Grouping of Non-connected Structures by an Irregular Graph Pyramid*

Walter G. Kropatsch and Yll Haxhimusa

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

Abstract. Motivated by claims to 'bridge the representational gap between image and model features' and by the growing importance of topological properties we discuss several extensions to dual graph pyramids: structural simplification should preserve important topological properties and content abstraction could be guided by an external knowledge base. We review multilevel graph hierarchies under the special aspect of their potential for abstraction and grouping.

1 Introduction

Regions as aggregations of primitive pixels play an extremely important role in nearly every image analysis task. Regional (internal) properties (color, texture, shape, ...) help to identify them and their external relations (adjacency, inclusion, similarity of properties,...) are used to build groups of regions having a particular meaning in a more abstract context. A question is raised in [11] referring to several research issues: "How do we bridge the representational gap between image features and coarse model features?" They identify the 1-to-1 correspondence between: *salient image features* (pixels, edges,...) and *salient model features* (generalized cylinders, invariant models,...) as *limiting assumption* that makes generic object recognition impossible. It is suggested to *bridge* and not to *eliminate the representational gap*, and to focus efforts on: region segmentation, *perceptual grouping* and *image abstraction*.

Connected components form the bases for most segmentations. The region adjacency graph (RAG) describes the relations of connected regions. However not all regions of the RAG have the same importance like a dotted line on white background. In such cases the more important regions are offten close to each other but not adjacent and adjacency prevents further grouping. We overcome this problem by letting more important regions (foreground) grow into the non important regions (background) until the close regions become adjacent and can be grouped. We address some of these issues in the context of gradually generalizing our discrete image data across levels where geometry dominates up to levels of the hierarchy where topological properties become important.

* Supported by the Austrian Science Foundation under grant FSP-S9103-N04.

J.S. Marques et al. (Eds.): IbPRIA 2005, LNCS 3523, pp. 107–114, 2005.

We review the formal definition of abstraction (Sec. 2) and the concept of dual graphs (Sec. 3) including a 'natural' example of vision based on an irregular sampling. Image pyramids of dual graphs are the main focus of Sec. 4. Abstraction in multilevel structures can be done either by modifying the contents of a representational cell or by 'simplifying' the structural arrangement of the cells while major topological properties are preserved (Sec. 5).

2 Visual Abstraction

By definition abstraction extracts essential features and properties while it neglects unnecessary details. Two types of unnecessary details can be distinguished: *redundancies* and *data of minor importance*. Details may not be necessary in different contexts and under different objectives which reflect in different types of abstraction. In general we distinguish: *isolating abstraction*: important aspects of one or more objects are extracted from their original context; *generalizing abstraction*: typical properties of a collection of objects are emphasized and summarized. *idealizing abstraction*: data are classified into a (finite) set of ideal models, with parameters approximating the data and with (symbolic) names/notions determining their semantic meaning. These three types of abstraction have strong associations with well known tasks in cognitive vision: recognition and object detection tries to *isolate* the object from the background; perceptual grouping needs a high degree of *generalization*; and categorization *assigns* data to *ideal classes* disregarding noise and measurement inaccuracies. In all cases abstraction drops certain data items which are considered less relevant. Hence the *importance* of the data needs to be computed to decide which items to drop during abstraction. The importance or the relevance of an entity of a (discrete) description must be evaluated with respect to the purpose or the goal of processing.

Multiresolution hierarchies, image pyramids or trees in general posses the potential for abstraction. We consider the structure of the representation and the content stored in the representational units separately. In our generalization we allow the resolution cell to take other simply connected shapes and to describe the content by a more complex 'language'. The first generalization is a consequent continuation of the observations in [2] to overcome the limited representational capabilities of rigid regular pyramids. Since irregular structures reduce the importance of explicitly representing geometry, topological aspects become relevant.

3 Discrete Representation – Dual Graphs

A digital image is a finite subset of 'pixels' of the discrete grid $\mathbb{Z}^2$. The discretization process maps any object of the continuous image into a discrete version if it is sufficiently large to be captured by the sensors at the sampling points. Resolution relates the unit distance of the sampling grid with a distance in reality. The properties of the continuous object, i.e. color, texture, shape, as well as its relations to other (nearby) objects are mapped into the discrete space,

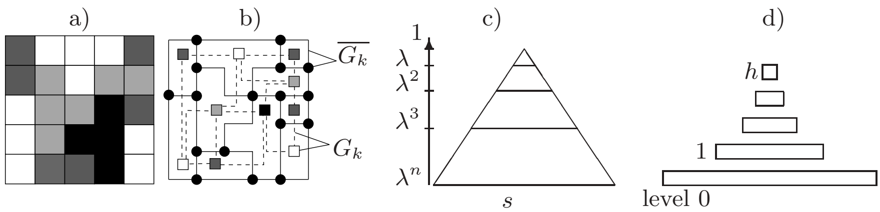


Fig. 1. a) Partition of pixel set into cells. b) Representation of the cells and their neighborhood relations ($G_k, \overline{G_k}$). c) Pyramid concept, and d) discrete levels.

too. The most primitive discrete representation assigns to each sampling point a measurement, be it a gray, color or binary value. In order to express the connectivity or other geometric or topological properties, the discrete representation must be enhanced by a neighborhood relation. In the regular square grid 4- or 8-neighborhood have the well known problems in conjunction with Jordan's curve theorem. The neighborhood of sampling points is represented by a graph. Although this data structure consumes more memory space it has several advantages, among which we find the following: *the sampling points need not be arranged in a regular grid*; *the edges can receive additional attributes too*; and *the edges may be determined either automatically or depending on the data*.

The problem arising with irregular grids is that there is no implicit neighbor definition. Usually Voronoi neighbors determine the neighborhood graph. The neighborhood in irregular grids needs to be represented explicitly. This creates a new representational entity: the binary relation of an edge in the neighborhood graph similar to the concept of relations between observational entities in [5]. Together with the fact that a $2D$ image is embedded in the continuous image plane, the line segments connecting the end points of edges partition the image plane into connected faces which are part of the *dual graph* (Fig. 1a,b).

4 Pyramids

In this section we summarize the concepts developed for building and using multiresolution pyramids [10, 15] and put the existing approaches into a general framework. The focus of the presentation is a representational framework, its components and the processes that transfer data within the framework. A pyramid [15] (Fig. 1c,d) describes the contents of an image at multiple levels of resolution. The base level is a high resolution input image. Successive levels reduce the size of the data by a constant *reduction factor* $\lambda > 1.0$ while local *reduction windows* relate one cell at the reduced level with a set of cells in the level directly below. Thus local independent (and parallel) processes propagate information up and down in the pyramid. The contents of a lower resolution cell is computed by means of a *reduction function*, the input of which are the descriptions of the cells in the reduction window.

The number of levels n is limited by λ: $n \leq \log(image\_size)/\log(\lambda)$. The main computational advantage of *image pyramids* is due to this *logarithmic com-*

plexity. We intend to extend the expressive power of these efficient structures by several generalizations. In order to interpret a derived description at a higher level, this description should be related to the original input data in the base of the pyramid. The *receptive field* (RF) of a given pyramidal cell c_i, $RF(c_i)$, collects all cells (pixels) in the base level of which c_i is the ancestor.

Content Models and Reduction Functions

In connected component labeling each cell contains a label identifying the membership of the cell to the class of all those cells having the same label. In this case the contents of the cells merged during the reduction process can be propagated by simple inheritance: the fused cell 'inherits' its label from its children. In classical gray level pyramids the contents of a cell is a gray value which is summarized by the mean or a weighted mean of the values in the reduction window. Such reduction functions have been used in Gaussian pyramids. Laplacian pyramids [4] and wavelet pyramids [16] identify the loss of information that occurs in the reduced level and store the missing information in the hierarchical structure where it can be retrieved when the original is reconstructed. These approaches use *one single globally defined model* [8] which must be flexible to adapt its parameters to approximate the data.

In our generalization we would like to go one step further and allow *different models* to be used in different resolution cells as there are usually different objects *at different locations* of an image. The models could be identified by a name or a symbol (e.g black, white, isolated etc.) and may be interrelated by semantic constraints (e.g adjacency etc.), Fig. 4. Simple experiments have been done with images of line drawings. This research used the experiences gained with a system for perceptional curve tracing based on regular $2{\times}2/2$ curve pyramid [12] and the chain pyramid [17] in the more flexible framework of graph pyramids. The model describes symbolically the way in which a curve intersects the discrete segments of the boundary of a cell and the reduction function consists in the transitive closure of the symbols collected in the reduction window. The concept works well in areas where the density of curves is low, although the rigidity of the regular pyramid causes ambiguities to arise when more curves appear within the same receptive field. This limitation can be overcome with irregular pyramids [15] in which we could limit the receptive field of a cell to a single curve.

The content abstraction in this representation has following features:

- models are identified by names[1], no parameters were used;
- adjacent models have to be consistent ('good continuation');
- only one consistent curve is covered in one receptive field;
- this selection process is governed by a few contraction rules (Fig. 4).

The knowledge about the models and in what configurations they are allowed to occur needs to be stored in a knowledge base [14]. In order to determine which are the best possible abstractions, the local configurations at a given level of the pyramid must be compared with the possibilities of reduction given in the

[1] Discrete names: empty cell, line end, crosses edge, junction etc.

Algorithm 1 – Graph Pyramid.

Input: Attributed graph G.

1: **while** { further abstraction is possible } **do**
2: determine contraction kernels (CKs),
3: perform dual graph contraction and simplification of dual graph,
4: apply reduction functions to compute content of new reduced level,
5: **end while**

Output: Irregular graph pyramid.

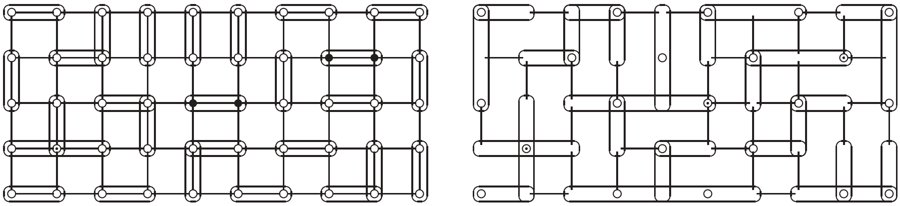
knowledge base. This would typically involve matching the local configuration with the right-hand sides of rules stored in the knowledge base. Such a match may not always be perfect, one may allow a number of outliers. The match results in a goodness of match, which can be determined for all local configurations. The selection can then choose the locally best candidates as contraction kernels (CKs) and reduce the contents according to the generic models which matched the local configuration. The goodness of match may also depend on a global objective function to allow the overall purpose, task or intention to influence the selection process.

5 Irregular Graph Pyramids

A graph pyramid is a pyramid where each level is a graph $G(V, E)$ consisting of vertices V and of edges E relating pairs of vertices. In the base level, pixels are the vertices, and two vertices are related by an edge if the two corresponding pixels are neighbors. This graph is called the neighborhood graph. The content of the graph is stored in attributes attached to both vertices and edges. In order to correctly represent the embedding of the graph in the image plane we additionally store the dual graph $\overline{G}(\overline{V}, \overline{E})$ at each level. Let us denote the original graph as the primal graph. In general a graph pyramid can be generated bottom-up [15] (see Alg. 1).

5.1 1st Iteration: Group Connected Components

The 2^{nd} step determines what information in the current top level is important and what can be dropped. A CK is a (small) sub-tree, the root of which is chosen to survive. Fig. 2a shows the window (G_0) and the selected CK $N_{0,1}$ each surrounded by an oval. The codes of the vertices are given in Fig. 4. Selection criteria (code adjacency of Fig. 4 is 'yes') in this case contract only edges inside connected components except for isolated black vertices (blobs) which are allowed to merge with their background, so that support of grouping is distributed over a large receptive field bridging areas of background [6]. All the edges of the contraction trees are dually contracted [15]. Dual contraction of an edge e (formally denoted by $G/\{e\}$) consists of contracting e and removing the corresponding dual edge $\overline{e}$ from the dual graph (formally denoted by $\overline{G} \setminus \{\overline{e}\}$).



a) Neighborhood graph G_0 and CK N_{01} b) G_1 and CK N_{12};

Fig. 2. Broken line.

This preserves duality and the dual graph need not be constructed from the contracted primal graph G' at the next level.

Since the contraction of an edge may yield multi-edges and self-loops there is a simplification step which removes all redundant multi-edges and self-loops (redundant edges). Note that not all such edges can be removed without destroying the topology of the graph since its removal would corrupt the connectivity! This can be decided locally by the dual graph since *faces of degree two* (having the double-edge as boundary) and *faces of degree one* (boundary = self-loop) cannot contain any further elements in its interior, since the original graph is connected. Since removal and contraction are dual operations, the removal of a self-loop or of one of the double edges can be done by contracting the corresponding dual edges in the dual graph. The dual contraction of our example remains a graph G_1 without redundant edges (Fig. 2b).

5.2 New Category: Isolated Blob

Step 3 generates a reduced pair of dual graphs. The content is derived in step 4 from the level below. In our example, reduction is very simple: the surviving vertex inherits the color of its son. A new category 'isolated blob' is introduced if a black vertex is completely surrounded by white vertices. This new label allows the RF to grow into its background and, eventually, close the gap to another isolated blob. In the only case where the CK contains two different labels, the isolated vertex is always chosen as surviving vertex.

The result of the second dual contraction is shown in Fig. 3. The selection rules and the reduction function are the same as in the first iteration. The isolated blob adjacency graph (IBAG) shows that the gaps between the isolated blobs of the original sampling have been closed and the three surviving isolated blobs are connected after two iterations. A top-down verification step checks the reliability of closing the gap. There are lots of useful properties of the resulting graph pyramids. If the plane graph is transformed into a combinatorial map the transcribed operations form the combinatorial pyramid [3]. This framework allowed to link dual graph pyramids with topological maps which extend the scope to $3D$.

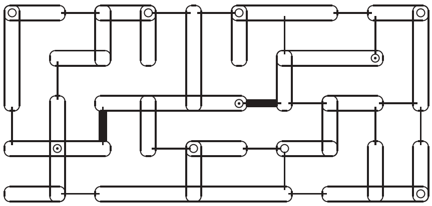
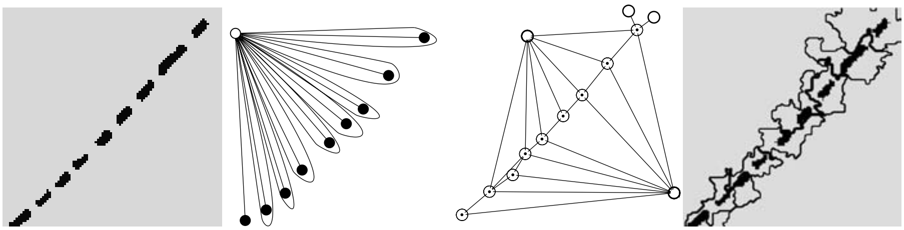


Fig. 3. The two gaps in graph G_2.

may contract with		○	●	⊙
empty background	○	yes	no	yes
black component	●	no	yes	no
isolated blob	⊙	yes	no	yes (*gap*)

Fig. 4. Contraction rules to close gaps.



a) Broken line b) CCA c) IBAG d) RF

Fig. 5. Closing the gaps of a broken line.

6 Experimental Result

Fig. 5 shows an example of closing the gaps of a broken line. Connected components analysis (CCA) alone creates self loops. Growing isolated blobs into its background produces vertices of isolated blobs connected by edges corresponding to the gaps. Fig. 5d shows the corresponding *RF* of the isolated blobs, which represent edgel hypotheses and the neighborhood of isolated vertices a line hypothesis. These hypotheses can be verified for confidence using the hierarchy of the pyramid. It seems that there are much less concepts working on discrete irregular grids than on their regular counterparts. How to group connected structures into an extended RAG has been show before [9]. The many islands of highly split structures remain isolated in these approaches. We show how to group isolated blobs or substructures into IBAG if the blobs have a 'common' background.

7 Conclusion

We motivated our discussion by the claim to *'bridge the representational gap'* [11] and to *'focus on image abstraction'*. We first discussed the basic concepts, visual abstraction and dual graphs in more detail. We then recalled a pyramidal approach having the potential to cope also with irregular grids. These pyramids have some useful properties: i) they show the need to use multi-edges and self-loops to preserve the topology; ii) they allow the combination of primitive operations at one level (i.e. collected by the CK) and across several levels of the pyramid (i.e. equivalent contraction kernels [13]); iii) repeated contraction converges to specific properties which are preserved during contraction; iv) termination criteria allow abstraction to be stopped before a certain property is lost. The new category of an isolated blob allowed to group non adjacent regions based on proximity.

References

1. Y. Bertrand, C. Fiorio, and Y. Pennaneach. Border Map: A Topological Representation for nD Image Analysis. *DGCI*, LNCS 1568:242–257, 1999.
2. M. Bister, J. Cornelis, and A. Rosenfeld. A Critical View of Pyramid Segmentation Algorithms. *Pattern Recognition Letters*, 11(9):605–617, 1990.
3. L. Brun and W. G. Kropatsch. Introduction to Combinatorial Pyramids. *Dig. and Im. Geom.*, 108–128, 2001.
4. P. J. Burt and E. H. Adelson. The Laplacian Pyramid as a Compact Image Code. *IEEE Trans. on Commun.*, 31(4):532–540, 1983.
5. J. L. Crowley, J. Coutaz, G. Rey, and P. Reignier. Perceptual Components for Context Aware Computing. In *Int. Conf. on Ubiq. Comp.*, 117–143, 2002.
6. S. C. Dakin and P. Bex. Local and Global Visual Grouping: Tuning for Spatial Frequency and Contrast. *J. of Vis.*, 99–111, 2001.
7. G. Damiand. *Définition et Étude d'un Modèle Topologique Minimal de Représentation D'images 2d et 3d.* PhD thesis, LIRMM, Uni. de Montpellier, 2001.
8. R. L. Hartley. *Multi–Scale Models in Image Analysis*. PhD thesis, Uni. of Maryland, CSC, 1984.
9. Y. Haxhimusa and W. G. Kropatsch. Hierarchical Image Partitioning with Dual Graph Contraction. *DAGM Symp.*, LNCS 2781:338–345, 2003.
10. J.-M. Jolion and A. Rosenfeld. *A Pyramid Framework for Early Vision.* Kluwer, 1994.
11. Y. Keselman and S. Dickinson. Generic model abstraction from examples. In *CVPR.*:856–863, Hawaii, 2001.
12. W. G. Kropatsch. Preserving Contours in Dual Pyramids. *ICPR*, 563–565, 1988.
13. W. G. Kropatsch. From Equivalent Weighting Functions to Equivalent Contraction Kernels. *DIP and CG2:*, SPIE 3346:310–320, 1998.
14. W. G. Kropatsch. Abstract Pyramid on Discrete Represtations. *DGCI* LNCS 2301, 1–21, 2002.
15. W. G. Kropatsch, A. Leonardis, and H. Bischof. Hierarchical, Adaptive and Robust Methods for Image Understanding. *Sur. on Math. for Ind.*, 9:1–47, 1999.
16. S. G. Mallat. A Theory for Multiresolution Signal Decomposition: The Wavelet Representation. *IEEE Trans. PAMI*, 11(7):674–693, 1989.
17. P. Meer, C. A. Sher, and A. Rosenfeld. The Chain Pyramid: Hierarchical Contour Processing. *IEEE Trans. PAMI*, 12(4):363–376, 1990.