ディジタル星型領域とその応用

全 眞嬉[1], コルマン マティアス[1], ノレンバグ マティン[2], 徳山 豪[1]
[1] 東北大学 大学院情報科学研究科 (jinhee,mati,tokuyama)@dais.is.tohoku.ac.jp
[2] Faculty of Computer Science, Karlsruhe University, Germany. noelle@iti.uka.de

概要

平面グリッド上の中心点 o が 与えられた時，o を通るディジタル線分と o が中心点となる任意の $p \in P$ がディジタル線分 dig(po) を含むディジタル星型図形 P について定義を行う．すべての $p \in P$ 上の dig(po) の結合はグリッド上で木の形をし，ユークリディアン線分 $\overline{po}$ と dig(po) との Hausdorff 距離は十分に小さい．高次元の場合も考慮する．最適星型図形と星型環形の抽出をイメージセグメンテーション問題に用いることによって有効なアルゴリズム設計が可能となる．さらに，山のような形をした図形の水平断面が星型領域である図形の最適近似を行うことができる．NP 困難性の結果を用いて２つのディジタル星型の結合を抽出することの難しいさについて述べる．

Digital Star Shapes and Their Applications

Jinhee Chun[1], Matias Korman[1], Martin Nöllenburg[2], Takeshi Tokuyama[1]
[1] GSIS, Tohoku University, Sendai, Japan. (jinhee,mati,tokuyama)@dais.is.tohoku.ac.jp
[2] Faculty of Computer Science, Karlsruhe University, Germany. noelle@iti.uka.de

abstract

Given a fixed center point o in a planar grid, we give a novel definition of a *digital line segment* through o and a *digital star-shaped region* centered at o, such that any digital star-shaped region P contains the digital line segment dig(po) for any $p \in P$. The union of dig(po) over all $p \in P$ forms a tree in the grid, and the Hausdorff distance between the Euclidean line segment $\overline{po}$ and dig(po) is optimally small. We also give the higher-dimensional analogue. This enables us to design efficient algorithms for the image segmentation problems that extract optimized star-shaped regions and star-shaped annuli. Moreover, we can optimally approximate a terrain by a mountain-like terrain whose horizontal slices are star-shaped regions. We also show an NP-hardness result to imply that it is difficult to extract the union of two digital star shapes.

1 Introduction

In Euclidean geometry, a *star-shaped region* (or polygon) P centered at a point o is characterized as a region (polygon) such that the line segment $\overline{po}$ for any $p \in P$ must be contained in P. The set difference of two star-shaped regions with a shared center o is called a *star-shaped annulus.* We would like to consider star shapes in a grid $\mathbf{G}$, such that a digital region, i.e., a set of grid points, P is a *digital star-shaped region* centered at a grid point o if the *digital line segment* dig(po) from p to o is contained in P for any $p \in P$. Our theory has a natural generalization to higher dimensions. Nonetheless, for better readability, we describe our results mainly for the two-dimensional case where $\mathbf{G}$ is the $n \times n$ orthogonal grid.

The above definition of digital star shapes relies on the definition of digital line segments. A natural definition of a digital line segment with slope between -1 and 1 (other slopes are handled similarly) consists of the nearest grid point to the line in each column. But this definition (and its variant) makes digital star shapes quite ugly, and also causes inconsistencies if we additionally demand that digital line segments themselves are star shaped.

Instead, we give a spanning tree T of the grid rooted at o such that the unique path from p to o in the tree T defines the digital line segment dig(po) that approximates the Euclidean line segment $\overline{po}$ for each grid point p. We show that the Hausdorff distance from dig(po) to $\overline{po}$ is bounded by $(\log n)/\sqrt{2}$ in the $n \times n$ pixel grid. We also give a matching $\Omega(\log n)$ lower bound by using discrepancy theory. Here, the Hausdorff distance $H(A, B)$ of two planar objects A and B is defined by $H(A, B) = \max\{h(A, B), h(B, A)\}$, where $h(A, B) = \max_{a \in A} \min_{b \in B} d(a, b)$ and $d(a, b)$ is the Euclidean distance between the points a and b.

We may describe the property of T by using the following random walk of a robot in the grid. The robot starts at the origin, and walks in the first quadrant of the grid. If it is located at a point (i, j), then, it can move to it adjacent grid such that the L_1 distance from the origin increases by 1. The robot has a random coin (i.e. generator of a random bit) to determine which way it should take next. If the robot may move freely using all the edges of the grid, its trajectory may give any x-y monotone path, and the probability of the reached vertex obeys a normal distribution. Instead, if we restrict the robot to move on T, and use the random coin only at branches of T, then the robot goes almost straight whatever the values of the random coin, and can reach every (i, j). In our tree T, the probability to reach (i, j) after $k = i + j$ steps is almost uniformly distributed. Thus, the random walk on T simulates brink of a star in which each light lay goes straight and the wave front form squares.

We consider the path in T from p to o as the digital line segment $\mathrm{dig}(po)$, and thus the set of digital star-shaped regions centered at o is the set of rooted subtrees of T. This gives a nice approximation method of star-shaped regions in Euclidian space, in such a way that for any star-shaped region P in Euclidean space, we can find a digital star-shaped region P' such that $H(P, P') = O(\log n)$.

Motivations and applications. Extracting a geometric object in a digital image is an important computational problem and has several applications in vision analysis (image segmentation), pattern matching, medical science, and robotics. Moreover, it is utilized in unexpected applications such as data mining [1]. We focus on the problem where the input is an $n \times n$ pixel grid $\mathbf{P}$ where each pixel p has a non-negative real value $f(p)$ called *brightness*. We want to find a function ϕ such that ϕ mathematically indicates the extracted object satisfying a given geometric or combinatorial condition, and such that the L_2 distance $|f - \phi|_2 = [\sum_{p \in \mathbf{P}}(f(p) - \phi(p))^2]^{1/2}$ is minimized.

The most fundamental case is the *segmentation problem* where ϕ is a constant function for an object $O \in \mathcal{O}$ and its complement (background) $B = \mathbf{P} \setminus O$, where $\mathcal{O}$ is the family of regions associated to the objects considered in the application. It is easy to the see that ϕ should take the respective average brightness value on each of O and B. Thus, the objective function is equivalent to the one often called *intra-class variance*. The computational complexity of the segmentation problem depends on the family $\mathcal{O}$ of regions. It is known that the problem is NP-hard if $\mathcal{O}$ is the set of all connected regions in $\mathbf{P}$. On the other hand, it is soluble in polynomial time in terms of the grid size n for the families of y-monotone regions, rectilinear convex regions, regions below a digital curve with smoothness condition, regions bounded by two digital curves with smoothness condition, and so on [1, 11, 12]. Thus, the computational complexity highly depends on the choice of a family of regions.

Another related problem is the *pyramid approximation* [2, 3], where ϕ is a function such that the level set $P(\phi, t) = \{p \in \mathbf{P} \mid \phi(p) \geq t\}$ is contained in the family $\mathcal{O}$ for any real number t. If we consider the trajectory of ϕ as a terrain, $P(\phi, t)$ is the horizontal slice of the terrain at height t. Thus, intuitively, ϕ extracts a pyramid obtained by piling up regions (or slabs with shapes of regions) in $\mathcal{O}$ which optimally approximate the terrain defined by the input function f. See Figure 1 for an example of pyramid approximation.

In usual Euclidean geometry, the family of star-shaped regions sharing a center point o is a popular and useful family of regions. We denote a function ϕ whose level sets are star-shaped regions as a *mountain*. In particular, the level sets (i.e., horizontal slices) of some real mountains, e.g., Mount Fuji in Japan, are most naturally described by using star shapes. However, the algorithms for optimized segmentation and pyramid approximation highly rely on the pixel grid structure, and their complexity is measured by the grid size. Thus, the family of star-shaped regions was not easy to handle. Indeed, we need to give a concrete definition of digital star shapes before considering optimization.

In a previous work, Wu [11] demonstrates that the optimal image segmentation problem using star-shaped annuli is important in medical application, since these shapes can model the outline of a tumor in a medical image. In [11], it is assumed that the data itself has already been transformed by using a central projection from the center point o such that the family of star-shaped regions corresponds to the family of regions below digital curves. Unfortunately, this transformation destroys the grid structure and it is difficult to transform the star shapes back into the grid while keeping optimality—at least if the input was given in the grid. Therefore, we would like to propose a method to deal directly with digital star-shaped regions and consider the optimized image segmentation problem and its variants in the digital plane.

We can identify $\mathbf{P}$ and the grid $\mathbf{G}$, and consider the problem in $\mathbf{G}$, where the brightness level of each pixel becomes weight of the corresponding vertex (i.e., grid point) in $\mathbf{G}$. As we show in this paper, a naive definition of star-shaped regions would not give a nice family of regions. For example, an important property of star-shaped regions with center o is that the union and intersection of two such regions are also star shaped. We would like to inherit this property to our digital star-shaped regions, so we need a careful definition not to destroy it. To the authors' knowledge, such a definition was not given in the literature.

Algorithms based on digital star-shaped regions. Most of the polynomial-time solvable cases for the optimized region segmentation problem are reduced to the maximum domination closure problem for a directed acyclic graph (DAG) $D = (V, A)$ with vertex weights [2, 11]. A *domination closure* is defined as a subset $W \subseteq V$ such that for each $v \in W$ all its descendents along directed paths are also contained in W. The weight of a domination closure is the sum of the weights of its elements. A *maximum* domination closure is a domination closure whose weight is maximum. In particular, if a family of regions is closed under the operation of taking union and intersection of regions, we can find a graph such that the family is expressed by the set of domination closures of the graph. By using our definition, the family of digital star-shaped regions corresponds to the family of domination closures (indeed, rooted subtrees) of the rooted tree T, in which each edge is directed towards the root o. Thus, plugging in the methodology by Chen *et al.* [2], the optimal mountain can be computed in $O(\min(h, \log N + \log \Gamma)N)$ time, where $N = n^2$ is the number of pixels (thus, the input size), $h \leq N$ is the number of different layers of the pyramid and $\log \Gamma$ is the precision of the gray levels (it is 8 if the input is a digital image using 256 gray levels).

We can also solve the optimal image segmentation

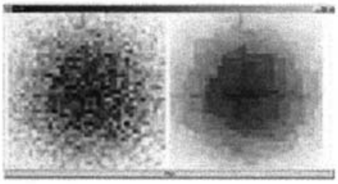

Figure 1: Pyramid approximation: $f(p)$ and $\phi(p)$ are represented by gray levels.

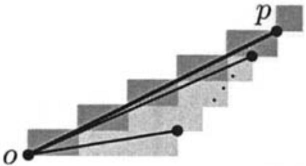

Figure 2: HDSL star-shaped region.

problems in polynomial time for star shapes and also star-shaped annuli. The results can be generalized to higher dimensional cases. We also show that the image segmentation problem is NP-hard for the family of such regions, if the objective function is the sum of (possibly negative) weights in the segmented object.

Due to the space limitation, we omit some proofs.

2 Star-shaped grid region

In Euclidean space, a region R is said to be *star-shaped* centered at a point o if for any point $p \in R$, the line segment $\overline{po}$ is entirely contained in R. This family is not only natural but also has nice mathematical properties. In particular, the family is closed under union and intersection of regions: Given two star-shaped regions R_1 and R_2 centered at o, both $R_1 \cup R_2$ and $R_1 \cap R_2$ are star-shaped with center o. Indeed, if we take the closure under union and intersection of the family of all convex regions containing o, we get the family of star-shaped regions centered at o.

However, a digital analogue of star-shaped regions in a grid geometry in an $n \times n$ pixel grid is not automatically defined, since it depends on the definition of the digital line segment between two pixels p and o.

We represent each pixel by its bottom-left corner grid point (i, j) and denote it by $\mathbf{p}(i, j)$. Thus the pixel grid $\mathbf{P} = \{\mathbf{p}(i, j) \mid 0 \leq i \leq n-1, 0 \leq j \leq n-1\}$ is identified with the grid $\mathbf{G}$ consisting of grid points $V = \{(i, j) \mid 0 \leq i \leq n-1, 0 \leq j \leq n-1\}$.

We define two graphs on the set V of grid points representing the adjacency relations of pixels. The graph $G_1 = (V, E_1)$ is defined such that the vertex (i, j) is connected to its eight neighbors (k, ℓ) satisfying $|k - i| \leq 1$ and $|\ell - j| \leq 1$. The graph $G_2 = (V, E_2)$ is a spanning subgraph of G_1, where (i, j) is only connected to its four neighbors $(i, j-1)$, $(i-1, j)$, $(i+1, j)$, and $(i, j+1)$. In both cases, we disregard the edges connecting to the pixels outside the grid $\mathbf{G}$. A subset of V is called connected in the *octagonal* grid topology (resp. *orthogonal* grid topology) if its induced subgraph in G_1 (resp. G_2) is connected.

One possible and practical definition of a digital star-shaped region in $\mathbf{P}$ is the set of all pixels intersecting with a given Euclidean star-shaped region. However, such a family of regions does not satisfy the condition that the intersection of two digital star-shaped regions centered at o is again a digital star-shaped region. For example, if we consider the line $\ell_1 : y = \frac{4n+1}{2n^2}x$ and $\ell_2 : y = \frac{3n-1}{n(n-1)}x$, both lines pass through the pixel $\mathbf{p}(n-1, 2)$. Indeed, ℓ_1 goes through the point $(n, 2+\frac{1}{2n})$ and ℓ_2 goes through the point $(n-1, 3-\frac{1}{n})$. However, they do not pass through a shared pixel in the $(n-2)$-th column. Thus, the intersection of the digital star-shaped regions corresponding to these lines becomes disconnected in the octagonal grid topology. This causes difficulties in several algorithmic problems that we discuss in this paper. Therefore, we need a better definition.

From now on, we consider a grid region as a set of grid points in $\mathbf{G}$ that defines a connected subgraph of G_1 (or in G_2). We give the following definition of digital star-shaped regions, which is based on uniquely defined digital line segments $\mathrm{dig}(po)$ between points p and a center o that simulate the Euclidean line segments $\overline{po}$.

Definition 2.1. *Digital star-shaped regions centered at o are defined such that:*

1. *any digital line segment* $\mathrm{dig}(po)$ *is a digital star-shaped region, and*
2. *for any digital star-shaped region P and any $p \in P$,* $\mathrm{dig}(po) \subseteq P$.

From this definition it is easy to see that the family $\mathcal{P}$ of digital star-shaped regions is closed under intersection and union just as in the Euclidean case. Now, the remaining task is to define digital line segments. They should be connected in G_1 (or G_2), and simulate the corresponding Euclidean line segment. By the second axiom in Definition 2.1, for each pixel $r \neq p$ in $\mathrm{dig}(po)$, $\mathrm{dig}(ro) \subset \mathrm{dig}(po)$ must hold.

There are many different formulations to define a line in the digital plane. A popular one is the *horizontal digital straight line segment (HDSL)*. For a line $y = mx+b$ with a slope $0 \leq m \leq 1$, its corresponding HDSL is defined as the set of grid points $\{(i, \lfloor mi+b \rfloor) \mid 0 \leq i \leq n\}$. This digital line is connected in the octagonal grid topology. However, HDSL and its vertical variant do not provide nice star-shaped regions. Indeed, consider any digital star-shaped region P containing a grid point $p = (i, j)$, where $1 \leq j <$

i. The HDSL corresponding to $\overline{po}$ contains the grid points (i,j) and $r = (i-1, j-1)$, but since $\text{dig}(ro) \subset \text{dig}(po)$, it must also contain $(i-2, j-2)$, and transitively, must also contain all points $(i-s, j-s)$ for $s = 0, 1, \ldots, j$. Thus, P must contain the triangular grid region defined by $\{(x,y) \mid 0 \le y \le (j/i)x,\ x \le i,\ x - y \le i - j\}$, see Figure 2. This is against the intuitive notion of a star-shape, and also inconsistent since the digital line segment itself is not a star-shaped region.

2.1 Spanning tree inducing digital lines

Let us define digital line segments such that the star-shaped regions based on them are nice. We first deal with digital line segments that are connected in the octagonal grid topology. The results are translated to the orthogonal topology later. Without loss of generality, we assume from now on that $o = (0,0)$, and consider a digital star-shaped region lying in the first octant wedge defined by $\{(x,y) \in \mathbf{G} \mid y < x\} \cup o$. Digital line segments in the other octants and for an arbitrary point o can be considered analogously. Furthermore, without loss of generality, we assume that the grid has 2^k+1 columns and rows for a natural number k, and the x-value of a grid point satisfies $0 \le x \le 2^k$. Abusing the notation, we often write n for 2^k, although the grid size is indeed $(n+1) \times (n+1)$.

In the following we give a deterministic construction of a spanning subtree T of G_1, such that for every $p = (i,j)$, the unique path from p to o defines the digital line segment $\text{dig}(po)$ simulating the line segment $\overline{po}$. Since T is a tree, we can easily see that for any $r \in \text{dig}(po)$ we have $\text{dig}(ro) \subseteq \text{dig}(po)$, as desired. Therefore, our task is to show that the distance between $\text{dig}(po)$ and $\overline{po}$ is small for this T. The L_∞ distance $L_\infty(\text{dig}(po), \overline{po})$ is the maximum vertical gap between $\text{dig}(po)$ and $\overline{po}$. Since $H(\text{dig}(po), \overline{po}) \le L_\infty(\text{dig}(po), \overline{po}) < \sqrt{2}H(\text{dig}(po), \overline{po})$ for any line segment $\overline{po}$ with a slope $0 \le m < 1$, a bound for the L_∞ distance automatically gives a similar bound for the Hausdorff distance. The tree T is illustrated in the left picture of Figure 3.

Since we define digital analogues of line segments whose slopes are positive and less than 1, we only use edges of G_1 which are horizontal or diagonal with positive unit slope. An edge connecting a vertex (i,j) and a vertex $(i+1, j)$ or $(i+1, j+1)$ is called an edge in the i-th edge-column (column, if it causes no confusion). The i-th column is called an even column (resp. odd column) if i is even (resp. odd). Note the column index starts from 0.

Any line segment should continue to a line reaching the boundary of the grid $\mathbf{G}$. Thus, the set of leaves of T must be the vertices on the boundary of $\mathbf{G}$; all other vertices should be internal vertices, except the origin o, which is considered as the root of T. Thus, the set of leaves is $\{(2^k, b) \mid b = 0, 1, 2, \ldots, 2^k - 1\}$. In order to define the tree, it suffices to define all paths from the leaves to the root.

Lemma 2.2. *If an edge $e \in T$ is horizontal (resp. diagonal), all the edges in T in the same column below e (resp. above e) must be horizontal (resp. diagonal).*

Proof. If e is horizontal and connects (i,j) and $(i+1, j)$, the edges below e connect j vertices below (i,j) to j vertices below $(i+1, j)$. If there is a diagonal edge among them, by pigeon hole principle, two edges in the column must share their right endpoint. This creates a cycle in T, and contradicts the requirement that T is a tree. If e is diagonal a similar argument holds. □

The above lemma implies that there is not much freedom for defining the tree. Indeed, in each column, there is a unique vertex (called branching vertex) that is the endpoint of both a horizontal edge and a diagonal edge.

We define the paths by giving a procedure to construct them. For convenience' sake, we denote T by T^k to show the grid size explicitly. The path towards $(2^k, 0)$ is uses only horizontal edges. This is the only path if $k = 0$ and defines T^0. If $k \ge 1$, we first give the path towards $(2^k, 2^{k-1})$, which we call the *center path* (see Figure 3). The center path is the alternating chain of horizontal and diagonal edges, starting with the horizontal edge connecting $o = (0,0)$ and $(1,0)$. Thus, the edge in each even (resp. odd) column is horizontal (resp. diagonal) for the center path. It is observed that the left vertex of the edge of an even column in the center path is on the diagonal line $y = x/2$, while its right vertex is below this line. The following lemma is a straightforward consequence of Lemma 2.2:

Lemma 2.3. *In the tree T^k, all the edges in an even column below the center path are horizontal and all the edges in an odd column above the center path are diagonal.*

Let us consider the part of T^k below and including the center path. All the even columns are fixed by Lemma 2.3 and consist of horizontal edges. The $(2i+1)$-th column can be naturally mapped mapped to the i-th column of the octant wedge of a grid of size $2^{k-1} \times 2^{k-1}$, and we copy the i-th column of T^{k-1} to the $(2i+1)$-th column of T^k. Similarly, we know the odd columns of the half of T^k above the center path, and fill the even columns by copying the i-th column of T^{k-1} to the $2i$-th column above (or on) the center path for $i = 0, 1, \ldots, 2^{k-1} - 1$. These copies do not conflict with boundary paths of T^k.

As shown in Figure 3, it is easily seen that this gives a spanning tree. Now, let us consider the distance from the line $\overline{po}$ to the digital line $\text{dig}(po)$ defined as the unique path in T from p to o. We first consider the path simulating the line segment $v_b o$, where $v_b = (2^k, b)$ and $0 \le b < 2^k$. This line segment is part of the line $y = 2^{-k}bx$. We consider the value $f_k(b, x)$ as the y-coordinate of the vertex of $\text{dig}(v_b o)$ at an integer abscissa $0 \le x < 2^k$. The vertical distance between the line $y = 2^{-k}bx$ and $\text{dig}(v_b o)$ at $x = a$ is $2^{-k}ba - f_k(b, a)$.

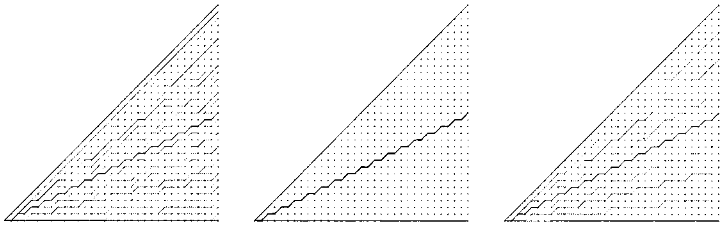

Figure 3: The spanning tree T to define digital lines (left), the center path in the tree(center), and center paths after the third level recursion.

Lemma 2.4. *For $k \in \mathbb{N}$ and $a, b \in \{0, 1, \ldots, 2^k - 1\}$ we have $2^{-k}ba \geq f_k(b, a) > 2^{-k}ba - \frac{k}{2}$.*

Proof. Let $c < 2^{k-1}$ be a nonnegative number and let $\sum_{i=0}^{k-1} b_i 2^i$ be the 2-adic expansion of b. Then, we have the following equations: If $b_{k-1} = 0$ then $f_k(b, 2c) = f_k(b, 2c+1) = f_{k-1}(b, c)$. If $b_{k-1} = 1$ and $c \geq 1$ then $f_k(b, 2c) = f_{k-1}(b - 2^{k-1}, c) + c$ and $f_k(b, 2c) = f_k(b, 2c-1) + 1$.

Indeed, if $b_{k-1} = 0$, then $b < 2^{k-1}$ and the path is below (or partially on) the center path. Since each edge in an even column is horizontal, $f_k(b, 2c) = f_k(b, 2c+1)$. Because the odd column is copied from the path in T^{k-1} simulating the line towards $(2^{k-1}, b)$ we have $f_k(b, 2c) = f_{k-1}(b, c)$. If $b_{k-1} = 1$, then $b \geq 2^{k-1}$ and the path is above or on the center path. Each edge in an odd column is diagonal, so $f_k(b, 2c-1) + 1 = f_k(b, 2c)$. Up to $2c$, these diagonal edges in odd columns gain c in the y-coordinate value. Because the odd column is copied from the path in T^{k-1} simulating the path towards $(2^{k-1}, b - 2^{k-1})$, $f_k(b, 2c) = f_{k-1}(b - 2^{k-1}, c) + c$. Clearly, $f_k(b, a)$ is defined by the above recursive formulas, and it is easy to prove the lemma by induction. □

Since $k = \log n$, we have shown that the vertical distance between $\overline{v_b o}$ and $\mathrm{dig}(v_b o)$ is less than $(\log n)/2$. This extends to the following theorem:

Theorem 2.5. *For any grid point $p \in G_1$, the vertical distance between $\mathrm{dig}(po)$ and $\overline{po}$ is less than $(\log n)/2$.*

Proof. The point must lie on the path towards some leaf v_b, and $\mathrm{dig}(po)$ is a subpath of $\mathrm{dig}(v_b o)$. Obviously, the line segment $\overline{po}$ lies below the line segment $\overline{v_b o}$. Now, suppose that the maximum vertical distance is taken at a grid point q. If q is above $\overline{po}$, it must be between $\overline{v_b o}$ and $\overline{po}$, and hence the vertical distance cannot exceed the vertical distance between p and $\overline{v_b o}$, which is less than $(\log n)/2$. If q is below $\overline{po}$, the vertical distance is less than that between q and $\overline{v_b o}$, and hence less than $(\log n)/2$. □

In order to consider the case when the point o is an interior point in the grid, we need to consider eight such trees and join them at the root o. The j-th octant covers the region $\{p \mid 2(j-1)\pi/8 \leq \theta(p) < 2j\pi/8\}$, where $\theta(p)$ is the argument angle of p. Hence, we need to be careful such that, e.g., in the second octant, the tree contains the diagonal line, but does not contain the y-axis. Thus, we transform T to obtain the tree T' in the second octant by applying a linear transformation mapping $(1, 0)$ and $(1, 1)$ to $(1, 1)$ and $(0, 1)$, respectively. The Hausdorff distance remains to be less than $(\log n)/2$ for this general case.

Now, for any Euclidean star-shaped region R with center o, we take the set $S(R)$ of pixels intersecting R, and consider $D(R) = \cup_{p \in S(R)} \mathrm{dig}(po)$. Then, $D(R)$ is a digital star shape, and the Hausdorff distance $H(R, D(R))$ is $O(\log n)$ (indeed, it is less than $1 + \frac{\log n}{2}$).

Instead of G_1, we may also use the subgraph G_2 to define the adjacency structure of the grid, where we can only use vertical and horizontal edges. For this purpose, we transform the first octant of the grid of G_1 to the first quadrant of G_2 by the linear transformation A mapping $(1, 0)$ and $(1, 1)$ to $(1, 0)$ and $(0, 1)$, respectively. This linear transformation maps T onto a spanning tree $A(T)$ of G_2. The combinatorial structure of $A(T)$ is exactly same as T, since A gives a graph isomorphism. We define a path to the root in $A(T)$ as the corresponding digital line segment in the orthogonal topology. Since $(0, 1)$ is mapped to $(-1, 1)$ by this transformation, the Hausdorff distance between a line and digital line in $A(T)$ is at most $\sqrt{2}$ times the vertical distance between their corresponding line and digital line in T. Thus, we have the following:

Theorem 2.6. *The tree $A(T)$ in the first quadrant of G_2 and its four reflected copies define line segments from the grid points towards o. The Hausdorff distance between a line segment $\overline{po}$ and the digital line segment $\mathrm{dig}(po)$ is at most $(\log n)/\sqrt{2}$.*

It is an interesting observation that the tree T has a quite uniform structure. The following observation is obtained from our recursive construction (proof is omitted in this version).

Proposition 2.7. *The path from o to $(2^\ell, b)$ has exactly ℓ branching vertices (including o) in T for any $0 \leq \ell \leq k$ and $0 \leq b < 2^\ell$.*

Let us consider the walk of a robot in T as described in the introduction. The probability to reach a vertex (i, j)

is 2^{-s} if the path from o to (i,j) visits s branching vertices in T. Thus, the probability to reach a vertex $(2^\ell, b)$ is uniformly $2^{-\ell}$ for all paths to leaves in T if we flip a random coin at each branching vertex. Accordingly, in the tree $A(T)$, the probability of reaching (i,j) such that $i+j=2^\ell$ is $2^{-\ell}$.

2.2 A lower bound on the Hausdorff distance

We show that any spanning tree of the first octant of G_1 rooted at o has a path such that the L_∞ distance from the corresponding line is $\Omega(\log n)$. Consequently, the same bound holds for the Hausdorff distance for the spanning tree covering the whole grid.

We use a classical result on pseudo-random number generation [8, 9, 10]. The following history is cited from Matousĕk [8] and Schmidt [10] in a combined fashion. Consider a sequence $X = x_0, x_1, x_2, \ldots$ of real numbers in $[0,1]$. Let $X_n(a) = |\{0 \le i \le n \mid x_i \in [0,a]\}|$ for a given $a \in [0,1]$. The *discrepancy* of the sequence $x_0, x_1, \ldots, x_n$ for a is $\max_{m<n} |am - X_m(a)|$. Van der Courpt asked in 1935 whether it is true or not that for any sequence X, there exists an a such that the discrepancy goes to ∞ when n goes to ∞. This was affirmatively answered by Van Aardenne-Ehrenfest in 1945. Roth gave an $\Omega(\sqrt{\log n})$ bound in 1954, and the correct order of magnitude of the discrepancy is $\Theta(\log n)$ given by Schmidt in 1972. It is also in a list of favorite questions of Erdős [4]. We use it in the following form:

Theorem 2.8. *Given any sequence $X = x_0, x_1, x_2 \ldots$ of real numbers in $[0,1]$ and a sufficiently large n, there exist an index $n/2 \le M < n$ and a real number $a \in [0,1]$ such that the subsequence $X_M = x_0, x_1, \ldots, x_M$ satisfies that $|aM - X_M(a)| \ge c \log n$, where c is a positive constant independent of n.*

Consider a tree T spanning the first octant in G_1 such that all the leaves are on the rightmost column (i.e., on the vertical line $x = n$). For simplicity, we assume that each path of the tree uses at most one edge in each edge-column. As we have seen, there exists exactly one branching vertex at each vertex-column. For each internal vertex v, we assign a real number $f(v) = y(l(v))/n \in [0,1)$, where $y(l(v))$ is the y-value of the highest leaf in the subtree rooted at v. By definition, $f(v) = f(u)$ if u is the unique child of v, and $f(v) = \max(f(u_1), f(u_2))$ if v has two children u_1 and u_2. Also, all values $f(v)$ in a vertex-column are sorted in increasing order if we arrange them from bottom to top in the column.

Thus, if we compare the $(i-1)$-th and the i-th column exactly one new value of f is inserted to the previous list of f values. We define x_i to be this new value of f inserted in the i-th column to obtain a sequence $x_0, x_1, \ldots, x_{n-1}$. From Theorem 2.8, we have $n/2 \le M < n$ such that $|aM - X_M(a)| \ge c \log n$.

Now, we can show that we have a path in T which is sufficiently far away from the corresponding line. The following two cases should be considered:

Case 1: $X_M(a) > aM + c\log n$. Now, consider the vertex v located at $(M, X_M(a))$. Because of the definition of $X_M(a)$, we have $f(v) \le a$, and v leads to a leaf $\ell = (n, nf(v))$. Now, consider the vertical distance of the line between o and ℓ and the path at $x = M$. The line goes through $(M, f(v)M)$ which is below (M, aM), while the path goes through $v = (M, X_M(a))$. Thus, the vertical distance is more than $c \log n$.

Case 2: $X_M(a) < aM - c\log n$. Consider the vertex v located at $(M, X_M(a)+1)$. Because of the definition of $X_M(a)$, we have $f(v) > a$, and v leads to the leaf $\ell = (n, nf(v))$. Now, consider the vertical distance of the line between o and ℓ and the path at $x = M$. The line goes through $(M, f(v)M)$ which is above (M, aM), while the path goes through $v = (M, X_M(a)+1)$. Thus, the vertical distance is more than $c \log n$. Therefore, we have the following theorem:

Theorem 2.9. *For any spanning tree T of the first octant of G_1 such that all leaves are on the grid boundary, there is a leaf q such that the vertical distance between the line $\overline{oq}$ and the path from q to o in the tree exceeds $c \log n$, where c is the constant considered in Theorem 2.8.*

We remark that the above argument implies that the sequence $x_0, x_1, \ldots x_{n-1}$ defined by $x_0 = 0$ and $x_i = 2^{-k} \max\{b | f_k(b,i) = y_{i-1}\}$ for $i \ge 1$ is a *low-discrepancy sequence* [9], where y_{i-1} is the y-value of the unique branching vertex in the $(i-1)$-th column of our tree T defined in the previous section. Although we have not yet obtained relation of this sequence to known low-discrepany sequences.

2.3 Star shapes in higher-dimensional grids

We can give an d-dimensional analogue of digital lines and digital star shapes. We utilize the fact that a line in the d-dimensional space is uniquely determined by its projections to all two-dimensional subspaces spanned by the first coordinate and the i-th coordinate for $i = 2, 3, \ldots, d$. We demonstrate our construction for the case $d = 3$.

We use our tree $T = T^k$ in G_1 covering the first octant. Although the path from $(2^k, 2^k)$ to o was not included in T in our construction, we add the path to T here in order to make the rest of the argument. Thus, the tree covers the range $0 \le y \le x \le 2^k = n$ in the plane. We call this tree $T(2)$ implying that it is a tree in the two-dimensional grid.

A vertex in the three-dimensional grid is represented by an integer triple (i,j,k). We first consider the grid structure with diagonal connections where (i,j,k) is connected to all grid points (i',j',k') with $|i'-i| \le 1$, $|j'-j| \le 1$, and $|k'-k| \le 1$. We set the point o to be the origin $(0,0,0)$ and consider the part $\mathbf{Q}$ defined by $0 \le z \le y \le x \le n$ of the grid.

In order to define a tree $T(3)$ in $\mathbf{Q}$, it suffices to define the parent of each vertex (i,j,k) in the tree. We define two copies $T(2;x,y)$ and $T(2;x,z)$ of $T(2)$ for the dimension pairs (x,y) and (x,z), which we call (x,y)-tree and (x,z)-tree, respectively. The (x,y)-tree covers the range $0 \le y \le x \le n$ and $T(2;x,z)$ covers the range $0 \le z \le x \le n$. Given a grid point $u = (i,j,k) \in \mathbf{Q}$, we call u (x,y)-horizontal (resp. (x,y)-diagonal) if the edge between (i,j) and its parent in the (x,y)-tree is horizontal (resp. diagonal). Similarly, u is called (x,z)-horizontal (resp. (x,z)-diagonal) if the edge between (i,k) and its parent in the (x,z)-tree is horizontal (resp. diagonal). The following rule defines the parent: If (i,j,k) is (x,y)-horizontal and (x,z)-horizontal, then it is connected to $(i-1,j,k)$. If it is (x,y)-horizontal and (x,z)-diagonal, it is connected to $(i-1,j,k-1)$. If it is (x,y)-diagonal and (x,z)-horizontal, it is connected to $(i-1,j-1,k)$. And if it is (x,y)-diagonal and (x,z)-diagonal, it is connected to $(i-1,j-1,k-1)$. The following lemma is observed:

Lemma 2.10. *For each $(i,j,k) \in \mathbf{Q}$, there is a unique path $\mathbf{p}$ towards o. The projection of $\mathbf{p}$ to the (x,y)-plane (resp. (x,z)-plane) coincides with the path from (i,j) (resp. (i,k)) to o in the (x,y)-tree (resp. (x,z)-tree). Moreover, all the leaves of the tree lie in the plane $x = n$.*

The following lemma is a consequence of Lemma 2.10 and Theorem 2.5:

Lemma 2.11. *For any plane $x = a$ where $0 \le a \le n$, let (a,b,c) and (a,b',c') be its intersection points with $\overline{po}$ and* dig(po)*, respectively. Then, $|b-b'| < (\log n)/2$ and $|c-c'| < (\log n)/2$.*

Therefore, the distance from $\overline{po}$ to dig(po) is less than $\sqrt{2}(\log n)/2$. If we consider the orthogonal topology such that (i,j,k) can only connect to the vertices to which the hamming distance is 1, we transform $T(3)$ by a linear transformation.

For the general d-dimensional grid, we have the following theorem, and we can define digital star-shapes in the d-dimensional grid accordingly:

Theorem 2.12. *Given a d-dimensional grid with n^d grid points in the orthogonal topology, we can define a spanning tree $T(d)$ such that the Hausdorff distance between the line segment $\overline{po}$ and the digital line segment* dig(po) *is less than $\frac{\sqrt{d-1}}{2}\log n$ if d is odd and less than $\frac{\sqrt{d}}{2}\log n$ if d is even.*

3 Mountain construction and image segmentation

In this section we deal with problems of approximating an image (or a function) on the pixel grid $\mathbf{P}$ with $N = n^2$ pixels. The results can be easily extended to the d-dimensional case.

3.1 Mountain construction

The mountain construction problem is as follows: Given a real-valued function f defined on $\mathbf{P}$, we would like to find a pyramid ϕ minimizing the L_2 distance $|f-\phi|_2 = [\sum_{p \in \mathbf{P}}(f(p)-\phi(p))^2]^{1/2}$ such that its level sets are in the family $\mathcal{S}$ of digital star-shaped regions. This is a natural variant of the least-squares method: Instead of giving some algebraic condition for ϕ, we give a geometric condition by using $\mathcal{S}$. Note that the origin o will give the peak of the mountain. We can either examine all candidates of peaks naively, or use some more efficient methods given by Chen *et al.* [2].

The following fact by Chen *et al.* [2] is our basic tool: Let $R = R(f,t)$ be the region in a family $\mathcal{O}$ maximizing $\sum_{p \in R}(f(p)-t)$ for a given real value t. If there is more than one such region, there is a maximum and a minimum (in terms of inclusion) among those regions if $\mathcal{O}$ is closed under intersection and union of regions. We denote them $R_{\max}(f,t)$ and $R_{\min}(f,t)$. Further, we call t a critical height if $R_{\max}(f,t) \ne R_{\min}(f,t)$. The following theorem shows that it suffices to compute $R(f,t)$ for each critical height t in order to compute ϕ.

Theorem 3.1. *If $\mathcal{O}$ is a region family closed under intersection and union of regions then $P(\phi,t) = R(f,t)$ for the optimal $\phi \in \mathcal{O}$ minimizing the L_2 distance from f. Moreover, if $\phi(p) = t$ for a pixel $p \in \mathbf{P}$ then $p \in R_{\max}(f,t) \setminus R_{\min}(f,t)$.*

Let us consider $\mathcal{S}$ and our tree T defining $\mathcal{S}$. For each vertex $v \in V$ of the tree T, we give a parametric weight $w(v,t) = f(v) - t$, where $f(v)$ is the value of the input function f at the pixel corresponding to v. Since $\mathcal{S}$ corresponds to the set of all rooted subtrees of T, $R(f,t)$ must be a rooted subtree of T maximizing the sum of the parametric weights of the vertices. For a given t, it is quite easy to compute $R(f,t)$: We traverse T in a bottom-up fashion starting at the leaves and cut off a vertex v (and the subtree rooted at v) if the sum of the parametric weights of v and all its descendents is negative. The final subtree obtained like this gives $R_{\max}(f,t)$. If we replace "negative" by "non-positive" in the above procedure, we obtain $R_{\min}(f,t)$. Clearly, this can be done in linear time in terms of the tree size.

Now, we can apply a so-called *hand probing* operation: Given $t_1 < t_2$ where $R_1 = R_{\max}(f,t_1) \ne R_2 = R_{\max}(f,t_2)$, we find $t_1 < t_3 < t_2$ such that R_1 and R_2 have the same parametric weight at t_3, and compute $R_3 = R_{\max}(f,t_3)$. Apparently, this operation can be done in linear time in terms of the tree size. Thus, we can either find a critical height or a new level set, and we can thus find all critical heights in $O(h)$ hand-probing operations, where h is the number of different level sets in the mountain. In total we have a $O(h|T|) = O(hN)$ time complexity. We can replace h by $\log N + \log \Gamma$ if each $f(p)$ is an integer value less than Γ by using a method given in [2], which is based on the fact that we can contract the region R_2 and also the outside of R_1 when we compute R_3. We omit the details here but observe that the time complexity to compute the mountain is $O(\min\{h, (\log N + \log \Gamma)\}N)$.

3.2 Image segmentation problems

Given f and $\mathcal{O}$, consider the optimized image segmentation problem to minimize the intra-class variance. Let R^{opt} and $\mathbf{P} \setminus R^{opt}$ be the extracted image and background, respectively. Without loss of generality, we assume that the average brightness of R^{opt} is darker than that of the background. Asano *et al.* [1] showed that for any given family $\mathcal{O}$ of regions, there exists a real number t_0 such that $R^{opt} = R_{\max}(f, t_0)$. Thus, for the family $\mathcal{S}$, R^{opt} is a level set of the optimal mountain, and the optimal image segmentation problem can be solved within the same time bounds as the optimal mountain problem. Consider the set $\mathcal{A} = \{P \setminus Q \mid P, Q \in \mathcal{S}\}$ of star-shaped annuli. We can apply a trick proposed by Wu [11] to segment a region that is the set difference of two regions in a family represented by a domination closure of a graph (in our case, T). With this trick we can solve the optimal image segmentation problem in $O(N^{2.5} \log N \log \Gamma)$ time.

It is an interesting question whether or not we can extract a region that is represented as the union of two digital star-shaped regions with different centers. More generally, we consider the following problem:

Problem 3.2. Maximum weight union of domination closures (MWUDC)
Given a vertex set V and vertex weight function $w : V \to \mathbb{R}$, consider two DAG's $H_1 = (V, E_1)$ and $H_2 = (V, E_2)$. A vertex set $X \subseteq V$ is called coupled-union of domination closures *if it is represented as the union $X_1 \cup X_2$ such that X_i are domination closures in H_i for $i = 1, 2$. The problem then is to compute the coupled-union of domination closures maximizing the weight $w(X) = \sum_{v \in X} w(v)$.*

To the authors' knowledge, the complexity of MWUDC has not been addressed before, and we show the NP-hardness of MWUDC even for a very restricted case, where H_i are rooted trees and X_i form rooted subtrees.

Theorem 3.3. *MWUDC is NP-hard even if V is the set of vertices of an $n \times n$ grid, and H_1 and H_2 are rooted trees with roots $o_1 \neq o_2$.*

Moreover, as shown in the proof, those trees can be realized as copies of our spanning tree in the grid.

4 Concluding remarks.

There are several open problems: 1. Convex regions are also useful regions in Euclidean geometry, and it is an interesting problem to define a family of regions in a digital grid approximating convex regions such that the segmentation problems can be solved in polynomial time.

2. The extraction of a union of two star-shaped polygons is NP-hard, and it is impossible to have a solution with a provable approximation ratio if we consider the sum of weights as the objective function, since we can control the value of the objective function to any small value ϵ by giving suitable weights to the roots o_1 and o_2. However, it is an interesting problem to design an approximation algorithm if the objective value is the L_2 distance of f and ϕ (i.e., the intraclass variance). Note that we have not yet proven NP-hardness for this objective function, although we believe we can modify our proof to attain it.

3. Although our $O(\log n)$ bound for the distance is asymptotically optimal, we may improve the constant factor. The current factor for the upper bound in the octagonal topology is $1/2$, while the current best lower bound factor obtained from the discrepancy theory is 0.06 [9].

The authors gratefully acknowledge to János Pach for his stimulating discussion on this topic, especially on the lower bound problem of the distance.

References

[1] T. Asano, D.Z. Chen, N. Katoh, T. Tokuyama, Efficient Algorithms for Optimization-Based Image Segmentation, *IJCGA*, **11(2)**, pp.145–166, 2001.

[2] D. Z. Chen, J. Chun, N. Katoh, T. Tokuyama, Efficient Algorithms for Approximating a Multi-dimensional Voxel Terrain by a Unimodal Terrain, *COCOON 2004, LNCS 3106*, pp.238–248, 2004.

[3] J. Chun, K. Sadakane, T. Tokuyama, Efficient Algorithms for Constructing a Pyramid from a Terrain, Proc. JCDCG02, LNCS 2866(2003) ,pp.108-117.

[4] P. Erdős, Problems and results on Diophantine approximation, *Composito Math.*, **16**, pp.52-66, 1964.

[5] D. S. Hochbaum, A New-old Algorithm for Minimum Cuts in Closure Graphs, *Networks* **37** (2001) 171-193.

[6] T. Fukuda, Y. Morimoto, S. Morishita, T. Tokuyama, Data Mining with optimized two-dimensional association rules, *ACM Transactions on Database Systems*, **26(2)**, pp.179–213, 2001.

[7] A. V. Goldberg, S. Rao, Beyond the Flow Decomposition Barrier, *J. ACM* **45-5**, pp. 783-797, 1998.

[8] J. Matoušek, *Geometric Discrepancy, An Illutrated Guide*, Springer Verlag, 1999.

[9] H. Niederreiter, *Random Number Generation and Quasi-Monte Carlo Methods*, CBMS-NSF Regional Conference Series in Applied Mathematics 63, SIAM, 1992.

[10] W. M. Schmidt, *Lectures on irreguralities of distribution,* Tata Institute of Fundamental Research, Bombay, 1977.

[11] X. Wu, Efficient Algorithms for the Optimal-Ratio Region Detection Problems in Discrete Geometry with Applications, *ISAAC 2006, LNCS 4288*, pp.289–299, 2006.

[12] X. Wu and D. Z. Chen, Optimal Net Surface Problems with Applications, *Proc. 29th International Colloquium on Automata, Languages and Programming* (2002) 1029-1042.