An Efficient Algorithm for Decomposing a Polygon into Star-Shaped Polygons

David Avis *and* **Godfried Toussaint**

School or Computer Science, McGill University, 3480 University Street, Montreal, Quebec H3A 2A7, Canada

Published in: *Pattern Recognition*, vol. 13, No. 6, 1981, pp. 395-398.

ABSTRACT

In this paper we show how a theorem in plane geometry can be converted into an O(n log n) algorithm for decomposing a polygon into star-shaped subsets. The computational efficiency or this new decomposition contrasts with the heavy computational burden of existing methods.

1.0 Introduction

The decomposition of a simple planar polygon into simpler components plays an important role in syntactic pattern recognition. Some examples of possible decompositions are decompositions into convex polygons [1], [2], decompositions into spiral polygons [3] and decompositions into monotone polygons [4]. A survey of these methods and many other additional references are contained in Pavlidis [5].

A *star-shaped* polygon is one in which the entire polygon is visible from at least one fixed point of the polygon. In this note we consider decompositions into star-shaped polygons and give an efficient algorithm for this problem. A similar decomposition has previously been suggested by Maruyama [6] in his thesis. His solution involves the creation of new *Steiner* points, yields overlapping star-shaped components and requires a complicated and expensive computation.

The main part of this paper describes an O(n log n) algorithm for decomposing a polygon with n vertices into disjoint star-shaped polygons. This decomposition does not involve the creation of any new vertices and can always yield a decomposition with at most [n/3] star-shaped polygons ([x] here denotes the greatest integer less than or equal to x). It does not, however, normally give a decomposition into the minimum number of star-shaped polygons. On the other hand, our procedure is extremely flexible and can easily be modified to give a set of radically different decompositions.

Figure 1: A simple polygon (the Chinese character for “reach” [3]).

Our algorithm follows closely Fisk’s constructive proof [7] of the following theorem due to Chvatal [8]: for every polygon with n vertices there exists a decomposition into at most [n/3] disjoint star-shaped polygons.

2.0 The Algorithm

Before stating the algorithm we introduce a few terms. A *triangulation* of a simple polygon P is a planar graph formed by adding as many non-intersecting edges as possible between the vertices of P, such that each edge lies inside P. Figures 1 and 2 illustrate a polygon and a possible triangulation. By a *coloring* of a triangulation, T, we mean an assignment of colors to the vertices of T such that no two vertices with the same color are adjacent. Clearly any coloring of T requires at least three colors. It can be shown that in fact three colors suffice. The numbers attached to the vertices of the triangulation in Figure 2 indicate a coloring using the three colors {1,2,3}.

From a three-coloring of T we can obtain three different decompositions of P into star-shaped polygons. Consider the set of vertices receiving the color $\{i\}$. Denote this set of vertices by $S_i=\{s_1, s_2,...,s_j\}$. P may be decomposed into j star-shaped polygons P_1, P_2,...,P_j where $P_k = \{x \mid x \text{ is a vertex of P and x is adjacent to } s_k \text{ in} T\} \cup \{s_k\}$, for k=1,2,...,$j$. Furthermore the star-shaped polygons are non-overlapping and:

$$P = P_1 \cup P_2 \cup ... \cup P_j. \quad (1)$$

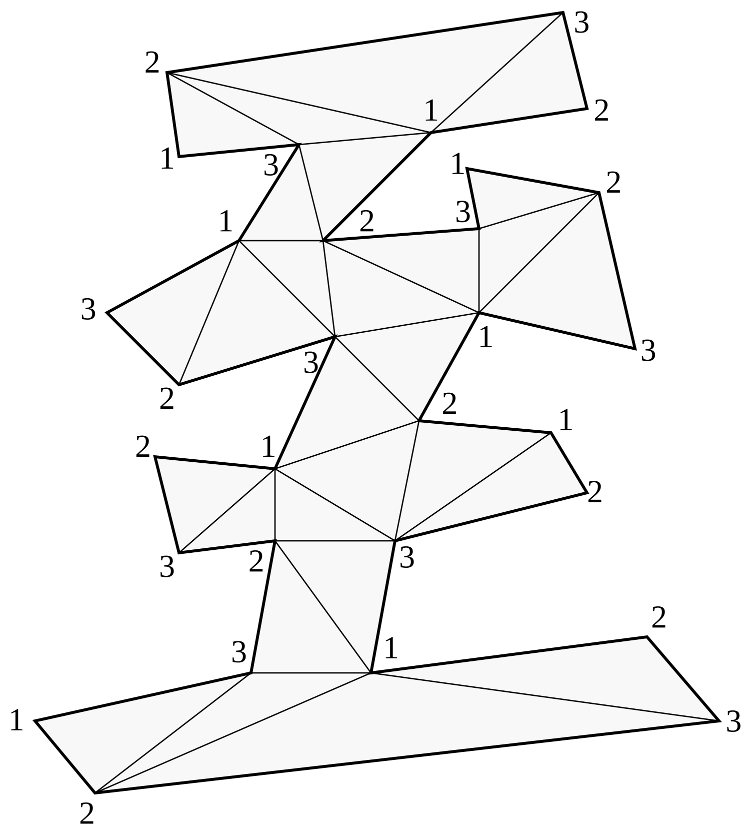

Figure 2: Triangulated and colored polygon.

Consider any triangle {x, y, z} of T. At least one of these vertices, say x, must receive color {i} and is therefore contained in S_i. Suppose x = s_k. Then the triangle {x, y, z} is contained in P_k. Since every triangle of T is contained in some star-shaped polygon, equation (1) follows.

Clearly each of the three colors used in coloring T induces a different decomposition. The three decompositions for the coloring in Fig. 2 are shown in Figs 3, 4 and 5.

The algorithm described above is now summarized.

procedure DECOMPOSE(P)

Step 1: Obtain a triangulation T of P.

Step 2: Color the vertices of T with colors {1, 2, 3}.

Step 3: Select a color arbitrarily and output each vertex with the chosen color together with a list of all adjacent vertices. (If a decomposition with at most [n/3] star polygons is desired, then the color with the minimum number of vertices should be chosen.)

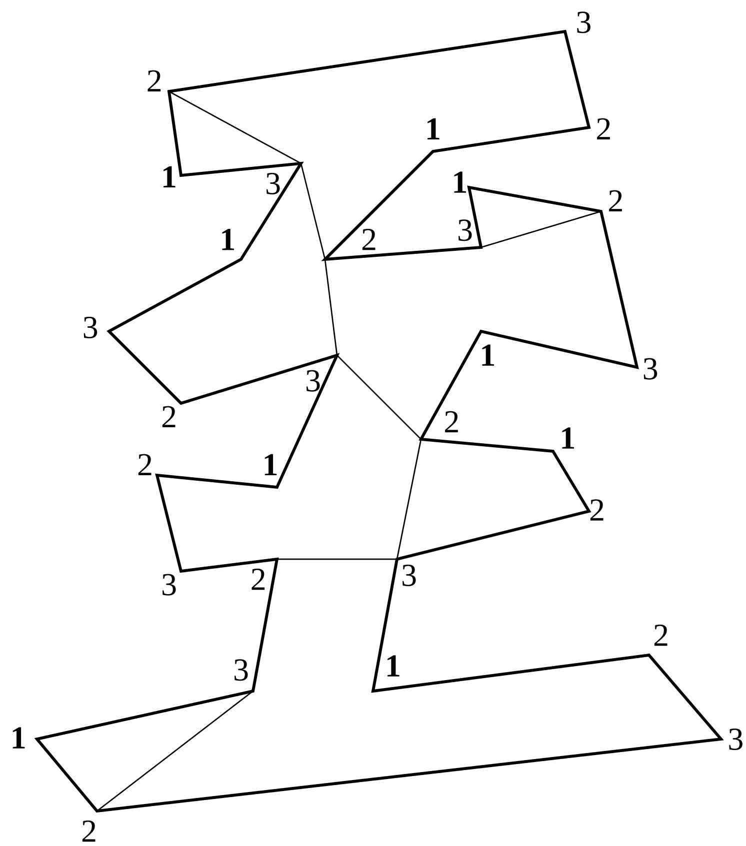

Figure 3: Decomposition using color class 1.

It is clear that the procedure DECOMPOSE is quite flexible, since in general a polygon has many different triangulations. It is straightforward to implement DECOMPOSE in $O(n^2)$ time, using a variety of triangulation strategies. We conclude this section by describing an implementation that requires O(n log n) time.

Firstly, Step 1 may be performed in O(n log n) time using an algorithm due to Garey et al. [9]. Step 2 may also be implemented in O(n log n) time by a divide and conquer technique that is described below. The third step, as it stands, merely identifies the vertices of each star-shaped polygon, and requires just O(n) time. If required, these vertices can be put in sorted angular order, by using the edges of T, also in O(n) time. If the relationships between the various star-shaped polygons are required, an appropriately modified Step 3 can be inserted.

It is immediate that every edge of T divides P into two smaller triangulated polygons. In order to use divide and conquer, we search for an edge that divides T into parts of roughly the same size. We then use the same technique repeatedly on the parts, and finally merge the two parts to give a coloring for P. This method yields an O(n log n) algorithm if the following two conditions are satisfied [10, p. 60].

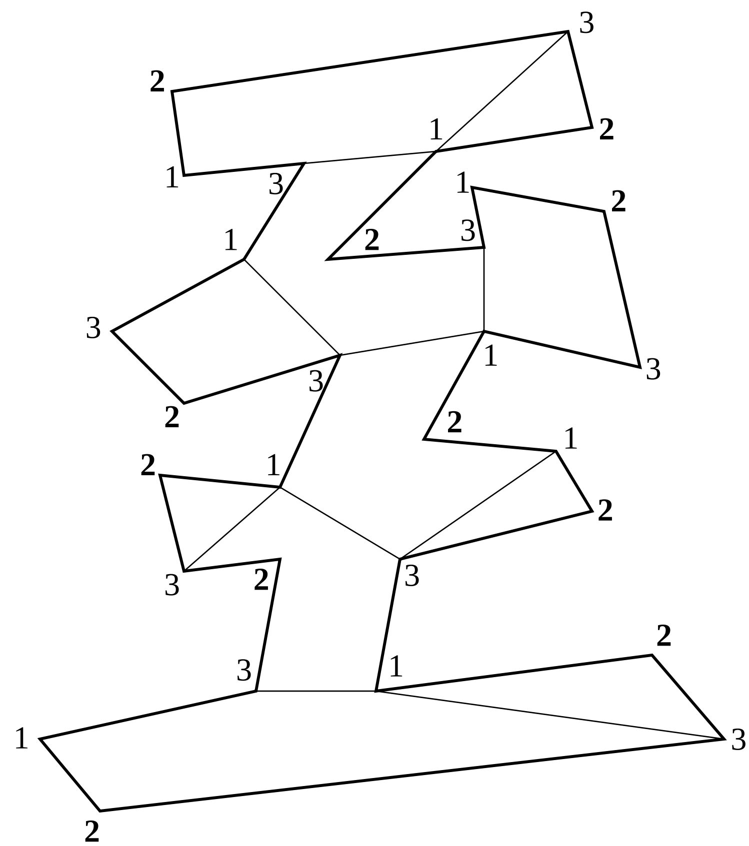

Figure 4: Decomposition using color class 2.

(a) At the divide step, each part should contain at least a fixed fraction, say one quarter, of the vertices.

(b) Both the divide and merge steps require O(n) time.

We sketch a proof of both conditions. For condition (a), consider dividing the boundary of T into four chains, A, B, C, D, each of length at least [n/4] (refer to Fig. 6). If there is any edge between chains A and C the proof is complete since this edge splits P into two chains of at least [n/4] vertices each. On the other hand, since T is a triangulation, if there are no edges between A and C there must be edges between chains B and D. Any such edge suffices for the divide step. To verify condition (b), consider first the divide step. This can be implemented in O(n) time by numbering the vertices of P sequentially from 1 to n and by then testing each of the O(n) edges to see if it satisfies condition (a). This shows that the divide step runs in O(n) time. For the merge step, we consider the colors that are assigned to the separating edge in each of the two parts. If these colors are the same in both parts, the merge step is trivial. Otherwise, we relabel the colors in one of the parts so that the colors match on the separating edge. This takes only O(n) time, so condition (b) is verified.

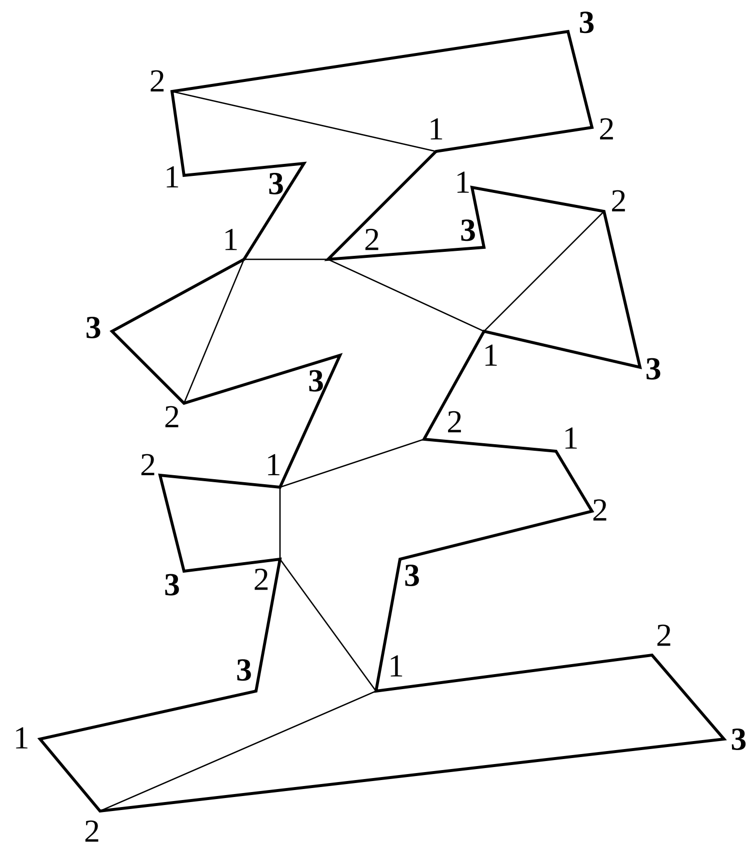

Figure 5: Decomposition using color class 3.

3.0 Conclusions

In this paper we have shown how a constructive proof of a theorem in plane geometry may be implemented into an efficient procedure for polygonal decomposition. It is clear from the examples that the decompositions can be improved by various post-processing steps. One such step would be to remove any edges that create triangles. In addition, different triangulation methods may give improved decompositions. An interesting open problem would be to try to find the decomposition into the minimum number of star-shaped polygons. For the case of convex decompositions, this problem was solved by Chazelle and Dobkin [11]. For the more special case when the polygons contain only vertical and horizontal edges, the decomposition into the minimum number of rectangles was solved by Ferrari et al. [12]. For additional recent results on polygon decomposition the reader is referred to Toussaint [13], [14].

4.0 Summary

The decomposition of a simple planar polygon into simpler components plays an important role in syntactic pattern recognition. One example of a *simpler component* is a *star-shaped* polygon. A star-shaped polygon is one which contains a subset, termed the *kernel*

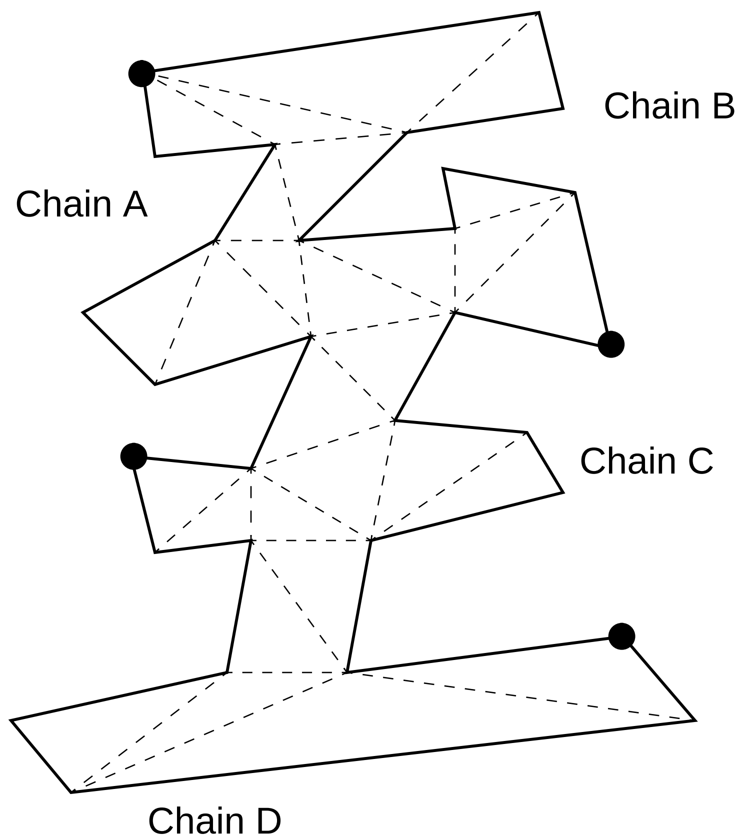

Figure 6: Division of the boundary of a triangulated simple polygon into four chains A, B, C and D.

of the polygon, such that from any point in the kernel the entire polygon is visible. This paper describes an O(n log n) algorithm for decomposing a simple polygon with n vertices into disjoint star-shaped polygons. This decomposition does not involve the creation of new vertices and can always yield a decomposition with at most [n/3] star-shaped polygons. The algorithm is based on Fisk's constructive proof of Chvatal's watchman theorem which states that for every simple polygon with n vertices one needs at most [n/3] guards to ensure that the entire polygon is completely visible by the guards at all times. The algorithm consists of three parts: triangulating the polygon, coloring the triangulation with three colors, and constructing the star-shaped polygons. The first part can be done in O(n log n) time using an algorithm due to Garey et al. [9]. A divide-and-conquer technique is presented for coloring the triangulation in O(n log n) time under the minimal assumption that the triangulation algorithm yields only a list of diagonals without adjacency information. Finally, the construction of the star shaped polygons can be done in O(n) time.

5.0 References

[1] T. Pavlidis, "Representation of figures by labelled graphs," *Pattern Recognition* 4, 5-17 (1972).

[2] B. Schacter, "Decomposition of polygons into convex sets," *IEEE Trans. Comput*. C-27, 1078-1082 (1978).

[3] H. Feng and T. Pavlidis, "Decomposition of polygons into simpler components: feature generation for syntactic pattern recognition, *IEEE Trans. Comput*. C-24 (1975).

[4] D. T. Lee and F. P. Preparata, "Location of a point in a planar sub-division and applications," *SIAM J. Comput*. 6, 594-606 (1977).

[5] T. Pavlidis, "A review of algorithms for shape analysis," Technical Report 218, Department of Electrical Engineering and Computer Science, Princeton University, September (1976).

[6] K. Maruyama, *A study of visual shape perception*. Department of Computer Science, University of Illinois, Urbana, October (1972).

[7] S. Fisk, "A short proof of Chvatal's watchman theorem," *J. Combin. Theory*, B 24, 374 (1978).

[8] V. Chvatal, "A combinatorial theorem in plane geometry," *J. Combin. Theory*. B 18, 39-41 (197S).

[9] M. Garey, D. Johnson, F. Preparata and R. Tarjan, "Triangulating a simple polygon," *Inf. Process. Lett*. 7, 175-179 (1978).

[10] B. Chazelle and D. Dobkin, "Decomposing a polygon into its convex parts," *11th ACM Symp. Theory of Computing*, pp. 38-48 (1979).

[11] A. Aho, J. Hopcroft, J. Ullman, *The Design and Analysis of Computer Algorithms*. Addison-Wesley (1974).

[12] L. Ferrari, P. V. Sankar and J. Sklansky, "Minimal rectangular partitions of digitized blobs," *Proc. 5th Int. Conf. Pattern Recognition*, pp. 1040-1043, Miami Beach, December (1980).

[13] G. T. Toussaint, "Pattern recognition and geometrical complexity," *Proc. 5th Int. Conf. Pattern Recognition*, pp. 1324-1347, Miami Beach, December (1980).

[14] G. T. Toussaint, "Decomposing a simple polygon with the relative neighborhood graph," *Proc. Allerton Conf*. Urbana, Illinois, October (1980).