# On Restricted Boundary Covers and Convex Three-Covers

Patrice Belleville[1]
School of Computing Science
Simon Fraser University
Burnaby, B.C., Canada

## Abstract

Let $P_i$ be a subpolygon of a simple polygon $P$. A vertex of $P_i$ is called *critical* if it is an endpoint of a connected component of $P_i \cap bd(P)$ that has non-zero length. Let $X$ be any subset of $m$ points of the boundary of $P$ containing all reflex vertices of $P$. We give an $O(k^4 m^{3k-3})$ time algorithm that determines whether the boundary of $P$ can be covered by $k$ or fewer convex subpolygons of $P$ whose critical vertices belong to $X$. We then present a characterization of polygons whose boundary admits a convex cover of cardinality three, and conditions under which this cover of the boundary can be extended to a cover of the interior of the polygon. These results together imply the existence of an $O(n^6)$ time algorithm to determine whether a simple polygon admits a convex cover of cardinality three. We then show how to exploit additional properties of polygons that admit such covers to reduce the running time of the algorithm to $O(n \log n)$.

## 1 Introduction

A simple polygon $P$ is said to be *convex* if, for each pair of points $x$, $y$ of $P$, the line segment $[x\ y]$ is contained in $P$. A polygonal chain $C_i$ contained in a simple polygon $P$ will be called *convex* if there exists a convex subpolygon of $P$ of whose boundary it is a subset. The subset of $bd(P)$ that joins a point $p$ of $bd(P)$ to a point $q$ of $bd(P)$ counterclockwise will be written as $P_{p \ldots q}$.

A point $p$ of a simple polygon $P$ is said to be *visible* from a point $p'$ of $P$ if the line segment $[p\ p']$ does not intersect the exterior of $P$. We say that a subset $Q$ of $P$ is *completely visible* from another subset $Q'$ of $P$ if every point of $Q$ is visible from every point of $Q'$, and denote by $V(P, Q)$ the set of points of $P$ from which $Q$ is completely visible. A polygon $P$ is called *star-shaped* if there exists a point $x$ of $P$ such that $P = V(P, \{x\})$. The *kernel* of a star-shaped polygon $P$, denoted by $Kr(P)$, is the set of all points $x$ of $P$ for which $P = V(P, \{x\})$.

Two points $p$, $p'$ of a simple polygon $P$ are *link-k visible* if there exist points $p = p_0, p_1, \ldots, p_{k-1}, p_k = p'$ such that $p_i$ is visible from $p_{i-1}$ for $i = 1$ to $k$. Such a path is said to have *link-length k*. The *link-distance* between $p$ and $p'$ is the minimum link-length of a path between $p$ and $p'$. The *link-diameter* of $P$ is the maximum link-distance between any pair of points of $P$.

A set $C = \{C_1, \ldots, C_k\}$ is said to be a *k-cover* of a simple polygon $P$ if $P = \cup_{i=1}^{k} C_i$. It is called a *convex k-cover* of $P$ if each $C_i$ is convex. A *convex cover* of a subchain $P_{p \ldots q}$ of $bd(P)$ is a set of convex polygonal chains whose union contains $P_{p \ldots q}$.

Let $P_{p \ldots q}$ be a subchain of the boundary of $P$, $C = \{C_1, \ldots, C_k\}$ be a convex $k$-cover of $P_{p \ldots q}$, and $w$ be a point of $P_{p \ldots q}$. Given $\varepsilon > 0$, we shall denote the subset of $C_i$ whose distance to $w$ is less than $\varepsilon$ by $N_\varepsilon(C_i, w)$. Suppose that $w$ is also a vertex of $C_i$. The vertex $w$ will be called *critical* if for each $\varepsilon > 0$, $N_\varepsilon(C_i, w) \setminus \{w\}$ contains both points of $bd(P)$ and points of $int(P)$; $w$ will be called *internal* if there exists $\varepsilon > 0$ such that $N_\varepsilon(C_i, w) \subset bd(P)$.

Computing minimum covers by convex polygons has been proved NP-hard by Culberson and Reckhow [CR88], even when only the boundary of the polygon needs to be covered. Shermer [She93] gave an optimal linear time algorithm that solves the convex 2-cover problem. In this paper, we present an $O(n \log n)$ algorithm to solve the convex 3-cover problem. The basic ideas for our algorithm will come from an algorithm that solves the convex $k$-cover problem for the

[1]Supported by the Natural Sciences and Engineering Research Council of Canada under a Post-Graduate scholarship.

boundary of a simple polygon $P$ in $O(k^4m^{3k-3})$ time, provided that the critical vertices of the covering subpolygons are restricted to belong to a given subset of $bd(P)$ of size $m$.

We develop this more general algorithm in Section 2. In Section 3, we present a characterization of convex 3-covers that shows that, if the boundary of a simple polygon admits a convex 3-cover, then it admits one in which all critical vertices belong to a subset of $bd(P)$ whose size is linear in the number of vertices of $P$. We then apply the result of the preceding section to obtain a polynomial time (but very inefficient) algorithm that solves the convex 3-cover problem. In Section 4, we show how to exploit the specific nature of this particular problem to reduce the running time of the algorithm to $O(n \log n)$. Finally, Section 5 presents the conclusion and some open problems.

## 2 Covering the boundary of simple polygons

Let $P$ be a simple polygon, $k$ be a positive integer, and $X = \{x_1, \ldots, x_m\}$ be a set of points of $bd(P)$ that contains all reflex vertices of $P$. In this section, we present an algorithm to determine whether there exists a convex $k$-cover $\{C_1, \ldots, C_k\}$ of $bd(P)$, such that for $i = 1$ to $k$, each critical vertex of $C_i$ belongs to $X$. This algorithm runs in time polynomial in $m$ for each fixed $k$. We shall assume a real RAM model in which elementary arithmetic operations can be performed in constant time.

A *standard* convex chain is one all of whose vertices are either critical or internal. Let $C = \{C_1, \ldots, C_k\}$ be a convex $k$-cover of a subchain $P_{p \ldots q}$ of the boundary of a simple polygon $P$. The cover $C$ of $P_{p \ldots q}$ will be called *standard* if every chain of $C$ is standard. Since a subchain $P_{p \ldots q}$ of $P$ has a convex $k$-cover if and only if it has a standard convex $k$-cover, we can restrict our attention to standard $k$-covers.

We shall say that a chain $C_i^*$ is the *restriction* of a chain $C_i$ to a subset $P_{p \ldots q}$ of the boundary of $P$ if $C_i^*$ is a connected subset of $C_i$ whose endpoints belong to $P_{p \ldots q}$, and that contains all open intervals of $bd(P)$ that $C_i$ contains. The *composition* of two chains $C_i^l = \{v_1^l, \ldots, v_{n^l}^l\}$ and $C_i^r = \{v_1^r, \ldots, v_{n^r}^r\}$, denoted by $C_i^l \circ C_i^r$, is the chain $\{v_1^l, \ldots, v_{n^l}^l, v_1^r, \ldots, v_{n^r}^r\}$ if $v_{n^l}^l \neq v^r$, and the chain $\{v_1^l, \ldots, v_{n^l}^l, v_2^r, \ldots, v_{n^r}^r\}$ otherwise. An empty chain shall be denoted by the value *nil*.

Let $e_i$, $e_j$ be edges of $P$, and $x$, $y$ be points such that $x \in e_i \backslash \{v_i\}$ and $y \in e_j \backslash \{v_{j+1}\}$. If there exist points $x'$, $y'$ that lie clockwise from $x$ on $e_i$ and counterclockwise from $y$ on $e_j$ respectively, such that the chain $x'xyy'$ is convex (and thus contained in $P$), then we shall say that $x$ sees $y$ convexly, and denote it by $x \sim y$. Suppose that chain $C_i^l = \{v_1^l, \ldots, v_{n^l}^l\}$ is the restriction of a standard convex chain to $P_{p \ldots w}$, and that $C_i^r = \{v_1^r, \ldots, v_{n^r}^r\}$ is the restriction of a standard convex chain to $P_{w \ldots q}$. To determine whether there exists a standard convex chain whose restriction to $P_{p \ldots q}$ is $C_i^l \circ C_i^r$, it suffices to verify that either $C_i^l$ or $C_i^r$ is nil, or that (a) $v_{n^l}^l \sim v_1^r$, and (b) $v_{n^r}^r \sim v_1^l$.

Conditions (a) and (b) can be checked in constant time given $C_i^l$ and $C_i^r$, provided that the vertices of $C_i^l$ and $C_i^r$ belong to a known subset of $bd(P)$, whose visibility graph can be precomputed in time proportional to its size [Her87]. We will say that $C_i^l$ and $C_i^r$ are *compatible*, denoted $C_i^l \diamond C_i^r$, if conditions (a) and (b) hold. We will call two covers $C^l$ and $C^r$ *compatible*, denoted $C^l \diamond C^r$, if $C_i^l \diamond C_i^r$ for $i = 1$ to $k$.

Consider the subchain $P_{p \ldots q}$ of $bd(P)$. We will represent a standard convex $k$-cover $C = \{C_1, \ldots, C_k\}$ of $P_{p \ldots q}$ by a $2k$-tuple $t = (l_1, \ldots, l_k, r_1, \ldots, r_k)$, where $l_i$ ($r_i$) is the clockwisemost (counterclockwisemost) point of $C_i$ on $P_{p \ldots q}$. We will call a $2k$-tuple to which corresponds at least one standard convex $k$-cover of $P_{p \ldots q}$ *admissible* for $P_{p \ldots q}$.

We observe that we can determine whether $C^l \diamond C^r$ from the tuples representing $C^l$ and $C^r$. Thus admissible tuples and standard convex covers are equivalent in this sense, even though one admissible tuple may correspond to an exponential number of different covers. For this reason, we shall apply the notation and terminology developed since the beginning of this section to admissible tuples as well as to covers.

This suggests a divide and conquer approach to the problem of determining the set $T(p, q)$ of all tuples admissible for $P_{p...q}$, whose use we justify by the following lemma.

**Lemma 2.1** *An admissible tuple exists for a chain $P_{p...q}$ if and only if, for each point $w$ of $X \cap P_{p...q}$, there are admissible tuples $t^l$ of $P_{p...w}$, and $t^r$ of $P_{w...q}$ such that $t^l \diamond t^r$.*

We now describe the way in which the two sets $T(p, w)$ and $T(w, q)$ are combined to obtain $T(p, q)$. Let us assume for the purpose of this discussion that both $P_{p...w}$ and $P_{w...q}$ contain $x$ points of $X$. One could combine these sets by examining every pair of tuples $t^l \in T(p, w)$, $t^r \in T(w, q)$, and adding $t^l \circ t^r$ to the output if $t^l \diamond t^r$. This procedure is clearly correct, by Lemma 2.1, but is not very efficient, as it requires $O(k^5 x^{4k-4})$ time. The merging operation can in fact be performed in $O(k^4 x^{3k-3})$ time using a more complicated algorithm, but the details will be omitted here due to lack of space.

Since admissible tuples in $T(p, p)$ correspond to valid convex covers of $bd(P)$, a polygon has a convex $k$-cover all of whose critical vertices belong to $X$ if and only if the set returned by the algorithm is non-empty. However, one cannot effectively recover a convex $k$-cover of $P$ from the admissible tuple corresponding to it. To be able to obtain a convex $k$-cover of $P$ from the corresponding admissible tuple, we need to store, for each tuple $t = t^l \circ t^r$ generated by our algorithm, pointers to the tuples $t^l$ and $t^r$ from which it was obtained.

Thus, each admissible tuple returned by our algorithm becomes the root of a binary tree containing the tuples from which it was obtained. A convex $k$-cover of $P$ can be retrieved by visiting the leaves of this tree using a preorder traversal. We shall refer to this algorithm as algorithm *Boundary-Cover*; this section is summarized in the following theorem.

**Theorem 2.1** *Let $P$ be a simple polygon, $X$ be a subset of $bd(P)$ containing all reflex vertices of $P$ and $k$ be a fixed integer. There is an algorithm that determines whether $bd(P)$ admits a convex $k$-cover all of whose critical vertices belong to $X$ in $O(k^4 m^{3k-3})$ time and $O(k^3 m^{2k-2})$ space, and returns such a cover if one exists.*

# 3 Simple Unions of Three Convex Polygons

In this section, we apply algorithm *Boundary-Cover* to recognize simple polygons that admit a convex cover of cardinality three. We first need to show how to reduce each instance of the convex 3-cover problem for a simple polygon to an instance of the convex 3-cover problem for the boundary of this same polygon. This is accomplished using the next two lemmas.

**Lemma 3.1** *Let $P$ be a simple polygon that admits a convex 3-cover. If $P$ is not star-shaped, then it has link-diameter three.*

**Lemma 3.2** *A simple polygon that is star-shaped or has link-diameter three admits a convex 3-cover if and only if its boundary admits one.*

The proof of Lemma 3.2 in fact gives a linear time algorithm to convert a convex 3-cover of $bd(P)$ into a convex 3-cover of $P$.

## 3.1 Restricting the set of potential vertices

We now proceed to exhibit a set $X$ of points of $bd(P)$ that satisfies the property that, if $bd(P)$ admits a convex 3-cover, then it admits one in which all critical vertices of the covering subpolygons belong to $X$.

A point $p$ of $bd(P)$ will be called *primary* if it is an endpoint of an extension of an edge $e$ of $P$ that is not a vertex incident to that edge; we shall then say that $e$ *generates* $p$.

A *tip* of a simple polygon $P$ is a maximal convex subchain of $bd(P)$. A point $p$ that belongs to a tip $T_p$ of $P$ will be called *secondary* if there exists a primary point $q$ that belongs to a tip $T_q$ of $P$ adjacent to $T_p$, and if a common endpoint of $T_p$ and $T_q$ is collinear with $p$ and $q$, and belongs to $Kr(P)$. Primary and secondary points of $bd(P)$ will be referred to as *potential* points.

We will call the leftmost and rightmost tips of a subchain $P_{p\ldots q}$ of $bd(P)$ *extreme*. We will say that a tip $T$ is *wedged* with respect to a cover $C$ if $T$ contains a vertex of $C$ that is not primary.

We shall say that a chord $c$ of $P$ is *maximal* if $bd(P) \cap int(c)$ is a reflex vertex of $P$. An edge $e$ of a covering polygon $C_i$ that is chord of $P$ will be called *quasi-maximal* if it is maximal, or if

1. $e$ determines exactly two subpolygons of $P$,
2. the subpolygon that does not contain $C_i$ has a single reflex vertex $v$ that belongs to $Kr(P)$, and
3. $e$ intersects both extensions of the edges of $P$ adjacent to $v$.

We will say that the reflex vertex of $P$ that lies in $int(e)$ or belongs to the subpolygon of $P$ that does not contain $C_i$ *supports* $e$. An edge $e$ will be called *out of place* if

1. it is not quasi-maximal,
2. one of its endpoints is not a reflex vertex of $P$, and
3. it does not join a primary point $p$ of $P$ to an endpoint of an edge that generates $p$.

**Lemma 3.3** *If the boundary of a simple polygon $P$ admits a standard convex $k$-cover $C$, then it admits a standard convex $k$-cover with no more critical vertices than $C$ and no out of place edges.*

We will say that a standard convex $k$-cover $C$ of $bd(P)$ is *suitable* if it minimizes the following two parameters among all covers with no out of place edges, in this order :

1. The number of critical vertices of $C$;
2. The number of distinct vertices of polygons of $C$ that are not primary.

Lemma 3.3 ensures the existence of at least one suitable cover of $bd(P)$. The next lemma describes a very important property of suitable covers.

**Lemma 3.4** *A suitable 3-cover $C = \{C_1, C_2, C_3\}$ of the boundary of a simple polygon $P$ does not contain more than two adjacent wedged tips.*

The idea behind the proof of Lemma 3.4 is to first characterize precisely the conditions under which vertices of a suitable cover may fail to be primary, and then to show by contradiction that, if $v$ is a vertex of a subpolygon $C_i$ in a suitable cover, and if $v$ is not primary, then one of the vertices of $C_i$ adjacent to $v$ must be. By applying local modifications based on Lemma 3.4 to a suitable cover of $bd(P)$, we obtain the following theorem.

**Theorem 3.1** *If the boundary of a simple polygon $P$ admits a convex 3-cover, then it admits a standard convex 3-cover in which every critical vertex is a potential point of $bd(P)$.*

## 3.2 Algorithm and running time analysis

We are now ready to describe the algorithm that recognizes polygons admitting a convex 3-cover. It first verifies whether $P$ is star-shaped, or has link-diameter 3. If so, it uses algorithm *Boundary-Cover* to obtain a list of convex 3-covers of $bd(P)$, and returns one after transforming it into a cover of $P$ (provided at least one convex 3-cover of $bd(P)$ exists). Otherwise it returns the empty set. The correctness of the algorithm follows directly from theorems 2.1 and 3.1.

To obtain a bound on the running time of the algorithm, we only need to bound the number of potential points of $bd(P)$. This is done in the next lemma.

**Lemma 3.5** *A simple polygon $P$ with $n$ vertices contains at most $9n - 27$ potential points on its boundary.*

Since each primary point can be obtained from the list of vertices of $P$ using a single ray-shooting query, and since each secondary point can be obtained from the list of primary points of $P$ in the same manner, it follows from a result of Guibas et al. [GHL+87] that all potential points of $bd(P)$ can be generated in $O(n \log n)$ time using $O(n)$ space. Combining this with Theorem 2.1, we conclude that this algorithm runs in $O(n^6)$ time using $O(n^4)$ space.

# 4 Computing Convex 3-Covers Faster

In this section, we improve the running time of the algorithm described in the previous section to $O(n \log n)$, and its space requirements to $O(n)$. This is done by reducing the number of tuples that need to be considered. Conceptually, the reduction is performed in two steps :

- Show that only a constant number of points of $P_{p...q}$ are of interest apart from reflex vertices of $P$;
- Show that the reflex vertices of $P_{p...q}$ can be divided into a constant number of equivalence classes whose elements are interchangeable with respect to compatibility.

Combining these two results then yields the improved algorithm.

Let $p$, $q$ be two points of $bd(P)$, and let $e$ be an edge of $P$ that does not belong to $P_{p...q}$, and whose extension $c$ has an endpoint $w$ in the interior of $P_{p...q}$. We shall say that $e$ is oriented clockwise (counterclockwise) with respect to $P_{p...q}$ if there exists $\varepsilon > 0$ such that all points of $N_\varepsilon(P, w)$ that lie clockwise (counterclockwise) from $w$ on $P_{p...q}$ belong to the interior halfplane determined by $c$. We start by considering visibility among those edges.

**Lemma 4.1** *Let $p$, $q$ be two points of $bd(P)$, and let $e_i$, $e_j$ be two edges of $P$ that do not belong to $P_{p...q}$ and whose extensions have endpoints $w_i$, $w_j$ on $P_{p...q}$. If $e_i$ and $e_j$ have the same orientation, then no interior point of $e_i$ sees any interior point of $e_j$.*

We can use the previous lemma to obtain a bound on the number of primary points of a chain $P_{p...q}$ that may be generated by edges of $P$ not belonging to $P_{p...q}$. This bound is stated in the following lemma.

**Lemma 4.2** *Let $P$ be a simple polygon, and $p$, $q$ be two points of $bd(P)$. If the boundary of $P$ admits a convex $k$-cover, then at most $2k-2$ primary points of $P_{p...q}$ distinct from $p$ and $q$ can be generated by edges of $P$ that do not belong to $P_{p...q}$.*

The chain $P_{p...q}$ will be called *$k$-primal* if it contains at most $k$ primary points generated by edges of $P_{q...p}$, and if $P_{q...p}$ contains at most $k$ primary points generated by edges of $P_{p...q}$.

We can define a labeling function $f$ on the reflex vertices of $P_{p...q}$, whose value depends in part on its position relative to the points mentioned in Lemma 4.2. This labeling defines an equivalence relation on the set of tuples, and has the following important property.

**Lemma 4.3** *Let $P_{p...q}$ and $P_{p'...q'}$ be two subchains of $bd(P)$ whose interiors are disjoint, let $x$ be a reflex vertex of $P_{p...q}$, and let $y$, $z$ be reflex vertices of $P_{p'...q'}$. If $f(y) = f(z)$, then $x \sim y$ if and only if $x \sim z$.*

Under the same conditions, it is also the case that $y \sim x$ if and only if $z \sim x$. Let $f(t)$ denote the tuple obtained from a tuple $t$ by replacing each element $x$ of $t$ by $f(x)$, and let $RT(p, q)$ be the set obtained from $T(p, q)$ by replacing each tuple $t$ of $T(p, q)$ by $f(t)$. We will call such $f(t)$ a *restricted admissible* tuple. We associate with each distinct value $x$ in the range of $f$ an arbitrary reflex vertex $v$ of $P_{p...q}$ for which $f(v) = x$. It follows from the definition of $f$ that the size of $RT(p, q)$ is bounded above by a constant.

We are now ready to describe the manner in which our improved algorithm works. The base case remains the same as that described in Section 2. The merge step however becomes much simpler, since we are only dealing with a constant number of tuples. First we determine whether $P_{p...w}$ and $P_{w...q}$ are 4-primal. It follows from Lemma 4.2 that we can abort immediately if this is not the case. Otherwise we are provided with a constant upper bound on the size of the set of tuples that can be obtained by merging $RT(p, w)$ and $RT(w, q)$. We then label each potential point $x$ of $P_{p...w}$ and $P_{w...q}$ by $f(x)$, compute all admissible tuples that can be obtained by combining a tuple of $RT(p, w)$ with a tuple of $RT(w, q)$, and finally remove duplicates.

We can thus apply the same divide and conquer technique as in Section 2 to compute the set of restricted tuples corresponding to admissible tuples of a subchain $P_{p...q}$ of $bd(P)$. The correctness of this algorithm can be proved by induction.

One can retrieve a cover corresponding to a given restricted admissible tuple $t$ in the same manner as in the case of the algorithm presented in Section 3.2. Since merging two sets of restricted tuples can be done in $O(n)$ time, we get:

**Theorem 4.1** *Given a simple polygon $P$, it is possible to determine whether it admits a convex 3-cover or not, and to return such a cover if it exists, in $O(n \log n)$ time and $O(n)$ space.*

## 5 Conclusion

In this abstract, we described an $O(k^4 m^{3k-3})$ time and $O(k^3 m^{2k-2})$ space algorithm that determines whether the boundary of a polygon $P$ can be covered by $k$ or fewer convex subpolygons of $P$ whose critical vertices belong to a subset $X$ of $bd(P)$ of cardinality $m$. We then characterized polygons whose boundaries admit a convex 3-cover, and presented conditions under which these covers of their boundaries can be extended to covers of their interiors. This allowed us to derive an $O(n \log n)$ time and $O(n)$ space algorithm to recognize polygons that admit a convex 3-cover.

The methods used in Section 2 can be extended to obtain algorithms for several other problems. These problems are those in which only a constant amount of information about covers of two adjacent subchains of $bd(P)$ need to be maintained in order to determine covers of their union. Such problems include covering the boundary of a simple polygon with $r$-spirals, and covering the boundary of orthogonal polygons with rectangles. These algorithms run in time polynomial in $m$ for each fixed value of $k$.

It also follows from the algorithm presented in Section 2 that the style of constructions used in most NP-Hardness results for covering problems [O'R87, CR88] will fail to prove that computing convex $k$-covers of the boundary of simple polygons is NP-Hard for any fixed $k$.

We leave a number of questions unanswered. In particular, in view of the facts presented in the previous paragraph, one might conjecture that the convex $k$-cover problem for the boundary of simple polygons can be solved in polynomial time for each fixed $k$. We do not know if the techniques used in Section 3 can be extended for $k \geq 4$. Determining whether the convex 3-cover problem can be solved in $o(n \log n)$ or has an $\Omega(n \log n)$ lower bound also deserves investigation.

## Acknowledgements

The author wishes to thank Thomas Shermer and David Bremner for their helpful suggestions and comments on this paper and on the problems considered therein.

## References

[CR88] J. Culberson and R. Reckhow. Covering Polygons is NP-Hard. In *Proceedings of the Twenty-Ninth IEEE Annual Symposium on the Foundations of Computer Science*, pages 601–611, 1988.

[GHL+87] L. Guibas, J. Hershberger, D. Leven, M. Sharir, and R. Tarjan. Linear Time Algorithms for Visibility and Shortest Path Problems Inside Triangulated Simple Polygons. *Algorithmica*, 2:209–233, 1987.

[Her87] J. Hershberger. Finding the Visibility Graph of a Polygon in Time Proportional to its Size. In *Proceedings of the Third ACM Symposium on Computational Geometry*, pages 11–20, 1987.

[O'R87] J. O'Rourke. *Art Gallery Theorems and Algorithms.* Oxford University Press, Inc., 1987.

[She93] T. C. Shermer. On Recognizing Unions of Two Convex Polygons and Related Problems. *Pattern Recognition Letters*, to appear.