

PROPERTY PRESERVING HIERARCHICAL GRAPH TRANSFORMATIONS

WALTER G. KROPATSCH

*Vienna University of Technology, Institute of Automation 183/2
Pattern Recognition and Image Processing Group, Treitlstr.3, A-1040 Wien, Austria
email: krw@prip.tuwien.ac.at*

Minsky and Papert proved in 1969 that connectivity of an arbitrary pixel set X cannot be computed locally in general. Their theorem essentially depends on their two stage architecture. The present paper introduces a pyramidal strategy to solve similar complex problems with the major difference that the data are not organized in a regular grid but in planar graphs and that the repeated transformations depend adaptively on the data. The algorithm is demonstrated by two examples: (1) Finding connected boundary segments and how they are connected with each other; (2) Given two arbitrary non-intersecting curves in an image determine whether two marks 'x' are placed on the same or on different curves. In both cases computational complexity depends on $\log|X|$. It is made possible by controlling the contraction process in a way that does not destroy the desired global property.

1 Introduction

The analysis of shapes often involves checking properties like connectivity, convexity, circularity, arrangement of curvature extrema along the boundary etc. (see Nagy[?] for many other shape extraction methods). Minsky and Papert[?] describe such properties formally by predicates $\psi(X)$ of a subset X of pixels belonging to the shape. Our approach allows the domain of X to be any subset of regions of a partition of the continuous image plane I . A given shape boundary intersects the partition cells $X \subset I$ which we shortly call boundary segments.

The connectivity predicate $\psi_{\text{CONNECTED}}(X) = \text{true}$ if the set of boundary segments X is connected in the image I . Checking the connectivity of a set X involves finding connected paths between any pair of elements in X . This is a difficult task even for humans. A simplified psychological test lead to conjectures about the type of visual routines used in human perception. The image contains two separate curves and two marks 'X', placed on the same or on different curves (Fig. 1). The two curves $C_1, C_2 \subset I$ do not intersect each other but may have complicated intertwined shapes. The task is to decide, as rapidly as possible, whether the two Xs are on the same curve expressed by $\psi_{xx}(X) = \text{true}$ or on different curves, $\psi_{xx}(X) = \text{false}$.

Psychophysical experiments[?] suggest that people can trace curves internally at high speed, and that curve tracing is a 'visual routine'. Fig. 1(a)

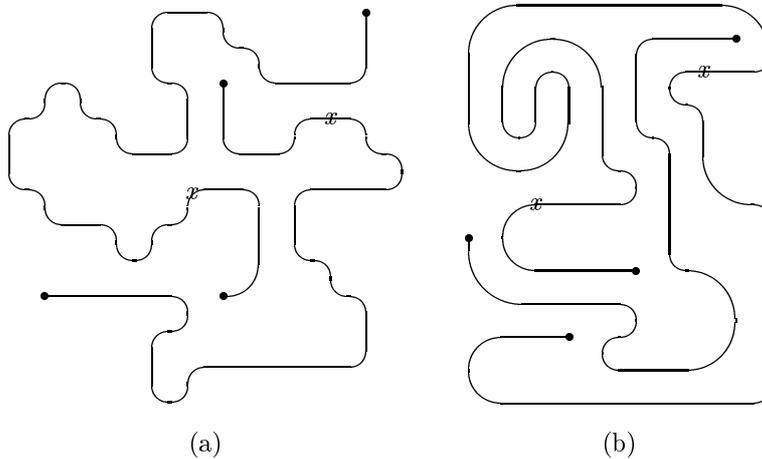


Figure 1: Are the 2 x on the same curve?

approximates the configuration of the experiments used by Jolicoeur[?]. Sequential curve tracing was hypothesized from their experiments because (1) 'more time was required ... when longer distances along the curve separated the two curves' and because (2) 'response times were slower, on average, for "different" trials than for "same" trials'.

The complexity of the task is related to a principle problem of $\psi_{\text{CONNECTED}}$ as pointed out by Minsky and Papert[?]: ' $\psi_{\text{CONNECTED}}$ is not conjunctively local of any order.' They assume a two stage computation: in stage I many properties are computed in parallel and each individual property accesses maximally $k < n = |X|$ elements of X ; Stage II combines all results of stage I in a simple way (e.g. by logical AND). The above theorem refers to this two stage computation scheme and it means that the computational complexity of computing the connectivity of an arbitrary set $X \subset I$ needs $\mathcal{O}(|X|)$ steps.

In this paper we show a computational solution that needs less complex computations. In our case one computation stage consists in a transformation $\tau : X \mapsto X'$ that reduces the data size by a factor $\lambda > 1$

$$|X'| < |X|/\lambda \quad (1)$$

while the reduced data set X' preserves the property to compute:

$$\psi(X') = \psi(X) \quad (2)$$

The application of τ can be repeated on the reduced data set X' until any

further reduction would destroy property ψ . Assume that $X^{(n)} = \tau^n(X)$ is the result after n repetitions. From (1) follows that the overall size reduction is bounded by $\lambda^n < \frac{|X|}{|X^{(n)}|}$ and also the number n of repetitions:

$$n < \frac{\log |X| - \log |X^{(n)}|}{\log \lambda} \quad (3)$$

Together with constraints (1) and (2) and the assumption that τ needs only local parallel computations $\psi(X) = \psi(X^{(n)})$ can be computed in $\mathcal{O}(\log |X|)$ parallel steps.

Classical regular pyramids[?] possess property (1), but, in general, counter examples for property (2) can be constructed[?]. It has been demonstrated[?] that adaptive pyramids in which transformation τ depends on the data X overcome the problem. The price to pay is the loss of regularity. Section 2 shortly reviews the transition from regular pixel grids to graph structures and the specific transformation τ to contract graphs: dual graph contraction. Then selection rules are described (in section 3) that preserve the connectivity of the boundary based shape representation. Section ?? relates the efficiency of the method with results from psychophysical experiments. The conclusion summarizes the paper and gives an outlook to several related issues.

2 From Pixels to Graphs

Fig. 2 illustrates how a pixel grid can be represented by a neighborhood graph $G(V, E)$ and its dual face graph $\overline{G}(F, \overline{E})$. Vertices V correspond to pixels, geometrical properties like coordinates or gray value are stored as attributes. Vertices containing curve information will be called cells to highlight this fact. Two vertices are joined by an edge if the corresponding pixels are neighbors^a. The faces of the dual graph are placed inside the square formed by 4 vertices. Two faces are joined by a dual edge if the corresponding squares are adjacent in G . There is a one-to-one relationship between E and \overline{E} . This graph representation extends the scope to any tessellation of the image plane.

Dual graph contraction $(G', \overline{G}') = C[(G, \overline{G}), (S, N)]$ is the transformation τ that reduces the size of the pair of dual graphs using contraction kernels (S, N) as parameters[?]. Contraction kernels can be any spanning subgraph of G without cycle. The roots of the forest (V, N) are called surviving vertices and they constitute the vertices of the contracted graph $G', V' = S \subset V$. The contraction process contracts all edges N into the root in V' , edges E' are established in G' if the two surviving endpoints are connected in G by a

^a4-neighbors are used because they make G planar.

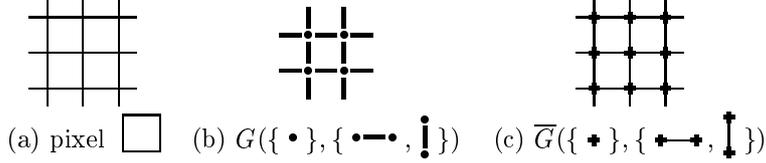


Figure 2: (a) pixel grid, (b) neighborhood graph $G(V, E)$, (c) dual face graph $\overline{G}(F, \overline{E})$

connecting path consisting of two branches of the two contraction kernels and one edge $e \in E \setminus N$ which is called a bridge. We shall control this process by a data dependent selection of the contraction kernels such that the predicate ψ is preserved.

The stack of successively reduced graphs $((G_0, \overline{G}_0), (G_1, \overline{G}_1), \dots, (G_n, \overline{G}_n))$, $G_0 = G$, is called a dual irregular pyramid. Any number of contraction kernels can be combined together in an equivalent contraction kernel (ECK) that contracts the base graph G directly into the top graph. Every tree of the ECK spans the complete receptive field of its root vertex i.e. all vertices of V_0 that have been contracted into this root.

3 Connectivity Preserving Boundary Contraction

The following solution combines the concepts of curve relation and dual irregular pyramid. In contrast to the irregular curve pyramid[?], here, the cells where the curves are represented are contracted and not the dual graph. Like in the regular curve pyramid[?], curve and cell may be related by several cell classes:

- 0-cell ...** denotes an empty cell (no curve within the receptive field)
- 2-cell ...** one single curve crosses the receptive field; it enters and exits the cell at two particular boundary segments (e.g. across two edges $e_1, e_2 \in E$).
- 1-cell ...** a curve ends in this cell (a filled circle marks 1-cells in the figures); it enters the cell at a particular boundary segment.
- *-cell ...** a cell where more than one curves meet.^b

Fig. 3(a) overlays a pixel grid and a curve network. In a first phase all cells in the base (squares in Fig. 3(a)) are assigned to one of the four cell-classes. A simple algorithm starts with edges between 4-adjacent pixels: If a curve

^bSince both $\psi_{\text{CONNECTED}}$ and ψ_{xx} do not need any differentiation between different types of crossings, it is considered also as the end of all curves that meet there.

crosses the edge the edge receives an attribute 1 otherwise 0. Then all pixels sum the attributes of their incident edges. Sums 0, 1, and 2 correspond the 0-cell, 1-cell, and 2-cell, respectively, cells with a sum higher than 2 are *-cells. It is assumed that

1. the cell classes are consistent, i.e. if a curve crosses a boundary segment both cells adjacent to this segment are in the correct class;
2. all curves are well distinguishable in the base, e.g. there are no more than one single curve in one single cell of the base (except at *-cells).

3.1 *Selecting the Contraction Kernels*

Fig. 3b,c,d illustrate three successive contractions. Since contraction of vertices corresponds to merging receptive fields these are drawn as closed lines. Root vertices are indicated by circles.

Selection rules: 1-cells and *-cells must always survive. *-cells are not allowed to have children. This prevents the area of unclear information^c from growing. Branches of contraction kernels follow the curves if possible and are selected in following order: 1-cell, 2-cell, 0-cell. Receptive fields of contracted edges merge and collect the information from the individual parts:

1. A 1-cell can merge with its adjacent 2-cell (this is the only neighbor connected to the 1-cell by an edge with attribute 1), then with any adjacent 0-cell and will become an 1-cell again;
2. a 2-cell can merge with both adjacent 2-cells (connecting edges must have attribute 1!) or with any adjacent 0-cell and remains a 2-cell;
3. a 0-cell can merge with any adjacent cell and remains a 0-cell only if it is merged with another 0-cell.

Random selection (like in adaptive pyramids[?]) applies whenever the above rules do not determine the contraction kernels completely. If surviving cells form a maximum independent set connecting paths have at least length two and hence cells along boundaries are reduced by a factor $\lambda > 2$.

During contraction the bridges of connecting paths inherit their attribute to the new edge. It can be easily verified that the above merging rules preserve the consistency between edge attributes and cell classes, i.e. the sum of attributes of incident edges determine the class of cells. When a 1-cell meets another 1-cell (the other end!) we have found a complete curve segment.

^cCurves may intersect or may be just close to each other.

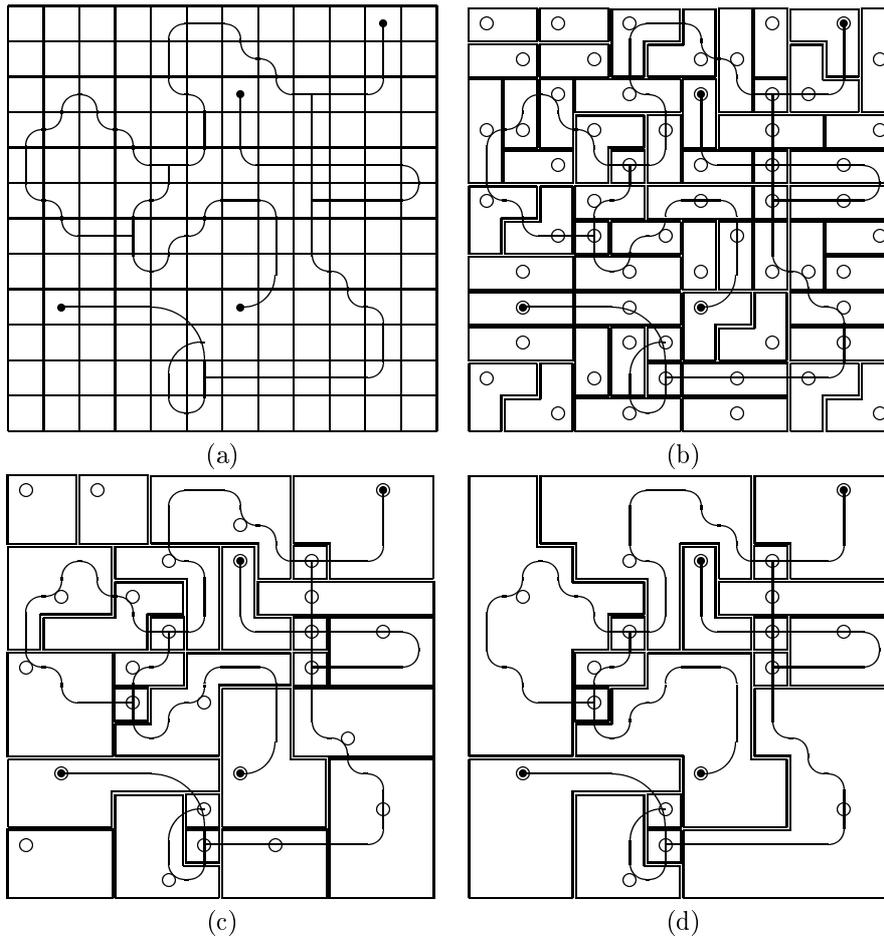


Figure 3: Three step boundary contraction preserving $\psi_{CONNECTED}$

3.2 The Final Graph

Above selection criteria determine the parameters with which we dually contract the graphs until no further contraction is possible. The resulting graph has following properties (verify in Fig. 3d):

- The number of 1-cells (4 in Fig. 3d) and the number of *-cells (7 in Fig. 3d) correspond to the base graph.
- No more 0-cells are present with one exception: An empty area completely surrounded by *-cells leaves an isolated 0-cell.
- 2-cells may appear in only two cases:
 - (a) between two *-cells (7 in Fig. 3d), or
 - (b) forming a self-loop.

Curves remain in separated cells if they are distinct in the base. In this final graph all empty space has been contracted, and all curves have been contracted to minimal length, but all connectivity information of the base is still present. The receptive fields depend the cell classes in the top level:

- *-cells correspond to *-cells in the base one-to-one;
- the receptive field of a 2-cell located between two *-cells consists of the sequence of 2-cells in the base that connect the two *-cells plus some 0-cells on both sides of the curve;
- the receptive field of a 2-cell self-loop accumulates all 2-cells forming the closed curve in the base, the complete interior area of 0-cells if there is no other curve inside, and some 0-cells outside;
- the receptive field of two adjacent 1-cells collects all cells the corresponding curve crosses in the base plus some 0-cells at the sides;
- the receptive field of a 1-cell adjacent to a *-cell is the complete set of 2-cells that connect the 1-cell in the base with the other *-cell plus some 0-cells at the sides.

The method finds a topologically correct ($\psi_{\text{CONNECTED}}(G) = \psi_{\text{CONNECTED}}(G_{(n)})$) and minimal description for all possible (planar) configurations of curves as complicated the layout may be, as long as the sampling allows discrimination of curves in the base. Even strange cases as space filling curves can be treated since the receptive fields may have arbitrary shapes as long as they are planar and connected. In those cases there exist spanning trees covering the curve that can serve as kernels for contraction.

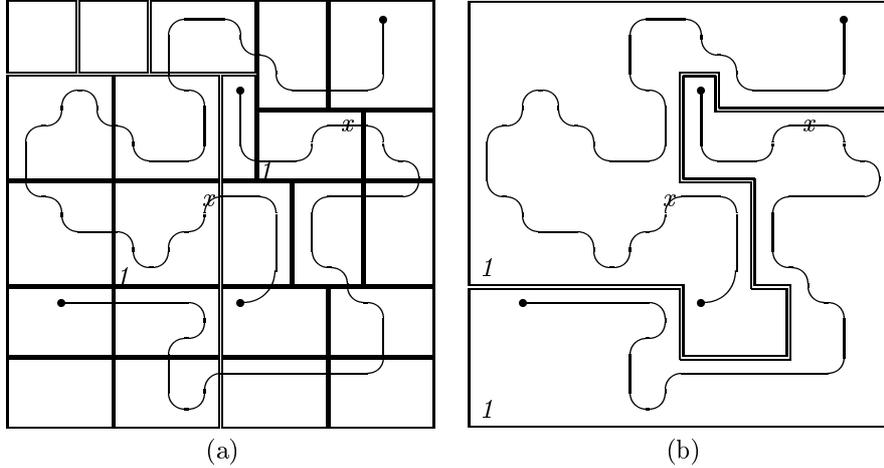


Figure 4: Step 1 and 3 of boundary contraction preserving $\psi_{xx} = false$.

4 Contracting Two Xs On a Curve

We use 0-, 1-, and 2-cells as introduced for computing $\psi_{\text{CONNECTED}}$, *-cells do not appear in the 2x problem. X-marks can be implemented as additional numeric labels of cells, e.g. $\text{label}(2\text{-cell})=k, k \in \{0, 1, 2, \dots\}$ if k x-marks are found in the receptive field of the 2-cell. Labels greater than 0 are displayed in figures ?? and ?? in the left bottom corner of the box delineating the receptive field. All cells are filled with the appropriate labels in the base. It is assumed that, in addition to the two consistency conditions for $\psi_{\text{CONNECTED}}$, the two x-marks are correctly detected and entered as $\text{label}(\text{cell})=1$.

Contraction proceeds as in the general case with following additions: If two 2-cells or 1-cells are merged their labels are summed. The two curves remain in two separated cells if they are distinct in the base. Note that dual graphs need not be considered in this case because we know that the image contains two separate curves and, consequently, no cycles can appear that give rise to faces in the final result. The second example of Figure 1(b) leads to $\psi_{xx} = true$ after four contractions (Fig. ??(a)-(d)).

Our proposal offers a computational solution to this problem that needs $\mathcal{O}(\log(\text{curvelength}))$ parallel steps. This is consistent with experimental observations: if both x are on the "same" curve, the response can be given after the time needed to contract the arc in between both x whereas the "different"-response needs at least the shorter curve to be completely contracted.

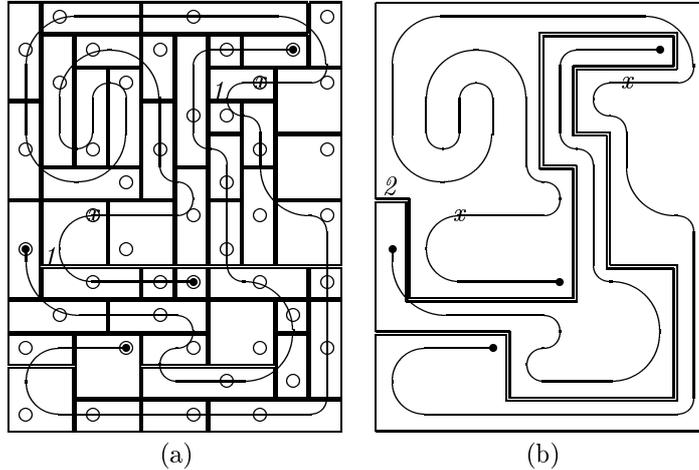


Figure 5: Steps 1 and 4 of example "same" with $\psi_{xx} = true$

5 Conclusion and Outlook

Dual graph contraction transforms graphs into a hierarchy of smaller graphs without destroying properties like connectivity: $\psi_{CONNECTED}$ and ψ_{xx} can be computed in $\mathcal{O}(\log|X|)$ steps. Hierarchical aggregation replaces sequential search and speeds up the computational complexity.

The generality of the approach lies in a clear separation between the contraction process and the choice of control parameters that can be adapted to many different predicates ψ . The efficiency is based on the fact that the contraction can reduce the data by a factor $\lambda > 1$ without losing any piece of information necessary for computing the predicate ψ .

The presented contour segmentation complements the universal region segmentation[?]. Both partition the image plane completely but in a different way: Contours remain in the interior of ECK in the contour segmentation, whereas they separate regions in the region segmentation. Regions in the contour segmentation are split among the contour segments that surround them. The bridges of the contour segmentation that fall inside a region form a sort of axis of the region.

Acknowledgments

The author thanks Jean-Michel Jolion and Horst Bischof for contributing several ideas. Part of the work was supported by the Austrian National Fonds zur Förderung der wissenschaftlichen Forschung under grant S7002-MAT.

References

1. George Nagy. An Induced Taxonomy of Visual Form. In Carlo Arcelli, Luigi P. Cordella, and Gabriella Sanniti di Baja, editors, *2nd Intl. Workshop on Visual Form*, pages 410–419, Capri, Italy, June 1994.
2. Marvin Lee Minsky and Seymour A. Papert. *Perceptrons: An Introduction to computational geometry*. The MIT Press, expanded edition, 1988.
3. Pierre Jolicoeur, Shimon Ullman, and Marilyn Mackay. Curve tracing: A possible basic operation in the perception of spatial relations. *Memory & Cognition*, 14(2):129–140, 1986.
4. Jean-Michel Jolion and Azriel Rosenfeld. *A Pyramid Framework for Early Vision*. Kluwer Academic Publishers, 1994.
5. M. Bister, J. Cornelis, and Azriel Rosenfeld. A critical view of pyramid segmentation algorithms. *Pattern Recognition Letters*, Vol. 11(No. 9):pp. 605–617, September 1990.
6. Walter G. Kropatsch and Souheil BenYacoub. A revision of pyramid segmentation. In Walter G. Kropatsch, editor, *13th International Conference on Pattern Recognition*, volume II, pages 477–481. IEEE Comp.Soc., 1996.
7. Walter G. Kropatsch and Dieter Willersinn. Parallel line grouping in irregular curve pyramids. In *Proceedings Computer Vision and Pattern Recognition - CVPR'93*, pages 784–785. IEEE Comp.Soc.Press, 1993.
8. Walter G. Kropatsch. Curve Representations in Multiple Resolutions. *Pattern Recognition Letters*, Vol. 6(No. 3):pp.179–184, August 1987.
9. Jean-Michel Jolion and Annick Montanvert. The adaptive pyramid, a framework for 2D image analysis. *Computer Vision, Graphics, and Image Processing: Image Understanding*, 55(3):pp.339–348, May 1992.