Voxelization in Common Sampling Lattices

Haris Widjaya
GrUVi Lab
Simon Fraser University
htw@cs.sfu.ca

Torsten Möller
GrUVi Lab
Simon Fraser University
torsten@cs.sfu.ca

Alireza Entezari
GrUVi Lab
Simon Fraser University
aentezar@cs.sfu.ca

Abstract

In this paper we introduce algorithms to voxelize polygonal meshes in common sampling lattices. In the case of Cartesian lattices, we complete the separability and minimality proof for the voxelization method presented by Huang et al [5]. We extend the ideas to general 2D lattices, including hexagonal lattices, and 3D body-centred cubic lattices. The notion of connectedness in the two lattice structures is discussed along with a novel voxelization algorithm for such lattices. Finally we present the proof that meshes voxelized with our proposed algorithm satisfy the separability and minimality criteria.

1. Introduction

The dominant 3D object representations are polygonal surface mesh representations. Most modeling techniques and programs are based on polygonal representations, where they are most effective. Surface modeling seems the most natural choice, since most real-world objects are non-transparent.

In contrast, the objects of study in scientific visualization are more of an amorphous nature, where polygonal surface representation does not sufficiently capture the details of the phenomena. Hence applications in the fields of Computational Field Simulations (CFS) as well as medical imaging (e.g Magnetic Resonance Imaging and Computed Axial Tomography) sample the underlying continuous object of interest on a 3D sampling lattice. Many visualization algorithms have been developed to display this volume data with high precision or with interactive frame rates [6].

To enhance or interact with 3D volumetric data we need to merge it with polygonal representations of objects, which is achieved by discretizing the polygonal data into the volume lattice.

The discretization step, which is commonly called *voxelization*, can have different goals. Visual fidelity can be achieved by smooth voxelization, which reduces aliasing artifacts by quantizing into 8 bit values. On the other hand surface topological qualities can be maintained by fulfilling the separability and minimality criteria, this is akin to one bit quantization of the continuous data. We will focus on these topological qualities in this paper.

While voxelization on Cartesian lattices has been studied before we will introduce algorithms for the voxelization on optimal regular lattices. These lattices have recently emerged in the field on volume graphics and show great promise in storing volumetric data more efficiently, which can lead to faster rendering algorithms [9, 7, 1, 8, 3]. The Body-Centred Cubic (BCC) lattice is one lattice that has been chosen previously for its simple indexing scheme [9]. Hence we will focus on this lattice in this paper.

2. Previous work

Voxelization can be classified into two major categories. One method of voxelization focuses on topological properties, and can be seen as a binary voxelization. Another method focuses on alias-free/smooth representation, which usually quantizes into 8 bits. In smooth voxelization two major approaches have emerged. Isotropic treatment of the volumetric data has lead to a *pre-filtering* approach, which applies a filtering step on the analytic surface data equally in all dimensions before sampling [11, 2]. Another approach is based on sampling of distance fields, which essentially smoothes the data in the direction of the surface normal [4, 10].

In this paper we focus on the topological aspects of the voxelization process. Two major concerns during binary voxelization are the quality of the voxelized surface and the efficient implementation of the algorithm. The notion of *separability* and *minimality* were introduced by Huang et al [5] as necessary conditions for an accurate and efficient voxelization. The separability condition states that any voxelization must be thick enough to prevent any rays from penetrating the surface. This condition guarantees that a discrete surface will not introduce non-existent holes. The minimality condition ensures minimum number of pixels/voxels are used to maintain separability, which translates into an efficient representation.

Huang et al [5] provide voxelization algorithms for 2D and 3D Cartesian lattices. Furthermore they formally prove the separability and minimality of the resulting voxelized surfaces. However, their proof does not cover all possible scenarios and hence is incomplete. We provide a more general proof in section 3 which covers Cartesian lattices as well.

All previous work have been based on Cartesian lattices. This paper will extend voxelization algorithms to hexagonal lattices, specifically to body-centred cubic (BCC) lattices. The lattices are superior over Cartesian lattices for their efficient distribution of sample points. A good summary of the theoretical aspects can be found in Theußl et al [9]. Many volume rendering algorithms have been adapted to such grids [7, 1, 8, 3]. Providing efficient and robust voxelization algorithms will be essential for a wide acceptance of this alternative lattice structure.

2.1. Connectedness

In order to have the notion of a *connected surface* we need to ensure that nothing can pass through the surface, i.e. that the surface separates space. Hence we require that the voxelized surface also *separates* two sides of the surface. Minimality, on the other hand, requires that all voxels set are absolutely necessary in order to ensure that the discrete surface *separates* the two sides.

Since the notion of separability (and hence minimality) is tightly bound to the notion of connectedness, we have to examine the connectedness of our underlying lattices. Two samples in a lattice are connected, when their Voronoi cells share either a face, edge, or vertex.

In the case of 2D and 3D Cartesian lattices we find Voronoi cells that are connected by only vertices (2D case) or only edges and vertices (3D case). These are really degenerate cases, which lead to special families of connectedness (or neighbors). In a Cartesian lattice we can define the notion of 4 or 8 neighborhoods for the 2D case. Note that the neighborhood in this case is defined by neighbors sharing an edge and a vertex respectively. In 3D there are three possible neighborhoods - 6, 18 and 26 neighborhoods - where neighbors share a face, an edge, and one vertex respectively.

Binary voxelization is concerned with creating a voxelized surface that is n-separable and n-minimal, where n characterizes the neighborhood notion. The foundation of the voxelization method discovered by Huang et al [5] is an observation that separability is a manifestation of topological thickness, which can be used to control separability.

Given a plane L we can control the thickness of our discrete voxelization by choosing pixels that lie in between two parallel planes L_A and L_B (see figure 1a).

Given the plane L in the Hessian normal form $Ax+By+Cz+D=0$ a pixel centered at grid point (x, y, z) lies between the (normalized) planes L_A and L_B when

$$-t \leq Ax + By + Cz + D \leq t \tag{1}$$

3. 2D lattices

In this section we will outline the algorithm for a general 2D lattice.

The point lattice is the mathematical model used when describing a regular sampling scheme [3]. A lattice is described by a set of basis vectors; moreover, every point in space is described by an integer linear combination of the basis vectors of the sampling lattice. This can be expressed mathematically using the following sampling matrix D:

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{bmatrix} V_1 V_2 \end{bmatrix} \begin{bmatrix} i \\ j \end{bmatrix} \tag{2}$$

V_1 and V_2 denote the basis vectors and i, and j denote arbitrary integer indices.

The general Voronoi diagram of this sampling lattice is shown in Figure 1c. It is easy to convince oneself of this fact by starting with an orthogonal set of basis vectors (see Figure 1b), which form a rectilinear grid. By simply rotating the second basis vector into place, one of the diagonals of the quad shortens while the other becomes longer. Hence two new edges appear on the shortened diagonal, which split the respective vertices. Hence the result is a hexagon with three parallel sets of edges. The lines perpendicular to these sets of edges create three principal directions in this lattice, which we simply denote by l_1, l_2 and l_3. The distance of the sample point (the cell centre) to the edges of these parallel edges is denoted by d_1, d_2 and d_3 (see Figure 2a). Further we denote the angle between the line to be voxelized and the three principal directions α_1, α_2, and α_3 by picking this angle such that it is always less than or equal to $\pi/2$. Now we can formulate the following theorem:

Theorem 1 *The set of all voxels $\hat{L} = \{(x, y) : -t \leq Ax + By + D \leq t\}$ where $t = \max(d_i \cos \alpha_i)$ for $i \in \{1, 2, 3\}$ is a separable and minimal representation of the line defined by A, B, and D.*

The proof follows in the next two sections.

3.1. Separability proof

Let us assume $\hat{L}$ is not separable. In this case there has to exist a path (a series of connected voxels) crossing the voxelized line, which creates a *hole*. I.e. none of the voxels on that path are in $\hat{L}$ and the endpoints of that path are on opposite sides of the surface in their entirety. Let us connect the voxel centres of this path, which create a path-line. This line intersects $\hat{L}$ at some place. Let us assume that the intersection happens between voxels A and B (see Figure 2b) in the principal direction l_k. Let us denote the intersection point I. Considering the fact, that $|AB| = 2d_k$ we can assume,

without loss of generality, that $|AI| \leq d_k$. The distance from the voxel centre A to the line L is $d = |AI|\cos(\alpha_k)$. By definition, $d \leq \max(d_i \cos(\alpha_i))$ for $i \in \{1,2,3\}$. We can conclude that $d \leq t$ and hence $A \in \hat{L}$ according to our definition of $\hat{L}$. This is a contradiction to our assumption that A is not a part of $\hat{L}$ and hence we have proven that $\hat{L}$ must be separable.

3.2. Minimality proof

Note, that for a given line orientation the angles α_1, α_2, and α_3 are identical in each Voronoi cell, such that we can determine t and the principal direction l_k such that $d_k \cos\alpha_k$ is the largest of all $d_i \cos(\alpha_i)$ for $i \in \{1,2,3\}$.

With this principal direction we can construct so called *tunnels* which are path of samples that are all co-linear along the principal direction of l_k. It is easily seen that these tunnels tile the space (see figure 1c).

Let A and B be two pixels in such a tunnel (figure 2b),line L intersects $\overline{AB}$ at one point I. Without losing generality we assume that point I is closer to A than point B. From our assumption $t = d_k \cos\alpha_k$ which means $A \in \hat{L}$. Since $|IB| = 2d_k - |IA|$, we can deduce that $|IB| > d_k$, therefore $B \notin \hat{L}$.

In fact any point other than A is at least $t = d_k$ away from I. We therefore conclude that for any tunnel formed in a principal direction there exists exactly one pixel in $\hat{L}$. Therefore removing that pixel would lead to a hole in the surface which makes this surface minimal.

3.3. Cartesian lattice

Huang et al [5] have proposed and proved the following two theorems.

Theorem 2 *Let W be the length of the side of the square pixel in figure 1b. For $t_4 = \frac{W}{2}\cos\alpha_k$ the set of pixels $\hat{L} = \{(x,y) : -t_4 \leq Ax + By + D \leq t_4$ is a 4-separable and 4-minimal representation of the line defined by A, B, and D.*

Theorem 3 *Let W be the length of the side of the square pixel in figure 1b. For $t_8 = \frac{\sqrt{2}}{2}W\cos\alpha_k$ the set of pixels $\hat{L} = \{(x,y) : -t_8 \leq Ax + By + D \leq t_8$ is a 8-separable and 8-minimal representation of the line defined by A, B, and D.*

The respective proofs had to be omitted due to space constraints. Please see [12] for details.

4. 3D lattices

While we focus on the BCC lattice for the clarity of the algorithm, the idea can be easily extended to any lattice once the Voronoi cell of that lattice is defined.

The Voronoi cell of the BCC lattice is a truncated octahedron (see figure 2c), it meets all of its neighbors through a face. There are two types of faces in the truncated octahedron, the square face and the hexagonal face. There are six square faces and eight hexagonal faces.

We can extend the idea used in Section 3 to the 3D case with the introduction of two criteria as we have two types of faces. There are seven principal directions, three perpendicular to the square faces l_i^4 and four perpendicular to the hexagonal faces l_i^6. The distance of the centre of the cell to these faces is denoted with d_i^4 and d_i^6. We also denote the angle between the plane to be voxelized and the seven principal directions with α_i^4 and α_i^6. We can now formulate the following theorem:

Theorem 4 *The set of all voxels $\hat{L} = \{(x,y,z) : -t \leq Ax + By + Cz + D \leq t\}$ where $t = \max(d_i^4 \cos\alpha_i, d_j^6 \cos\alpha_j)$ for $i \in \{1,2,3\}$ and $j \in \{1,2,3,4\}$ is a separable and minimal representation of the plane L defined by A, B, C, and D.*

Here (x,y,z) denotes the world coordinates according to our Body-centred cubic sampling matrix D (compare to the 2D case in equation 2).

The proof of this theorem is analogous to the proofs in section 3.1 and 3.2.

4.1. Cartesian lattices

Following the ideas of section 3.3 we can adapt theorem 4 to the rectilinear case or the Cartesian case as in Huang et al [5]. We can show that for the 6-neighborhood we would only have to consider the 3 principal directions l_i^4. In this case our criteria from theorem 4 simplifies to $t = \max(d_i^4 \cos\alpha_i)$. Considering that $d_i^4 = W/2$ our criterion simplifies to the identical criterion for 6-separable surfaces of Huang [5].

Considering 18-separable surfaces we have to include edge-connected cells besides all face-connected cells. This means we have to take into consideration 6 new principal directions l_k^{12}, which are formed by connecting the centre of the Voronoi cell to one of the 12 edge centres. We find the distance to the edge centres to be $d_k^{12} = \frac{\sqrt{2}}{2}W$. We can now easily formulate the criteria for 18-separable Cartesian surface voxelization - a new result that was not included in the work by Huang [5].

Theorem 5 *The set of all voxels $\hat{L} = \{(x,y,z) : -t_{18} \leq Ax + By + Cz + D \leq t_{18}\}$ where $t_{18} = \max(d_i^4 \cos\alpha_i, d_k^{12} \cos\alpha_k)$ for $i \in \{1,2,3\}$ and $k \in \{1,\ldots,6\}$ is a 18-separable and 18-minimal representation of the plane L defined by A, B, C, and D.*

Considering 26-separable surfaces, we will treat each corner as a special (degenerate) hexagon and include the principal directions l_j^6 (in addition to l_i^4, and l_k^{12}). Hence our adapted plane thickness t of theorem 4 is now: $\max(d_i^4 \cos\alpha_i, d_j^6 \cos\alpha_j, d_k^{12} \cos\alpha_k)$. (A detailed proof

Table 1. Comparative performance of the voxelization algorithm for a 256^3 Cartesian lattice, a 256^3 BCC lattice (BCC) and a 227^3 BCC lattice (BCC 70%)

Mesh	6-sep. Voxels	26-sep. Voxels	BCC Voxels	BCC 70% Voxels
batwing	26156	36649	36098	28278
bug	42232	79479	56875	44754
piano	83058	107077	106326	83878
pyramid	131071	196604	165854	130503

has been omitted due to space constraints. Please see [12] for details).

5. Results

The algorithms we have presented assume that we are voxelizing infinite planes. To be of practical use the algorithm need to be adjusted to work with polygonal meshes. This adaptation of our algorithm works similarly to the algorithm presented in [5].

We implemented the BCC voxelizer as well as the regular Cartesian lattice algorithm. The regular Cartesian lattice was implemented for 6 and 26 separable surfaces.

We tested the resulting voxelized meshes with a flooding algorithm for n-separability where $n \in \{6, 26\}$ for the Cartesian lattice. We also developed a flooding algorithm for the BCC lattice in which case we flood in all 14 directions. The test was done on a gridsize of 32^3 and the meshes are rotated in 5 degree increments from 0 to 360 degree. We applied the 6-flooding test towards 6-separable surfaces, 26-flooding towards 26-separable surfaces, and 14-flooding (in BCC lattices) towards the 14-separable surfaces. The result of this testing confirms that the surface produced by our voxelizer is indeed n-separable.

The results in table 1 were obtained from the following meshes, batwing (639 triangles), bug (912 triangles),piano (266 triangles), tetrahedron (4 triangles).

In figure 3 we show the rendering of a insect dataset (courtesy of 3D cafe). In (a) we show the traditional Cartesian lattice using 26 separable surfaces. Part (b) uses the same number of samples for the voxelization into a BCC lattice and in (c) the size of the underlying sampling lattice was reduced to 70% (which represents the same frequency content as the Cartesian lattice - see Theußl [9] for details). It can be seen that we preserve more details using the same number of sampling points in a BCC arrangement vs. a Cartesian arrangement. The insect legs show noticeably more detail in the BCC case than in the Cartesian case. It is interesting to note that the number of surface voxels is comparable to the number of 26-connected surface voxels of the Cartesian lattice. The second BCC lattice, which had 70% less samples on its base lattice is comparable in quality to the Cartesian version. Also the number of surface samples in the 70% smaller version of the BCC lattice is comparable to the number of surface samples of the 6-connected Cartesian lattice, as can be seen in table 1.

6. Conclusion and future work

We have presented surface voxelization algorithms proven to satisfy both separability and minimality constraints. We could prove our algorithm for general 2D lattices. In the 3D case we have given an algorithm for BCC grids and have improved the algorithm for regular Cartesian lattices as well. We provided a novel criteria for 18-separable and 18-minimal surfaces.

We are currently working on a better understanding of the general 3D lattice Voronoi cell, which will help us in proving separability and minimality criteria.

References

[1] H. Carr, T. Theußl, and T. Möller. Isosurfaces on Optimal Regular Samples. *Joint EUROGRAPHICS and IEEE TCVG Symposium on Visualization*, 17(10):939–954, May 2003.

[2] F. Dachille IX and A. Kaufman. Incremental triangle voxelization. In *Graphics Interface*, pages 205–212, 2000.

[3] A. Entezari, N. Röber, and T. Möller. Wavelets on optimal sampling lattices. *Technical Report, School of Computing Science, Simon Fraser University, (SFU-CMPT-TR2003-01)*, Apr. 2003.

[4] S. F. F. Gibson. Using distance maps for accurate surface representation in sampled volumes. In *IEEE Symposium on Volume Visualization*, pages 23–30, 1998.

[5] J. Huang, R. Yagel, V. Filippov, and Y. Kurzion. An accurate method for voxelizing polygon meshes. In *IEEE Symposium on Volume Visualization*, pages 119–126, 1998.

[6] M. Meißner, J. Huang, D. Bartz, K. Mueller, and R. Crawfi s. A practical comparison of popular volume rendering algorithms. In *Symposium on Volume Visualization and Graphics*, pages 81–90, 2000.

[7] N. Neophytou and K. Mueller. Space-time points splatting in 4d. *Symposium on Volume Visualization and Graphics 2002*, 17(10):97–106, October 2002.

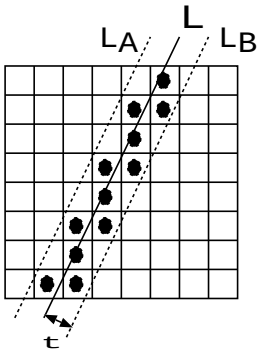
[8] N. Röber, M. Hadwiger, A. Entezari, and T. Möller. Texture based volume rendering of hexagonal data sets. *Technical Report, School of Computing Science, Simon Fraser University, (SFU-CMPT-TR2003-02)*, Apr. 2003.

[9] T. Theußl, T. Möller, and E. Gröller. Optimal regular volume rendering. *IEEE Visualization*, 17(10):939–954, Oct. 2001.

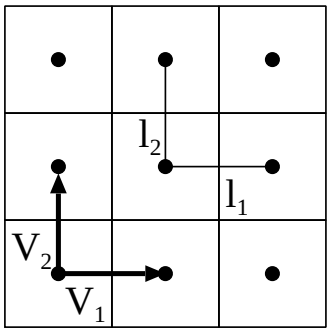
[10] A. Šrámek, M. Kaufman. Alias-free voxelization of geometric objects. In *Visualization and Computer Graphics, IEEE Transactions on*, pages 251–267, 1999.

[11] S. Wang and A. Kaufman. Volume sampled voxelization of geometric primitives. In *IEEE Visualization*, pages 78–84, 1993.

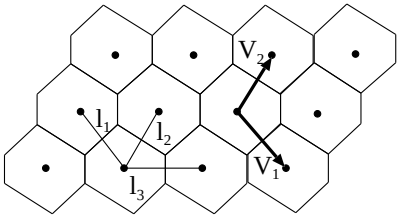
[12] H. Widjaya, T. Möller, and A. Entezari. Voxelization in common sampling lattices. *Technical Report, School of Computing Science, Simon Fraser University, (SFU-CMPT-TR2003-06)*, Aug. 2003.



(a) The plane L, and two parallel planes L_A and L_B. Any voxels that lie in between L_A and L_B is included in the representation of L

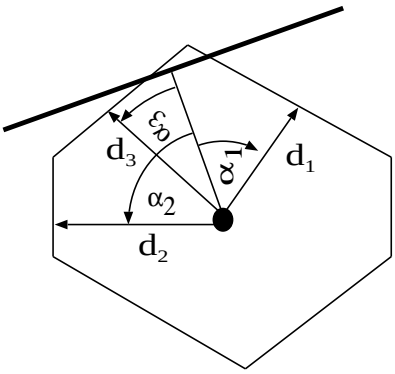


(b) A regular Cartesian lattice (also called rectilinear grid, since the basis vectors are orthogonal)

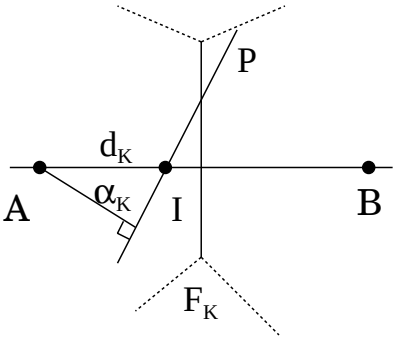


(c) A general 2D lattice. Note that we have three principal directions of connectedness – l_1, l_2, and l_3.

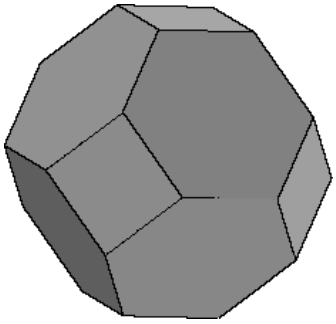
Figure 1. Reference Images 1



(a) One cell of figure 1c

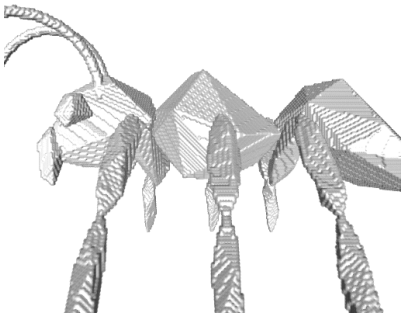


(b) A and B are the center of adjacent sample points on the lattice, F_k is the edge/face shared between A and B. $|AB| = 2d_k$

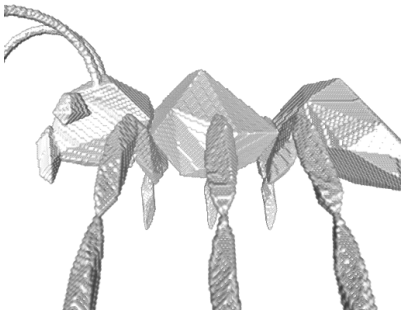


(c) The Voronoi cell of the BCC grid

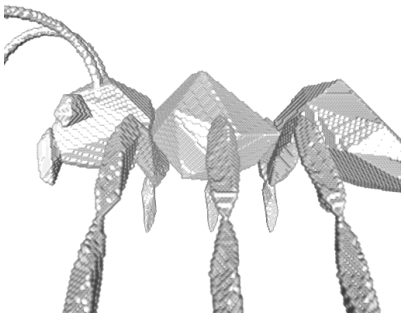
Figure 2. Reference Images 2



(a) Voxelized bug in Cartesian lattice of size 256^3 using 26-separable surface algorithm



(b) Voxelized bug in BCC lattice of size 256^3



(c) Voxelized bug in BCC lattice of size 227^3

Figure 3. Comparison images of voxelized bugs in different lattices