The art gallery theorem for polyominoes

Therese Biedl* Mohammad T. Irfan† Justin Iwerks‡ Joondong Kim§
Joseph S. B. Mitchell§

April 30, 2012

Abstract

We explore the art gallery problem for the special case that the domain (gallery) P is an m-*polyomino*, a polyform whose cells are m unit squares. We study the combinatorics of guarding polyominoes in terms of the parameter m, in contrast with the traditional parameter n, the number of vertices of P. In particular, we show that $\lfloor \frac{m+1}{3} \rfloor$ point guards are always sufficient and sometimes necessary to cover an m-polyomino, possibly with holes. When $m \leq \frac{3n}{4} - 4$, the sufficiency condition yields a strictly lower guard number than $\lfloor \frac{n}{4} \rfloor$, given by the art gallery theorem for orthogonal polygons.

Keywords: Art gallery theorem, Polyomino, Visibility coverage, Guard number

1 Introduction

Victor Klee (1973) posed the problem of determining the minimum number of point guards sufficient to cover the interior of an art gallery modeled as a simple polygon P with n vertices. The solution, first given by Vasek Chvátal, is that $\lfloor \frac{n}{3} \rfloor$ guards are sometimes necessary and always sufficient to cover a polygon possessing n vertices [15]. This original problem has since grown into a significant area of study in computational geometry and computer science. Art gallery problems are of theoretical interest but also play a central role in visibility problems arising in applications in robotics, digital model capture, sensor networks, motion planning, vision, and computer-aided design [18, 19].

We explore the art gallery problem when the given gallery P is an *m-polyomino*, a polyform whose cells are integral unit squares. (In other words, an m-polyomino P is the union of m (closed) integral unit squares such that the interior of P is connected.) We refer to the unit squares as *pixels*. An example with $m = 29$ is shown in Figure 1. We often write (m) for an m-polyomino. For example, (7) refers to any 7-polyomino. A polyomino P is *simple* if it has no *holes*, i.e., the interior of the complement of P is connected. The *dual graph* of an m-polyomino P has a node for each pixel of P and an edge joining two nodes that correspond to edge-adjacent pixels. We will frequently use the same name (such as v or p) for a node in this dual graph and the pixel in the polyomino that it represents.

A point $a \in P$ *covers* (or *sees*) a point $b \in P$ if the line segment ab is a subset of P. (Since P is closed, possibly ab contains points on the boundary, ∂P.)

*David R. Cheriton School of Computer Science, University of Waterloo, biedl@cs.uwaterloo.ca
†Department of Computer Science, Stony Brook University, mtirfan@cs.stonybrook.edu
‡Department of Applied Mathematics and Statistics, Stony Brook University, {jiwerks,jdkim,jsbm}@ams.stonybrook.edu

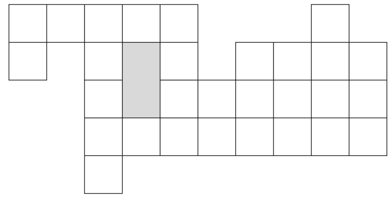

Figure 1: A polyomino with 29 pixels. The shaded region is a hole

We let $g(P)$ denote the *guard number* of P: $g(P)$ is the minimum number of guards, modeled as points, required to cover all of a given polyomino P. Then we define $G(m)$ to be the maximum value of $g(P)$ over all m-polyominoes P. Our main combinatorial question is that of determining upper and lower bounds on $G(m)$.

There are possible alternative models of visibility that one can study. One such model is *r-visibility*, where two points can see each other if and only if the axis-parallel rectangle defined by them is a subset of P. Another model, the *all-or-nothing* model, considers a pixel p to be guarded only if a single guard a sees all points of p (i.e., $ab \subset P$, for all $b \in p$). Figure 2 illustrates these models. It is easy to see that a cover in the r-visibility model is a cover in the all-or-nothing model, which in turn is a cover in the unrestricted model, and none of these implications hold in reverse in general.

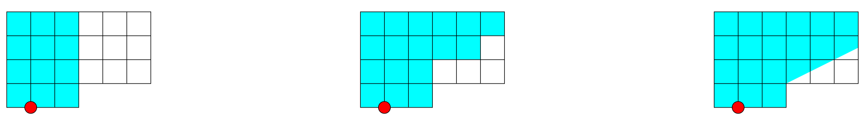

Figure 2: The three models of visibility. From left to right: r-visibility model, all-or-nothing model and unrestricted model

Previous work. While $\lfloor \frac{n}{3} \rfloor$ guards are sufficient and sometimes necessary to cover a simple polygon with n vertices, $\lfloor \frac{n}{4} \rfloor$ guards are sufficient and sometimes necessary to cover an orthogonal polygon [8, 9, 10, 14]. It is NP-hard to find the minimum number of guards for covering either a general simple polygon [12] or an orthogonal polygon [17]. Even covering the vertices of an orthogonal polygon has been shown to be NP-hard [11].

While no approximation algorithm with better than the naive factor $\lfloor \frac{n}{3} \rfloor$ is known for placing the fewest guards at arbitrary points in a simple polygon, there are results for approximation of optimal guard placement in polygons by restricting the set of candidate guards. For instance, if we restrict guards to lie only on vertices of a polygon or grid points, logarithmic approximations are achievable based on set cover [5, 6, 7].

In orthogonal polygons, Nilsson [13] gives an algorithm to compute $O(OPT^2)$ guards, based on the constant-factor approximation for guarding 1.5D terrains [2]. Worman and Keil [20] show that a minimum guard cover in the r-visibility model for orthogonal polygons can be found in polynomial time. The algorithm runs in time $\tilde{O}(n^{17})$ and requires computing a maximum independent set in a perfect graph as a subroutine.

Practical methods and heuristics for placing guards in general polygonal domains have been investigated by Amit et al. [1].

2 Our Results[1]

We show that $\lfloor \frac{m+1}{3} \rfloor$ point guards are sometimes necessary and always sufficient to cover an m-polyomino (possibly with holes). The necessity bound is in the unrestricted visibility model (and hence also holds in the all-or-nothing and the r-visibility model). The sufficiency bound is constructive and yields a guard set that is in the (most restrictive) r-visibility model. We demonstrate that these tight combinatorial bounds carry over to *rectanglominoes*, a generalization of polyominoes in which the cells are edge-aligned rectangles.

2.1 Necessary Condition

Figure 3 shows a polyomino analog of Chvátal's comb, illustrating that, for $m \geq 2$, $\lfloor \frac{m+1}{3} \rfloor$ guards are sometimes necessary to cover an m-polyomino.

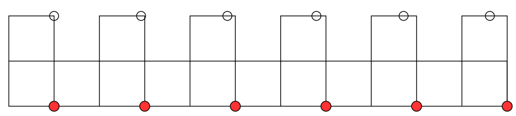

Figure 3: An example demonstrating that $\lfloor \frac{m+1}{3} \rfloor$ guards (depicted as filled dots) are sometimes necessary to guard P, since no two of the open circles can be covered by one guard

2.2 Sufficiency Condition

Given an m-polyomino P (possibly with holes) where $m \geq 2$, we show that $\lfloor \frac{m+1}{3} \rfloor$ guards are always sufficient to cover P. The proof of this sufficiency condition is obtained by arguing that we can iteratively remove certain types of *subpolyominoes* while leaving the remaining polyomino connected. The removal process will involve creating a BFS tree of the dual graph of P, and then iteratively clipping off small subtrees. First, we introduce a lemma to be employed later:

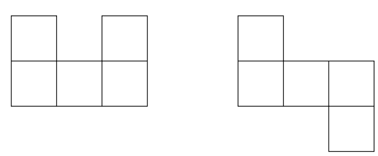

Figure 4: Special 5-polyominoes: $5'$ (left) and $5''$ (right). (The labels apply to all polyominoes that are one of the above after rotation and/or reflection.) Each requires two guards

Lemma 1. *For $1 \leq m \leq 4$, any m-polyomino can be covered with one guard.*
Any 5-polyomino that is not $5'$ or $5''$ (see Figure 4) can be covered with one guard.
For $m = 6, 7$, any m-polyomino can be covered with two guards.
In all cases, the polyominoes are covered even in the r-visibility model.

[1]In a related paper appearing in SoCG 2011, we also gave tight combinatorial bounds for guarding polyominoes with *pixel guards* (guards that are themselves pixels of the polyomino), established that guarding a simple polyomino with point or pixel guards is NP-hard, and presented various optimal algorithms for special cases [3]. Additional results and improvements to these algorithms have since been obtained and will appear together in a separate paper. Also see [16] regarding the combinatorics of guarding polyominoes with pixel guards.

Proof. One could prove this simply by inspection of the finite (though large) number of such polyominoes, but instead we give a proof based on locating the best places for guards using a spanning tree in the dual graph.

We do not prove each time that all of our guards are at pixel corners and that r-visibility suffices for the covering; this should be obvious from the constructions.

The crucial insight is that if a guard v is at the corner of some pixel p, then it guards not only p, but also all neighbors of p, even in the r-visibility model.

Lemma 2. *Any m-polyomino P, $m = 1, 2, 3, 4$, can be covered with one guard.*

Proof. If P is a rectangle, then any pixel corner will do. Otherwise P has a reflex corner. Placing a guard v at this corner means that v belongs to three pixels and covers the fourth (if any) since it is adjacent to one of the three. See Figure 5(a). □

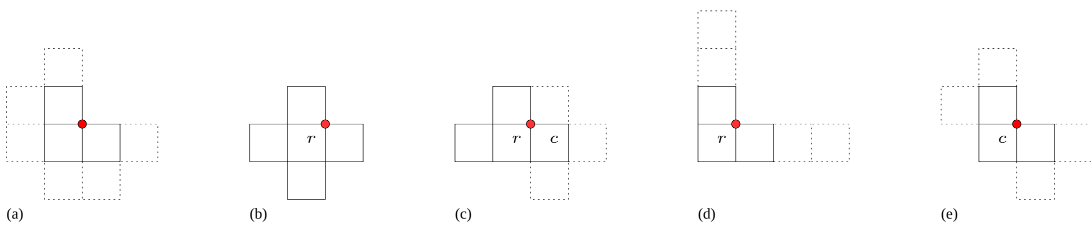

Figure 5: Cases for covering polyominoes with up to 5 pixels

Lemma 3. *Any 5-polyomino P that is not $5'$ or $5''$ can be covered with one guard.*

Proof. Compute a BFS tree T of the dual graph of P, rooted at a node r of maximum degree. If r has degree 4, then any other pixel is a neighbor of r, so a guard on one corner of r will do (Figure 5(b)). If r has degree 3, then exactly one neighbor c of r has exactly one other neighbor (in T), and placing a guard at a common point of r and c will cover everything (Figure 5(c)).

Now assume that r has degree 2, hence the dual of P is a path. If P is a rectangle, then placing a guard at any pixel corner will do. If P has exactly one reflex corner, then placing a guard on it will cover everything (Figure 5(d)). If it has multiple reflex corners, then let c be the middle node of this path. If the neighbors of c are not on opposite sides of c, then the point common to the neighbors covers everything (Figure 5(e)). This finally leaves the case where c and its two neighbors form a rectangle, but P has two reflex corners. This implies that P is $5'$ or $5''$. □

Lemma 4. *Every (6) and (7) can be covered with two guards.*

Proof. We first prove this for a 7-polyomino P. Compute a BFS tree T of the dual graph of P, starting at a node r of maximum degree. If any child c of r has a subtree T_c of size 3 or 4, then T_c and $T - T_c$ form two connected subpolyominoes that can be covered with one guard each, and we are done. If all children of r have subtrees of size 2 or 1, then place two guards at two diagonally opposite corners of r; these guards are then in all children of r and hence cover all grandchildren as well, and hence all of T.

So finally presume some child of r has a subtree of size 5 or 6. Then r can have at most two children, so the dual of P is a path. But then P can easily be split into a (3) and a (4) and hence be covered with two guards.

This proves the claim for a (7). For a 6-polyomino P, let q be a bottommost pixel, i.e., a pixel with smallest y-coordinate, breaking ties arbitrarily. Attach an extra pixel p below q to create a 7-polyomino P'. Then cover P' with two guards. If either one of them is in a bottom

Figure 6: If the polyomino shown has a BFS tree rooted at p_2, v' cannot exist since otherwise c_1 would have been made a child of v', not v

corner of p, then we can move it to the top corner of p without decreasing coverage, since p is a leaf in the dual graph. Hence the two guards will also cover P. □

Hence, Lemmas 2-4 establish the validity of Lemma 1. □

In the proof of the subsequent theorem, a BFS tree structure with an important construction property is utilized. Let T be a BFS tree of the dual graph of a polyomino P, rooted at a node r that has degree at most 2. Add the neighbors of r to the queue, with r as parent. As long as the queue is not empty, remove the next node v from it. Then, add all unvisited neighbors w of v to a queue in the following fashion: Assume that v has parent p_1. Then the neighbor w that is on the opposite side of v from p_1 is added *last* to the queue (if it exists and was not already visited). In other words, the BFS will always give preference to "making a turn" when exploring the dual graph. Because of this, we have the following observation, which will be crucial later.

Lemma 5. *Let v be a pixel that has a grandparent in this BFS tree T. If v has two children c_1, c_2 in T that are on opposite sides of v, then v has no sibling.*

Proof. Let p_1 and p_2 be the parent and grandparent of v, respectively. Then p_2 must be adjacent to p_1, but it cannot be adjacent to either c_1 or c_2, since this would violate the BFS property. Since c_1 and c_2 are on opposite sides of v, this implies that p_2 and v are on on opposite sides of p_1. Using our BFS convention, v is added last to the queue when exploring from p_1. If p_1 had any other child v', then v' would be adjacent to c_1 or c_2, and would have been added to the queue before v. So c_1 or c_2 would have been made a child of v', not v. Therefore, v has no siblings (see Figure 6). □

Theorem 1. *For an m-polyomino P_0 (possibly with holes) there are polyominoes $P_1, P_2, \ldots, P_f$ with the following properties:*

(a) P_i is a connected subpolyomino of P_{i-1} $(1 \leq i \leq f)$.
(b) Subpolyomino S_i, the difference between P_{i-1} and P_i is in the set Good Polyominoes, $GP = \{(3), (4), ((5)\backslash\{5', 5''\}), (6), (7)\}$.
(c) P_f has 0, 1, or 2 pixels.

Proof. Let T be a BFS tree of the dual graph of P_0 obtained as explained above. Every node in T has at most 3 children. If the height of T is less than 2, then T has at most 3 nodes since T is rooted at a node of degree at most 2. If T has exactly 3 nodes, then set $S_f = T$ (i.e., let S_f be the subpolyomino whose dual graph is T), set $P_f = \emptyset$ and we are done. If $|T| < 3$, set $P_f = T$ and again we are done.

Suppose now that the height of T is at least 2, and let q be a lowest leaf of T. If q has siblings, then let T_{p_1} be the subtree rooted at the parent p_1 of q. See Figure 7(a). T_{p_1} then has

3 or 4 nodes. Set $S_1 = T_{p_1}$, satisfying (b). Hence, $P_1 = P_0 - S_1$ is $T - T_{p_1}$; this is connected (so satisfies (a)) since T_{p_1} is a rooted subtree of T. By induction we can split P_1 as desired.

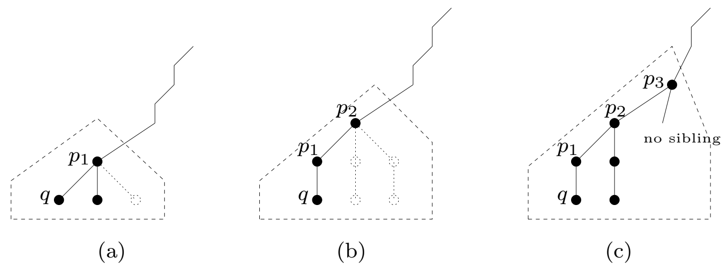

Figure 7: Finding a subtree that forms a Good Polyomino

If none of the lowest leaves of T has siblings, let q be a lowest leaf and p_1, p_2 be its parent and grandparent, respectively. Let T_{p_2} be the subtree rooted at p_2. See Figure 7(b). T_{p_2} has between 3 and 7 nodes. The minimum of 3 occurs when p_2 has no other children and the maximum of 7 occurs when p_2 has 3 children, each of whom has one child (since no leaf has siblings). If T_{p_2} is not $5'$ or $5''$, then set $S_1 = T_{p_2}$. As before, $S_1 \in GP$, satisfying (b) and since $T - T_{p_2}$ is connected, (a) holds as well and we are done by induction.

If T_{p_2} is $5'$ or $5''$, then p_2 must be the middle pixel, and we do one of the following:

(1) If p_2 has no parent, then the whole polyomino P_0 is a $5'$ or $5''$, and hence can be split into a (3) and a (2).

(2) If p_2 has a grandparent, then p_2 has no sibling since it is the middle node of a $5'$ or $5''$ and hence has children on opposite sides (Lemma 5). Therefore, the subtree T_{p_3} rooted at the parent p_3 of p_2 is a (6). See Figure 7(c). Set $S_1 = T_{p_3}$ and iterate as before.

(3) Finally presume p_2 has a parent p_3, but no grandparent. So p_3 is the root and has degree ≤ 2. Let T' be the (6) formed by T_{p_2} together with p_3; set $S_1 = T'$ as before, then $P_1 = P_0 - S_1$ is again a connected polyomino and we are done by induction.

□

Corollary 1. *For $m \geq 2$, $\lfloor \frac{m+1}{3} \rfloor$ guards are sometimes necessary and always sufficient to cover a connected m-polyomino P (possibly with holes), even in the r-visibility model.*

Proof. Theorem 1 shows that we can partition P into subpolyominoes $S_1, \ldots, S_f$ and P_f such that each S_i is covered by $\lfloor \frac{|S_i|}{3} \rfloor$ guards, and P_f has 0, 1 or 2 pixels. We use $1 = \lfloor \frac{|P_f|+2}{3} \rfloor$ guard for P_f if it is non-empty. These guards cover the polyomino even in the r-visibility model by Lemma 1. Hence, the number of guards is $\sum_{i=1}^{f} \lfloor \frac{|S_i|}{3} \rfloor + \lfloor \frac{|P_f|+2}{3} \rfloor \leq \frac{m+2}{3}$. Since the number of guards is an integer, this gives an $\lfloor \frac{m+2}{3} \rfloor$ sufficiency condition, but we can make a slight improvement to $\lfloor \frac{m+1}{3} \rfloor$.

If $m = 2 + 3k$ or $m = 3 + 3k$ with $k \in \mathbb{N}$, then $\lfloor \frac{m+1}{3} \rfloor$ and $\lfloor \frac{m+2}{3} \rfloor$ are equivalent. So assume $m = 1 + 3k$.

Suppose first that some S_j is not a (3) or a (6), and hence uses actually only $\lfloor \frac{|S_j|-1}{3} \rfloor$ guards. Redoing the above equation then shows that the number of guards is at most $\frac{m+1}{3}$ (and as before, therefore not more than $\lfloor \frac{m+1}{3} \rfloor$ since it is an integer.) On the other hand, if each S_i is a (3) or a (6), and $m = 1 + 3k$, then P_f consists of exactly one pixel. Hence, $S_f \cup P_f$ is a (4) or a (7), and can be covered with $\lfloor \frac{1}{3}(|S_f| + |P_f| - 1) \rfloor$ guards. Again redoing the equation shows that the number of guards is at most $\lfloor \frac{m+1}{3} \rfloor$.

The construction in Section 2.1 establishes the matching lower bound. □

Note that our proof is constructive and gives an algorithm to find a set of $\lfloor \frac{m+1}{3} \rfloor$ guards. The time complexity of this algorithm is dominated by the time to find the decomposition of P into good subpolyominoes. To see that this can be done in overall linear time, observe that there is no need to recompute the BFS tree every time: the remainder of the BFS tree is a BFS tree for the remaining polyomino. Thus we compute the BFS tree only once, and enumerate the nodes in it in backward level order. For each node q, in this order, we then find a good subpolyomino in the vicinity of q: it is either at q's parent, grandparent, or the great-grandparent, or at one of their children or grandchildren. Thus we only need to check a constant number of subtrees, all within constant distance of q. So finding a good subpolyomino takes constant time per removed subpolyomino.

2.3 Generalization to Rectanglominoes

We observe that the combinatorial results described above extend to a generalization of polyominoes, which we will call *rectanglominoes*: connected unions of m edge aligned rectangular pixels. If pixels p_i and p_j, with heights h_i and h_j and lengths l_i and l_j respectively, are horizontally adjacent in a rectanglomino R, then $h_i = h_j$. Similarly, if p_i and p_j are vertically adjacent, then $l_i = l_j$. Any rectanglomino R has an *associated* polyomino P_R that is obtained by setting all pixel heights and lengths of R to unit length. It is possible that P_R will self-overlap, but the combinatorial results obtained for polyominoes apply to self-overlapping polyominoes as well. Clearly, the dual graphs of R and P_R are equivalent since the number of pixels in R and P_R is the same and pixel adjacencies are preserved.

Despite having identical dual graphs, a rectanglomino and its associated polyomino do not always have the same guard number. Consider the example in Figure 8 where the rectanglomino on the left requires three guards while the associated polyomino on the right needs only two.

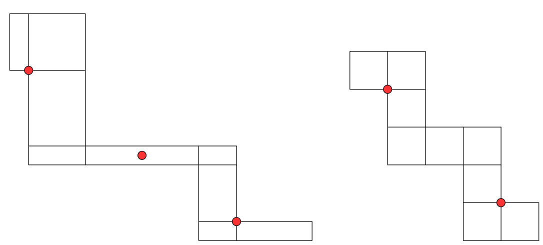

Figure 8: The rectanglomino (left) requires 3 guards, while its associated polyomino (right) requires only 2 guards

However, the two numbers are the same in the r-visibility model:

Lemma 6. *A rectanglomino R can be covered with k guards in the r-visibility model if and only if its associated polyomino P_R (possibly self-overlapping) can be covered with k guards in the r-visibility model.*

Proof. We only show one direction; the other one is similar. Assume we have a cover of P_R. Map each point p_r in P_R to a point r in R in the natural way: if p_r is at a corner, then it is mapped to the corresponding corner, and if it is not on a corner, then it is mapped to the linear interpolation between the corners of the pixel that contain it. Using this mapping on the set of guards of P_R gives a set of points in R, and we must now argue that this is a cover of R.

Since adjacent rectangles in a rectanglomino are edge-aligned, this transformation maps a rectangle in P_R to a rectangle in R. Hence, if point p_R in P_R is guarded in the r-visibility model by guard v_R, then the rectangle $\mathcal{R}$ spanned by them is inside P_R. Applying the transformation yields a rectangle inside R containing the images of p_R and v_R; hence, any point in R is covered by some of the chosen guards. □

As a consequence, all of our upper bounds on the guard number (Corollary 1) immediately transfer to rectanglominoes, since our covers were valid even in the r-visibility model. Of course the lower bounds transfer as well, since a polyomino is a special kind of rectanglomino. Hence, we have:

Corollary 2. $\lfloor \frac{m+1}{3} \rfloor$ *guards are sometimes necessary and always sufficient to cover a connected* m*-rectanglomino (possibly with holes), even in the* r*-visibility model.*

3 Conclusion

We have explored a variation of the art gallery problem set in a polyomino domain, in which the input parameter, m, denotes the number of pixels found in the polyomino as opposed to the usual parameter, n, the number of vertices of the polygon. It was shown that $\lfloor \frac{m+1}{3} \rfloor$ guards are sometimes necessary and always sufficient to cover a polyomino on m pixels. These tight combinatorial bounds also extend to rectanglominoes.

It should be emphasized that these conditions apply to polyominoes with or without holes. For general polygons, $\lfloor \frac{n+h}{3} \rfloor$ point guards are sometimes necessary and always sufficient where h is the number of holes [4]. When $m \leq \frac{3n}{4} - 4$ we have $\lfloor \frac{m+1}{3} \rfloor < \lfloor \frac{n}{4} \rfloor$, yielding a strictly lower sufficiency bound than obtained by the art gallery theorem for orthogonal polygons.

It remains open as to whether there is any approximation algorithm for guarding simple polyominoes.

Acknowledgments. This work was partially supported by NSF (CCF-0729019, CCF-1018388) and NSERC. We thank the reviewers for many helpful suggestions that improved the paper.

References

[1] Y. Amit, J. S. B. Mitchell, and E. Packer. Locating guards for visibility coverage of polygons. *International Journal of Computational Geometry and Applications*, 20(5):601–630, 2010.

[2] B. Ben-Moshe, M. J. Katz, and J. S. B. Mitchell. A constant-factor approximation algorithm for optimal terrain guarding. *SIAM Journal on Computing*, 36:1631–1647, 2007.

[3] T. Biedl, M. T. Irfan, J. Iwerks, J. Kim, and J. S. B. Mitchell. Guarding polyominoes. *Proc. 27th Annual Symposium on Computational Geometry (SoCG 2011)*, pp. 387-396, 2011.

[4] I. Bjorling-Sachs and D. Souvaine. An efficient algorithm for placing guards in polygons with holes. *Discrete and Computational Geometry*, 13:77–109, 1995.

[5] A. Efrat and S. Har-Peled. Guarding galleries and terrains. *Information Processing Letters*, 100(6):238–245, 2006.

[6] S. Ghosh. Approximation algorithms for art gallery problems. In *Proc. of Canadian Information Processing Society Congress*, pages 429–434, 1987.

[7] S. Ghosh. Approximation algorithms for art gallery problems in polygons. *Discrete Applied Mathematics*, 158(6):718-722, 2010.

[8] E. Györi. A short proof of the rectilinear art gallery theorem. *SIAM Journal on Algebraic and Discrete Methods*, 7(3):452–454, 1986.

[9] F. Hoffman. On the rectilinear art gallery problem. In *Proc. 17th International Colloquium on Automata, Languages and Programming*, vol. 443 *Lecture Notes in Computer Science*, pages 717–728. Springer, 1990.

[10] J. Kahn, M. Klawe, and D. Kleitman. Traditional galleries require fewer watchmen. *SIAM Journal on Algebraic and Discrete Methods*, 4(2):194–206, 1983.

[11] M. J. Katz and G. S. Roisman. On guarding the vertices of rectilinear domains. *Computational Geometry: Theory and Applications*, 39(3):219–228, 2008.

[12] D. T. Lee and A. K. Lin. Computational complexity of art gallery problems. *IEEE Transactions on Information Theory*, 32(2):276–282, 1986.

[13] B. J. Nilsson. Approximate guarding of monotone and rectilinear polygons. In *Proc. 32nd International Colloquium on Automata, Languages and Programming*, vol. 3580 *Lecture Notes in Computer Science*, pages 1362–1373. Springer, 2005.

[14] J. O'Rourke. An alternate proof of the rectilinear art gallery theorem. *Journal of Geometry*, 21(1):118–130, Dec. 1983.

[15] J. O'Rourke. *Art gallery theorems and algorithms.* Oxford University Press, Inc., 1987.

[16] V. Pinciu. Pixel guards in polyominoes. *Proc. of the Cologne-Twente Workshop on Graphs and Combinatorial Optimizations (CTW 2010)*, pages 137–140, 2010.

[17] D. Schuchardt and H. Hecker. Two NP-hard art-gallery problems for ortho-polygons. *Mathematical Logic Quarterly*, 41(2):261–267, 1995.

[18] T. Shermer. Recent results in art galleries. *Proc. of the IEEE*, 80(9):1384–1399, 1992.

[19] J. Urrutia. Art gallery and illumination problems. In: J.R. Sack, J Urrutia (Eds.), *Handbook of Computational Geometry*, Elsevier, pages 973–1027, 2000.

[20] C. Worman and J. M. Keil. Polygon decomposition and the orthogonal art gallery problem. *International Journal of Computational Geometry and Applications*, 17(2):105–138, 2007.