1
	2	1	5	2
level 1	1	2	4	1
	2	1	6	2
level 2	1	3	4	1

	level 0			*level 1*			*level 2*		
	α_0	α_1	α_2	α_0	α_1	α_2	α_0	α_1	α_2
1	2	4	1	2	4	1	3	4	1
2	1	5	2	1	6	2	−	−	−

	α_0	α_1	α_2
1	2	4	1
2	1	5	2

a   b   c

Fig. 5. Three representations of a same pyramid. For each representation, the corresponding array shows images by involutions α for the two darts 1 and 2. (a) *Explicit*. Removed 0-cells (resp. removed 1-cells) are marked by empty squares (resp. circles). (b) *Hierarchical*. Each dart is drawn in the last pyramid level where it exists. When two darts linked by α_i are drawn in the same level, their link α_i is drawn in the usual way. Otherwise, the links α_0^1 and α_0^0 between two darts of two different levels are represented by lines with empty lozenge and the link α_1^0 by line with filled lozenge. (c) *Implicit*. Removed 0-cells of level 0 (resp. removed 1-cells of level 1) are marked by empty squares (resp. circles).

The choice of one representation depends on the particular needs of the application, since these representations offer different advantages and drawbacks.

The *explicit* representation is characterized by an important redundancy of information, and thus an heavy cost in memory space. On the other hand, each pyramid level can be directly accessed. So, the extraction of a particular level or the modification of the pyramid can be easily achieved.

The *hierarchical* representation is less costly in memory space, since there are less redundant informations. Any level can be directly extracted by using involutions of this level. Moreover, given a cell, one can directly access to the set of cells of a lower level which are "merged" into this cell. This can be useful for example for the representation of different levels of details (see [16] where a data structure, based upon a similar principle, is proposed in order to model complex architectural environments). On the other hand, it is usually difficult to modify the pyramid, since all levels are not explicitly represented. More precisely, it is difficult to propagate modifications between the pyramid levels.

The *implicit* representation has an optimal complexity in memory space since there is no redundant information. Moreover, the pyramid can be directly modified, avoiding the problem related to the information propagation. A similar representation is proposed by Brun and Kropatsch [17] for representing a 2D combinatorial pyramid. The main drawback of this representation is that a pyramid level can not be directly accessed: it is necessary to compute it when required.

5 Construction of a pyramid for image segmentation

For image processing, pyramids are used in particular in order to keep in memory different segmentations of a same image. In order to construct an n-G-map pyramid associated with a multi-level segmented image, level 0 is associated with the initial image. A new level is constructed through two steps: first, the cells of the "previous segmentation" corresponding to homogeneous regions are "merged" into one cell; second, the resulting n-G-map is simplified.

An example is the following. We want to segment a 3D gray level image by using a simple gray level distance as homogeneity criterion. The initialization consists in associating a 3-G-map with the initial image (cubic volumes are associated with voxels). A label representing the gray level of the corresponding voxel is associated with each volume. The voxels corresponding to a homogeneous region are merged. This is achieved by removing the faces which are between the corresponding volumes. More precisely, a process based upon the homogeneity criterion marks level 0 faces, then the 3-G-map is duplicated and the removal and contraction operation is applied, producing level 1 map. The same principle is applied in order to compute the following levels. It is possible to modify the marking process in order to control the topology of the 3-G-maps, for instance for avoiding disconnections. In order to reduce the required memory space, the representation of the boundary between two regions can be simplified, by merging the boundary faces into one face. This simplification can be achieved by removing the degree 2 edges and then the degree 2 vertices. Other simplifications can be made in order to obtain a minimal representation (see [18,19]). A new level is

constructed by the same two phases: first marking, duplication and removal of faces; then simplifications by removing degree 2 edges and vertices.

For 2D images, pixels which make a homogeneous region (i.e. faces of the 2-G-map) are merged by removing the edges between these faces. The 2-G-map is then simplified by removing the degree 2 vertices. For 4D images, 4D cells are merged by removing the volumes which lie between them. The boundary between two regions can be simplified by removing the degree 2 faces, then the degree 2 edges and then the degree 2 vertices. More simplifications can be done depending on the particular needs for each application.

6 Comparison with graph pyramids

Irregular image pyramids are often represented by adjacency graph pyramids. A vertex of such an adjacency graph corresponds to a region of the image and an edge symbolizes the adjacency relation between the two regions associated with the extremity vertices. Figure 6 shows different representations of a segmented image. Note that one region is included into an other one, and that two regions are adjacent several times to each other. An adjacency graph represents all types of adjacency in the same way (figure 6-b), leading to a loss of information.

Dual graph pyramids [3] are an extension of adjacency graph pyramids, designed in order to take into account multi-adjacency and inclusion (see figure 6-c). A dual graph is defined as a multi-graph together with its dual graph (these two graphs have to be connected). However, dual graphs as well as adjacency graphs can not represent the topological order information for all cases (i.e. for instance the order of faces around a vertex or the order of volumes around an edge or a vertex). For example, figure 7 shows a gray level image representing a clover. It is composed by six parts: the background, the stem, three leaves and the roots (see figure 7-a). The corresponding dual graph is shown in figure 7-b. Due to the self-loop, the edge orientation around vertices is loss and it is not possible to know which leaf lies between the two others. On the other hand, n-G-maps, and so n-G-map pyramids, represent inclusion, multi-adjacency and order, for any dimension (cf. figure 6-d and figure 7-c).

An other drawback of adjacency graphs is the fact that all cells are not represented. For instance in 2D (resp. 3D), vertices (resp. edges and vertices) are not represented. More generally, adjacency graphs only represent n-cells and $n − 1$-cells. n-G-maps represent all cells of any dimension, and all incidence and adjacency relations between these cells. Figure 8 shows the example of a 3D pipe, which extremities share only one edge. The corresponding adjacency graph contains only two vertices (the pipe and the surrounding region) linked by one edge. So, it is not possible to retrieve the information about the two extremities of the pipe. This can be done using a 3-G-map.
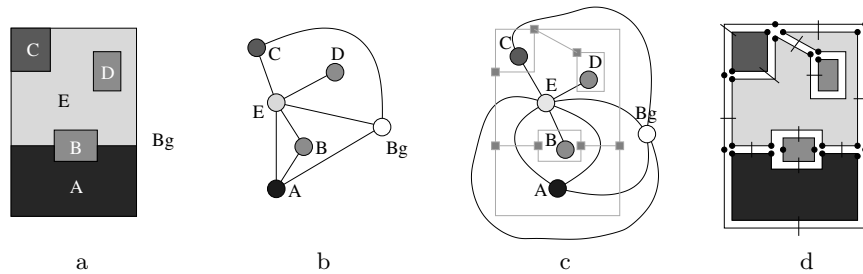
Fig. 6. Representation of a segmented image. (a) An image. (b) The corresponding adjacency graph. (c) The corresponding dual graphs (primal graph in black and its dual in gray). (d) The corresponding 2-G-map.
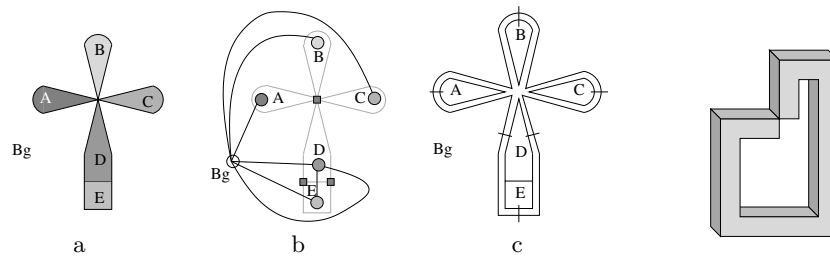

Fig. 7. A 2D image of a clover. (a) The image. (b) The corresponding dual graphs. (c) The corresponding 2-G-map.

Fig. 8. A 3D image representing a pipe.

7 Conclusion and Perspectives

Pyramids of n-dimensional generalized maps are here defined as stacks of reduce n-G-maps where each n-G-map is built from the previous level by contracting or removing cells. n-G-maps unambiguously represent the topology of subdivided n-dimensional objects (for instance n-dimensional images). So, n-G-map pyramids can be used in order to process 2D, 3D and 4D images. n-G-map pyramids can be very useful for applications in which it is necessary to check or to control the (evolution of the) topology of an object, for example when tracking objects in video sequences.

n-G-map pyramids can be represented in different ways. We have discussed three generic representations: *explicit*, *hierarchical* and *implicit*, and their advantages and drawbacks which are important for choosing an efficient representation according to the needs of the application (complexity in memory space and/or in time). But we need to compare more precisely these three representations in practical applications to list advantages, drawbacks and complexities.

n-G-map pyramids have several advantages, compared with adjacency graph pyramids. Mainly, n-G-maps pyramids represent the whole topological informa-

tion about n-dimensional multi-level subdivided objects. It is important since the reduction between levels (achieved by the applications of removal and contraction operation) leads to particular cases (for instance multi-adjacency of cells) which are usually not well handled by graphs.

We intend now to define and generalize the notion of receptive field for n-G-map pyramids. This classical notion, defined for graph pyramids and $2D$ combinatorial map pyramids, establishes a relation between a cell at a given level and the set of cells of lower levels which are contracted or removed into this cell.

It is also necessary to conceive operations for handling this structure: for instance, for modifying any level of the pyramid by contracting or removing a cell, by adding a new cell, etc. An important problem is the fact that it is necessary to efficiently propagate these modifications for the other levels.

8 Acknowledgements

References

1. Montanvert, A., Meer, P., Rosenfeld, A.: Hierarchical image analysis using irregular tessellations. PAMI **13** (1991) 307–316.
2. Jolion, J., Montanvert, A.: The adaptive pyramid : a framework for 2d image analysis. Computer Vision, Graphics and Image Processing **55** (1992) 339–348.
3. Kropatsch, W.: Building irregular pyramids by dual-graph contraction. Vision, Image and Signal Processing **142** (1995) 366–374.
4. Kropatsch, W., Macho, H.: Finding the structure of connected components using dual irregular pyramids. In: Cinquime Colloque DGCI, Université d'Auvergne (1995) 147–158
5. Kropatsch, W.: Abstraction pyramids on discrete representations. In: Discrete Geometry for Computer Imagery. Number 2301 in LNCS, Bordeaux, France (2002) 1–21.
6. Bertolino, P., Ribas, S.: Image sequence segmentation by a single evolutionary graph pyramid. In: Graph-Based Representations in Pattern Recognition, Springer Verlag (1998) 93–100.
7. Dombre, J.: Système de représentation multi-échelle pour l'indexation et la restauration d'archives médiévales couleur. Phd thesis, Université Poitiers (2002)
8. Marfil, R., Rodriguez, J., Bandera, A., Sandoval, F.: Bounded irregular pyramid: a new structure for color image segmentation. PR **37** (2004) 623–626.
9. Brun, L., Kropatsch, W.: Introduction to combinatorial pyramids. In G. Bertrand, A. Imiya, R.K., ed.: Digital and Image Geometry. Volume 2243 of LNCS. Springer Verlag (2001) 108–127.
10. Brun, L., Kropatsch, W.: Combinatorial pyramids. In Suvisoft, ed.: IEEE International conference on Image Processing (ICIP). Volume II., Barcelona, Spain, IEEE (2003) 33–37.

11. Brun, L., Kropatsch, W.: Contraction kernels and combinatorial maps. PRL **24** (2003) 1051–1057.
12. Lienhardt, P.: Subdivisions of n-dimensional spaces and n-dimensional generalized maps. In: Proceedings of the fifth annual symposium on Computational geometry, Saarbruchen, Germany, ACM Press (1989) 228–236.
13. Lienhardt, P.: N-dimensional generalized combinatorial maps and cellular quasi-manifolds. In: International Journal of Computational Geometry and Applications, Hamburg, Germany (1994) 275–324.
14. Damiand, G., Lienhardt, P.: Removal and contraction for n-dimensional generalized maps. In: Discrete Geometry for Computer Imagery. Number 2886 in Lecture Notes in Computer Science, Naples, Italy (2003) 408–419.
15. Lienhardt, P.: Subdivisions of surfaces and generalized maps. In: Proceedings of Eurographics, Hamburg, Germany (1989) 439–452.
16. Fradin, D.: Modélisation et simulation d'éclairage à base topologique: application aux environnements architecturaux complexes. PhD thesis, Université Poitiers (2004)
17. Brun, L., Kropatsch, W.: Implicit encoding of combinatorial pyramids. In Drbohlav, O., ed.: Proceedings of the Computer Vision Winter Workshop, Valtice, Czech Reublic (2003) 49–54
18. Damiand, G.: Définition et étude d'un modèle topologique minimal de représentation d'images 2d et 3d. Phd thesis, Université Montpellier II (2001)
19. Bertrand, Y., Damiand, G., Fiorio, C.: Topological map: minimal encoding of 3d segmented images. In: Workshop on Graph-Based Representations in Pattern Recognition, Ischia, Italy, IAPR-TC15 (2001) 64–73.