Automatic calculation of chamfer mask coefficients for large masks and anisotropic images

Céline Fouard, Grégoire Malandain

INRIA

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Automatic calculation of chamfer mask coefficients for large masks and anisotropic images

Céline FOUARD — Grégoire MALANDAIN

N° 4792

March 2003

THÈME 3

ISSN 0249-6399 ISRN INRIA/RR--4792--FR+ENG

INRIA
SOPHIA ANTIPOLIS

Automatic calculation of chamfer mask coefficients for large masks and anisotropic images

Céline FOUARD*, Grégoire MALANDAIN†

Thème 3 — Interaction homme-machine,
images, données, connaissances
Projet EPIDAURE

Rapport de recherche n° 4792 — March 2003 — 54 pages

Abstract: Chamfer disctances are widely used in image analysis, and many ways have been investigated to compute optimal chamfer mask coefficients. Unfortunately, these methods are not systematized: they have to be conducted manually for every mask size or image anisotropy. Since image acquisistion (e.g. medical imaging) can lead to anisotropic discrete grids with unpredictable anisotropy value, automated calculation of chamfer mask coefficients becomes mandatory for efficient distance map computation. This report presentes a systematized calculation of these coefficients based on the automatic construction of chamfer masks of any size associated with a triangulation that allows to derive analytically the relative error with respect to the Euclidean distance, in any 3-D anisotropic lattice and that also allows to compute norm constraints.

Key-words: image analysis, chamfer distance, norm constraint, anisotropic lattice

* INRIA Sophia Antipolis
† INRIA Sophia Antipolis

Unité de recherche INRIA Sophia Antipolis
2004, route des Lucioles, BP 93, 06902 Sophia Antipolis Cedex (France)
Téléphone : +33 4 92 38 77 77 — Télécopie : +33 4 92 38 77 65

Calcul automatique de coefficients de masques de chanfrein pour les grands masques et les images anisotropes

Résumé : Les distances de chanfrein sont largement utilisées en analyse d'image, et il exite plusieurs façons de calculer des coefficients optimaux pour les masques de chanfrein. Cependant, cest méthodes ne sont pas systematiques : elles doivent être effectuées "à la main" pour chaque taille de masque, ou chaque valeur d'anisotropie de l'image. Comme les acquisistions d'images (par exemple les images médicales) conduisent à des grilles discrète dont l'anisotropie n'est pas constante, l'on souhaite un calcul automatique des coefficients des masques de chanfrein pour un calcul de carte de distances efficace. Ce rapport présente une méthode de calcul systématique de ces coefficients basée sur une triangulation du masque de chanfrein qui autorise le calcul analytique de l'erreur relative par rapport à la distance Euclidienne, quelle que soit l'anisotropie ou la taille du masque. De plus, elle prend en compte des contraintes de norme.

Mots-clés : analyse d'image, distance de chanfrein, contrainte de norme, image anisotrope

Contents

1 Introduction

Distance transformations (DTs) are widely used in image analysis since they allow to recover morphometric features of a binary shape. Among other applications, they can be applied to skeleton computation [Pud98], Voronoï diagram construction, or shape-based interpolation [HZB92]. Distance transformation transforms a binary image into a grey level image where the value of each foreground pixel corresponds to its shortest distance to the background. Brute-force computation of DT is not compatible with expected image analysis requirements, so DTs are usually computed by propagation. Exact Euclidean maps can be computed using morphological operators [SM92, HM94] but with time and memory consuming algorithms. Several Euclidean distance transformations (EDT) have also been proposed (e.g. [Dan80, Rag93]) that propagate a vector instead of a scalar, leading to memory consuming approaches. A good trade-off between precision and computational cost for DT is achieved by chamfer maps that are been made popular by Borgefors [Bor86]. These maps are computed through two raster-scan on the image that propagate the distance values by the way of chamfer masks. The coefficients of the mask are (proportional) estimation of short-range distances: the larger the chamfer mask is, the closest to the Euclidean map the chamfer map will be. The calculation of optimal coefficients can be done by minimizing either an absolute error [Bor84] or a relative one [Ver91]. It has first been done for 2-D 3×3 masks [Bor84] in isotropic lattices, then extended to larger masks [Bor86, Ver91] and to higher dimensions [Bor96]. Anisotropic lattices have also been considered [MBLKF94, SB02]. However, those calculations remain tedious and are not systematized: thus they have to be conducted manually for every mask size or anisotropy value.

Our motivation is the computation of DT in 3-D medical images: they are usually acquired on anisotropic lattices (slice thickness is usually larger than the pixel size) and this anisotropy may vary from one acquisition to the other. The efficient computation of chamfer maps requires then the calculation of the chamfer mask's coefficient to be automated. calculation of these coefficients for any mask size and any anisotropy value. In addition to classical error criteria, we also consider norm constraints [Rem00] that guarantee predictable results. Our approach is based on the automatic construction of chamfer masks of

any size associated with a triangulation that allows to derive analytically the relative error with respect to the Euclidean distance.

In the following, we first recall some basic definitions. Then we describe error estimation and norm constraints. Some results (coefficients of isotropic 7^3 and anisotropic 3^3 masks) are given before we conclude.

2 Discrete Geometry

Before determining chamfer mask coefficients, we must first exactly define what a chamfer mask is, how it does work, and what its properties are. In this section, we will review some basic definitions of discrete geometry, and define the tools used to compute chamfer maps. Then, we will see some conditions on the mask, given by [Rem00], to ensure that the metric induced by the chamfer map is a norm.

2.1 Basic Properties

In this section, we review some basics in discrete geometry.

We consider the discrete space $\mathbb{E} = \mathbb{Z}^2$ *or* $\mathbb{Z}^3$.
An image I is an application from $\mathbb{E}$ to $\mathbb{Z}$:

$$I \ : \ \begin{cases} \mathbb{E} \longrightarrow \mathbb{Z} \\ (x, y, (z)) \longmapsto I(x, y, (z)) \end{cases}$$

As in the continuous space, we define a discrete distance and discrete norm as follows:

DEFINITION **2.1 (discrete distance)**
A function $d \ : \ \mathbb{E} \longrightarrow \mathbb{N}$ *is a* discrete distance *on* $\mathbb{E}$ *if and only if it verifies the following properties for each* $\vec{\mathrm{x}}, \ \vec{\mathrm{y}} \in \mathbb{E}$*:*

- $d(\vec{\mathrm{x}}) = 0 \Longleftrightarrow \vec{\mathrm{x}} = \vec{0}$ *(definition)*
- $d(\vec{\mathrm{x}}) \geq 0$ *(positivity)*
- $d(\vec{\mathrm{x}}) = d(-\vec{\mathrm{x}})$ *(symmetry)*
- $d(\vec{\mathrm{x}} + \vec{\mathrm{y}}) \leq d(\vec{\mathrm{x}}) + d(\vec{\mathrm{y}})$ *(triangular inequality)*

DEFINITION **2.2 (discrete norm)**
A function $n \ : \ \mathbb{E} \longrightarrow \mathbb{N}$ *is a* discrete norm *on* $(\mathbb{E}, \mathbb{Z})$ *if and only if it verifies the following properties for each* $\vec{\mathrm{x}}, \ \vec{\mathrm{y}} \in \mathbb{E}$ *and each* $\lambda \in \mathbb{Z}, \ \lambda.\vec{\mathrm{x}}, \ \lambda.\vec{\mathrm{y}} \in \mathbb{E}$ *:*

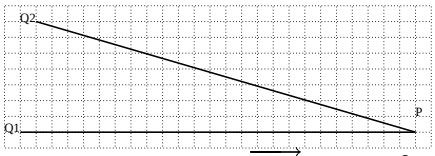
- $n(\vec{\mathrm{x}}) = 0 \Longleftrightarrow \vec{\mathrm{x}} = \vec{0}$ *(definition)*
- $n(\vec{\mathrm{x}}) \geq 0$ *(positivity)*
- $n(\vec{\mathrm{x}}) = n(-\vec{\mathrm{x}})$ *(symmetry)*
- $n(\vec{\mathrm{x}} + \vec{\mathrm{y}}) \leq n(\vec{\mathrm{x}}) + n(\vec{\mathrm{y}})$ *(triangular inequality)*
- $n(\lambda.\vec{\mathrm{x}}) = |\lambda|.n(\vec{\mathrm{x}})$ *(homogeneity)*

Two of the three usual continuous norms of $\mathbb{R}^n$:

- $N_1(u) = \sum_{i=1}^{n} |u_i|$
- $N_2(u) = \sqrt{\sum_{i=1}^{n} |u_i|^2}$
- $N_\infty = max_{i=1..n}|u_i|$

can be transposed into the discrete norms of $\mathbb{Z}^n$:

- $N_{4/6}(u) = \sum_{i=1}^{n} |u_i|$ ($= N_4$ if $n = 2$, N_6 if $n = 3$)
- $N_{8/26}(u) = max_{i=1..n}|u_i|$ ($= N_8$ if $n = 2$, N_{26} if $n = 3$)



$\|\|\overrightarrow{PQ1}\|\|_2 = 25$	$\|\|\overrightarrow{PQ2}\|\|_2 = 25$
$N_4(\overrightarrow{PQ1}) = 25$	$N_4(\overrightarrow{PQ2}) = 24$
$N_8(\overrightarrow{PQ1}) = 25$	$N_8(\overrightarrow{PQ2}) = 31$

$\overrightarrow{PQ1}$ is transformed into $\overrightarrow{PQ2}$ by rotation.

Figure 1: N_4 *and* N_8 are not invariant by rotation

These two metrics are discrete, simple.But they are not invariant by rotation (see figure 1). This means that the distance between two points of an object can vary according to the orientation of this object. This is an important drawback for many application, particularly for skeletonization, where the final result is expected not to vary under rotation. The Euclidean metric is invariant by rotation, but can not easily be transposed into a discrete distance. Indeed, neither the square nor the truncature of the Euclidean metric verify the conditions of a distance (they do not respect the triangular inequality). This will lead us to approximate the Euclidean metric, in order to keep the invariance by rotation.

To better understand how to obtain a chamfer map with good properties, we first detail what a chamfer map is, how a chamfer mask is constructed, and then which conditions on the mask yield a chamfer map which respects norm properties.

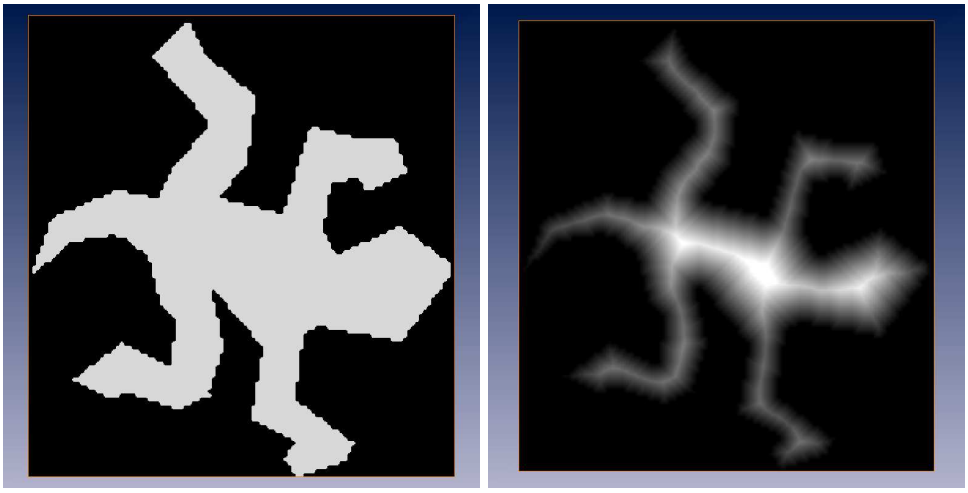
2.2 Chamfer Map

A distance map is a grey level image, where the value of each point of the foreground corresponds to its shortest distance to the background (and/or respectively, the value of each point of the background corresponds to its shortest distance to the foreground). Figure 2 gives an example of such a 2D distance map.

Formally, a distance map is defined as follows:

DEFINITION **2.3 (Distance Map)**
Let $X \subset \mathbb{E}$ be the foreground of an image I(i.e.$p \in \mathbb{E},\ I(p) > 0$) and $\overline{X}$ the background of the image (i.e.$p \in \mathbb{E},\ I(p) \leq 0$). The distance map of I is:

$$DM_X \ : \ \left\{ \begin{array}{l} \mathbb{E} \longrightarrow \mathbb{N} \\ p \longmapsto d(p, \overline{X}) = \inf\{d(p,q), q \in \overline{X}\} \end{array} \right.$$



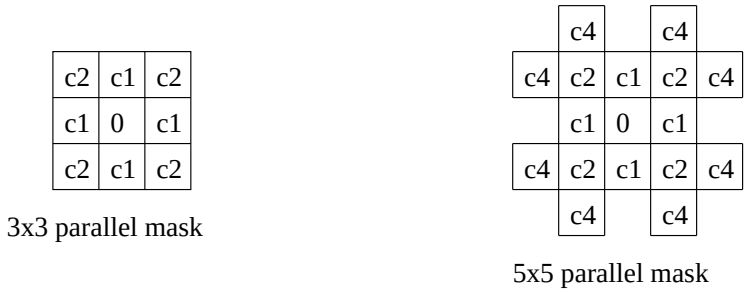
(a) Original image (b) Distance map

Figure 2: Example of 2D distance map

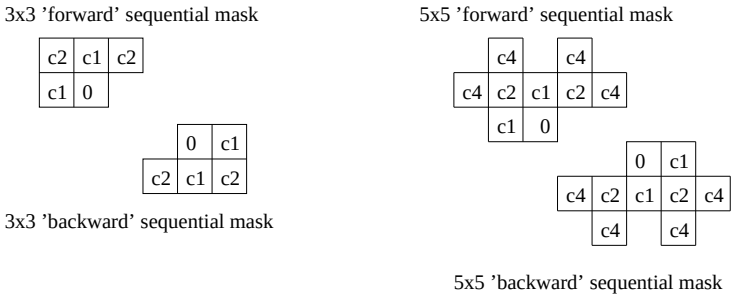
An efficient way to compute a distance map, in terms of precision and cost calculation, is to compute *chamfer distances*. Chamfer distances, first proposed by Montanari and Hilditch and popularized by Borgefors ([Bor84] et [Bor86]), are based on the idea of propagating local integer distances. A chamfer map is computed on a binary image. It uses chamfer mask to propagate local distances: chamfer masks (see figure 3) run over the image. As soon as they find a background pixel in the neighborhood of an object pixel, the minimal distance is given to the object pixel. Then the chamfer masks propagate this distance by adding the pixel size to the object neighbors of the former object pixel. Figure 4 shows the propagation of the N_8 norm inside the object.

There are two ways to compute chamfer maps : sequential and parallel.

The parallel algorithm is the most intuitive one. The parallel chamfer mask (figure 3 (a)), corresponds to a local neighborhood of the considered pixel, weighted by the different costs c_i. During the parallel process, a chamfer mask is centered on each pixel of the image. The new value of the centered pixel is obtained as follows : for each point of the mask, we compute the sum of the corresponding mask weight and the corresponding image value. The final value is the minimum between this sum and the former center pixel value. This



(a) Parallel chamfer masks of width 3 and 5



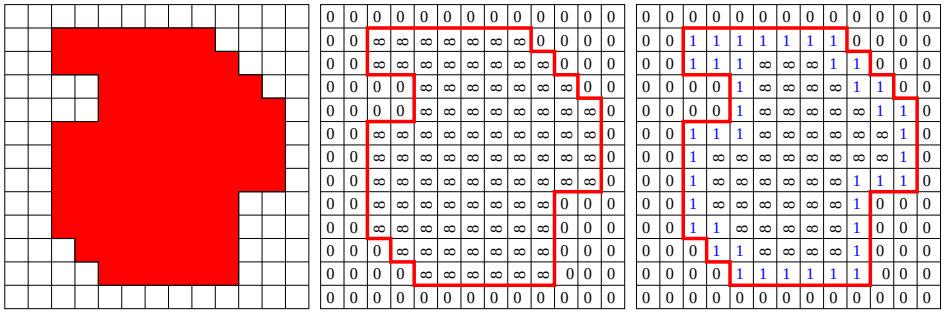
(b) Sequential chamfer masks of width 3 and 5

Figure 3: 3×3 and 5×5 2D chamfer masks

calculation is done simultaneously on every pixel of the original image, which yields to a new image. The process is applied on this new image, and so on, until no pixel is changed in the image (see figure 4). Although this algorithm can be easily parallelized, its calculation cost can not be predefined. Indeed, this cost depends on the size of the object.

Rosenfeld and Pfaltz have shown in [RP66] that the parallel algorithm could be replaced by two sequential operations on the image. To do this, we split up the parallel mask into two sequential masks symmetrically in relation to the origin, as shown in figure 3 (b).
The image is first initialized to 0 for the background and to infinity (in practice a high value) for the foreground. Then we proceed to :

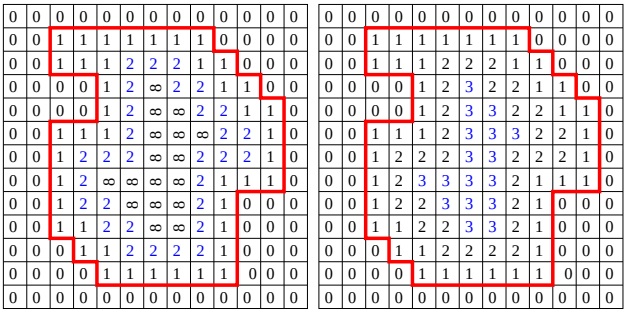
- a first *forward* pass, which goes all over the image, from the up left hand corner to the bottom right hand corner, with the forward mask



(a) initial object

(b) initialization (0 & infinity)

(c) border pixels



(d) first propagation

(e) second propagation

Figure 4: Parallel propagation of the norm N_8

- a second *backward* pass, which goes all over the image, from the bottom right hand corner to the up left hand corner, with the backward mask.

Figures 5 and 6 show the first and last iterations of the forward and the backward pass of a 3×3 chamfer mask propagating the norm N_8, i.e the 3×3 masks of the figure 3 with $c_1 = c_2 = 1$.

At each iteration, i.e. for each pixel of the image, the mask is centered on the pixel, and for each point of the mask, we calculate the sum of the cost of this point of the mask and the image value that it covers. The value of the center pixel becomes the minimum between this

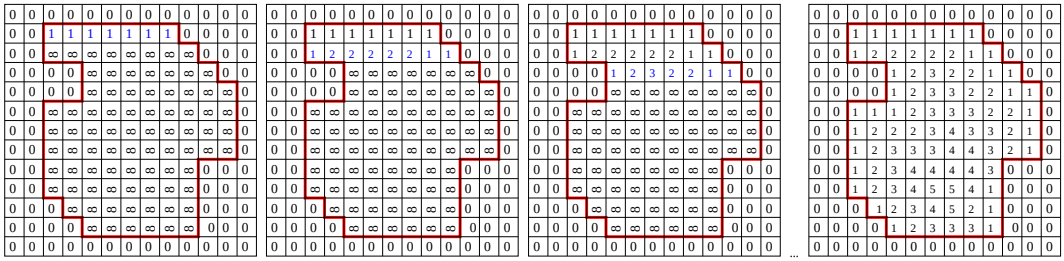


Figure 5: Forward pass of the sequential 3x3 forward mask on the initialized image 4 (b)

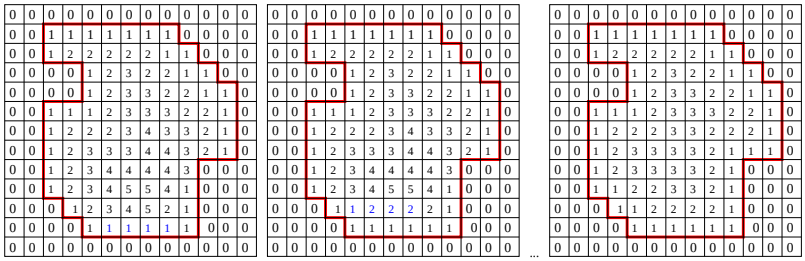


Figure 6: Backward pass of the sequential 3x3 backward mask on the last image of figure 5

sum and its previous value. Let us call (x_i, y_i) the coordinates of the different points of the mask (forward or backward), each weighted with the cost ω_i, and I the image application, we obtain :

- for the **forward pass**:

```
up to down
   left to right
      for each pixel (x,y) of the image, do:
      I(x,y) = min_{(x_i,y_i,ω_i)∈forwardMask}(I(x + x_i, y + y_i) + ω_i)
```

- for the **backward pass**:

```
down to up
   right to left
      for each pixel (x,y) of the image, do:
      I(x,y) = min_{(x_i,y_i,ω_i)∈backwardMask}(I(x − x_i, y − y_i) + ω_i)
```

This algorithm can be easily extended to three dimension. Let us call (x_i, y_i, z_i) the coordinates of the different points of the mask, and ω_i their weights, we obtain in the same way:

- for the **forward pass**:

```
front to back
   up to down
      left to right
         for each pixel (x,y,z) of the image, do:
         I(x,y,z) = min_{(x_i,y_i,z_i,ω_i)∈forwardMask}(I(x+x_i, y+y_i, z+z_i) + ω_i)
```

- for the **backward pass**:

```
back to front
   down to up
      right to left
         for each pixel (x,y,z) of the image, do:
         I(x,y,z) = min_{(x_i,y_i,z_i,ω_i)∈backwardMask}(I(x-x_i, y-y_i, z_i) + ω_i)
```

Even if we have only considered 3D images, the algorithm could be easily extended to any dimension.

2.3 Chamfer Mask

Here, we define more formally the notions that we covered more intuitively before. We define a chamfer mask as follows:

DEFINITION **2.4 (Chamfer mask)**
A chamfer mask *is a set* $\mathcal{M}_C = \{(\vec{v_i}, \omega_i),\ 1 \leq i \leq m\}$ *of weighted vectors representing authorized displacements. It is centered in* O*, symmetrical with respect to its center and contains at least a base of* $\mathbb{E}$*.*
Figure 7 shows examples of $3 \times 3(\times 3)$ chamfer masks used to compute (a): N_4, (b): N_8, (c): N_6, and (d) N_{26}.

By construction, a chamfer mask is symmetrical in relation to the axis $(O, \vec{x})$, $(O, \vec{y})$, $(O, \vec{z})$. It can thus be built from its restriction to:

- the first quarter of the plan for 2D masks, i.e. $\frac{1}{4}\mathbb{Z}^2 = \{(x, y) \in \mathbb{Z}^2 \ : \ 0 \leq x, 0 \leq y\}$ (see figure 8 a)
- the first eighth of the space for 3D masks, i.e. $\frac{1}{8}\mathbb{Z}^3 = \{(x, y, z) \in \mathbb{Z}^3 \ : \ 0 \leq x, 0 \leq y, 0 \leq z\}$ (see figure 8 c).

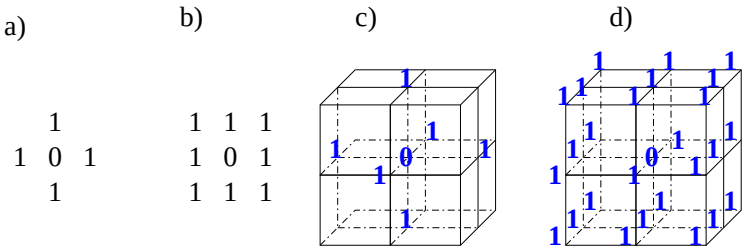
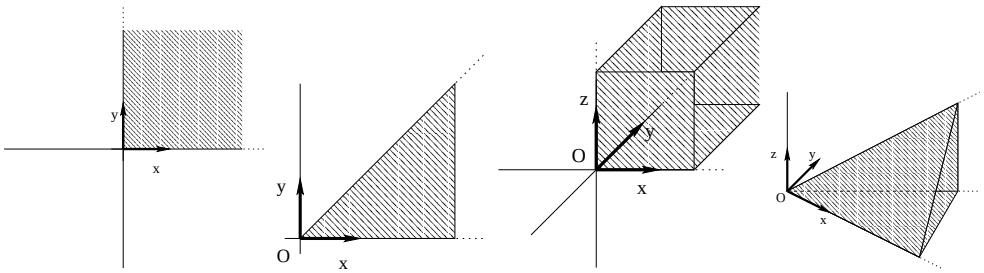


Figure 7: chamfer masks used to compute (a) N_4, (b) N_8, (c) N_6, and (d) N_{26}



(a) $\frac{1}{4}\mathbb{Z}^2$ (b) $\frac{1}{8}\mathbb{Z}^2$ (c) $\frac{1}{8}\mathbb{Z}^3$ (d) $\frac{1}{48}\mathbb{Z}^3$

Figure 8:

Moreover, for isotropic images, the voxel size is the same in the x, y and z directions. Therefore the mask is also symmetrical in relation with the bisectors of $(O, \vec{x})$, $(O, \vec{y})$ and $(O, \vec{z})$. Isotropic chamfer masks can thus be built from their restriction to:

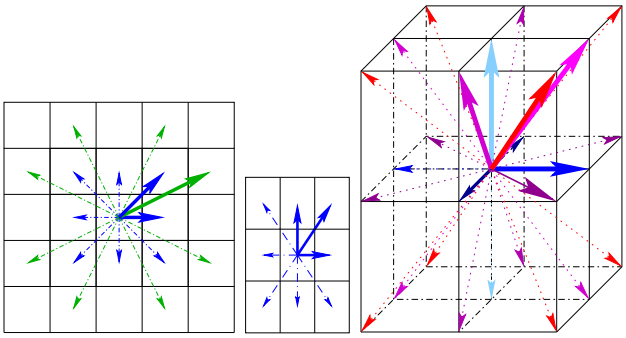
- the first eighth of the plan for 2D masks, i.e. $\frac{1}{8}\mathbb{Z}^2 = \{(x, y) \in \mathbb{Z}^2 \ : \ 0 \leq y \leq x\}$ (see figure 8 b)
- the first forty eighth of the space for 3D masks, i.e. $\frac{1}{48}\mathbb{Z}^3 = \{(x, y, z) \in \mathbb{Z}^3 \ : \ 0 \leq z \leq y \leq x\}$ (see figure 8 d).

Afterward, we will only consider the *generator* of a chamfer mask $\mathcal{M}_C^g$, that is to say, the part of the chamfer map included into :

- $\frac{1}{4}\mathbb{Z}^2$ for the 2D anisotropic masks
- $\frac{1}{8}\mathbb{Z}^2$ for the 2D isotropic masks
- $\frac{1}{8}\mathbb{Z}^3$ for the 3D anisotropic masks

- $\frac{1}{48}\mathbb{Z}^3$ for the 3D isotropic masks

Figure 9 shows some examples of such generators.



(a) 5x5 isotropic (b) 3x3 anisotropic (c) 3x3x3 anisotropic

Figure 9: Examples of chamfer mask generators

2.3.1 Chamfer Distance

DEFINITION **2.5 (path from A to B)**
Given a chamfer mask $\mathcal{M}_C$ and two points A, B, a path from A to B *is a sequence of vectors $\vec{v}_i \in \mathcal{M}_C^g$ going from A to B. We obtain :*

$$\overrightarrow{AB} = \sum_{\vec{v}_i \in \mathcal{M}_C} n_i.\vec{v}_i$$

The cost *(W) of the pass $\mathcal{P}_{AB}$ writes:*

$$W(\mathcal{P}_{AB}) = \sum_{i=1}^{m} n_i.\omega_i$$

A mask generator $\mathcal{M}_C^g$ contains at least a base of $\mathbb{E}$. Therefore we can decompose any vector $\overrightarrow{B-A}$ with vectors of $\mathcal{M}_C^g$. Associating a null weight to the null vector, we get: for all A, B in $\mathbb{E}$, it exists at least a cost $W(\mathcal{P}_{AB})$.

DEFINITION **2.6 (Chamfer distance)**
A chamfer distance *d_C between two points A and B in $\mathbb{E}$ is the minimum of the costs*

associated to every paths $\mathcal{P}_{AB}$ linking A to B :

$$d_C = min_{\mathcal{P}_{AB}} W(\mathcal{P}_{AB})$$

2.3.2 Visible Points

Let us look at the 5×5 isotropic mask generator shown in figure 9 (a). Imagine that we add a vector $\vec{v}(2,0)$ so that $\vec{v} = 2.\vec{a}$ in the mask generator (see figure 10). Then, there are

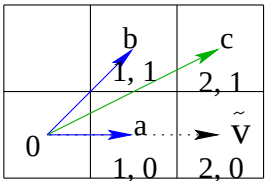


Figure 10: overloaded mask generator

two cases :

- if $\omega_{\vec{v}} = 2\omega_{\vec{a}}$ then, the mask is redundant. Rosenfeld's algorithm already associates the right weight to the point reached by $\vec{v}$. So this weight slows the algorithm unnecessarily.
- if $\omega_{\vec{v}} \neq 2\omega_{\vec{a}}$, then the distance map is not homogeneous along the $\vec{v}$ direction, which is undesirable.

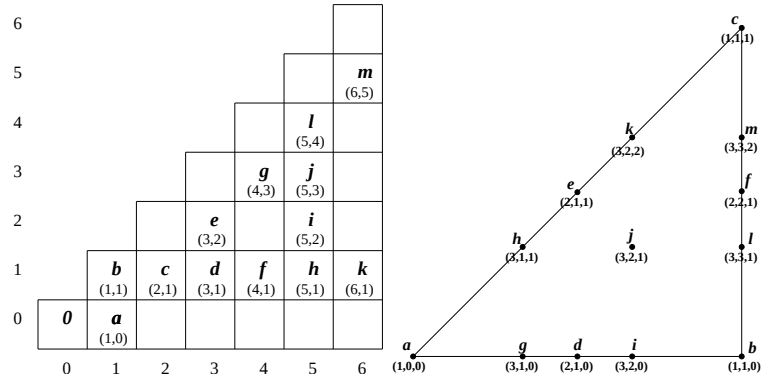
In the following we will not take any vector $\vec{v}$ in a generator such that there exists a vector $\vec{v_i}$ in the generator verifying : $\vec{v} = \lambda\vec{v_i},\ \lambda \in \mathbb{Z}$. That is to say, we will consider mask generators containing only *visible points.*

DEFINITION **2.7 (Visible point)**
A point $P(x, y(, z)) \in \mathbb{E}$ is said to be visible *from the origin, if there is no point on the fundamental network of $\mathbb{E}$ located on (OP) between O and P. A necessary and sufficient condition for P to be visible is $gcd(x, y(, z)) = 1$. (see [HW78] for a demonstration).*

Visible points can be automatically determined with Farey series (see Appendix A). Figure 11 shows 2D and 3D first visible points in the lexicographic order.

2.3.3 Influence Cone

Determining the property of the map induced by a chamfer mask, as for example its error to the Euclidean metric, is quite awkward when dealing with large masks. This difficulty can be reduced if we are able to *triangulate* the mask generator $\mathcal{M}_C^g$ into *regular cones.*



(a) first 2D visible points (b) first 3D visible points

Figure 11: First visible points in lexicographic order

In this section, we will see what a regular cone is, and how to decompose a chamfer mask generator into *regular cones*. Then, we will see that it is possible to know the property of a chamfer mask, only by knowing the properties of each of its cones. In dimension 2, the most intuitive cone organization consists in ordering Farey series points in increasing lexicographic order. Figure 12 shows an example of the organization of the cones for a 3D 7×7 mask. In dimension 3, however, a set of weighted vectors can be organized in cones in many ways: figure 13 shows two ways of ordering the vectors of the mask $\mathcal{M}_C^g$:
$\mathcal{M}_C^g = \{((1,\vec{0},0),\omega_1),((1,\vec{1},0),\omega_2),((1,\vec{1},1),\omega_3),((2,\vec{1},0),\omega_4),((2,\vec{1},1),\omega_5),((2,\vec{2},1),\omega_6)\}$.
The Farey organization (13 (a)(b)) induces the 4 following cones :
$\langle(2,\vec{1},1),(1,\vec{0},0),(2,\vec{1},0)\rangle$, $\langle(2,\vec{1},1),(2,\vec{1},0),(1,\vec{1},0)\rangle$, $\langle(2,\vec{1},1),(1,\vec{1},0),(2,\vec{2},1)\rangle$, $\langle(2,\vec{1},1),(2,\vec{2},1),(1,\vec{1},1)\rangle$.
The other organization (13 (c)(d)) induces the 4 following cones :
$\langle(2,\vec{1},1),(1,\vec{0},0),(2,\vec{1},0)\rangle$, $\langle(2,\vec{1},1),(2,\vec{1},0),(2,\vec{2},1)\rangle$, $\langle(2,\vec{1},0),(1,\vec{1},0),(2,\vec{2},1)\rangle$, $\langle(2,\vec{1},1),(2,\vec{2},1),(1,\vec{1},1)\rangle$.

Once the cone organization of a chamfer mask is decided, we can determine chamfer properties by looking at each of its cone. Let us look first at the cones properties. We can then deduce the properties of the mask.

DEFINITION **2.8 (Continuous cone)**
A continuous cone $\langle\vec{v_i},\vec{v_j}(,\vec{v_k})\rangle$ *represents the region of* $\mathbb{R}^{2,\ 3}$ *delimited by the vectors* $\vec{v_i}$, $\vec{v_j}$ *(and* $\vec{v_k}$*). That is:*
$\langle\vec{v_i},\vec{v_j}(,\vec{v_k})\rangle = \{M \in \mathbb{E} : \overrightarrow{OM} = \lambda_i.\vec{v_i} + \lambda_j.\vec{v_j}(+\lambda_k.\vec{v_k}),\ \lambda_i,\lambda_j(,\lambda_k) \in \mathbb{R}^+\}$

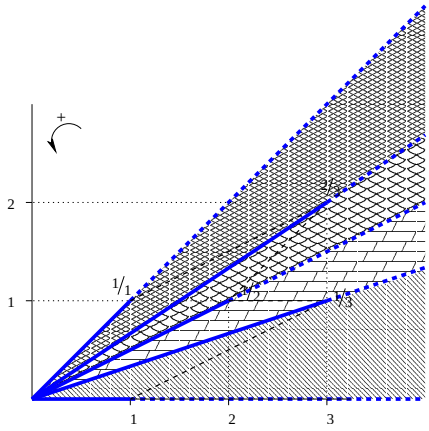


Figure 12: 2D triangulation of an isotropic 7×7 mask generator

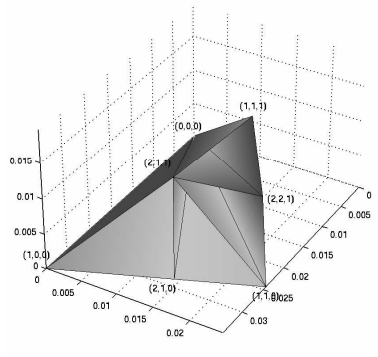
DEFINITION **2.9 (Discrete cone)**
A discrete cone $\langle\langle \vec{v_i}, \vec{v_j}(, \vec{v_k})\rangle\rangle$ *is the set of points in* $\mathbb{Z}^{2,\,3}$ *included in the continuous cone* $\langle \vec{v_i}, \vec{v_j}(, \vec{v_k})\rangle$.

DEFINITION **2.10 (Regular cone)**
A regular cone *is a discrete cone* $\langle\langle \vec{v_i}, \vec{v_j}(, \vec{v_k})\rangle\rangle$ *which verifies* $\Delta_{i,j(,k)} = \pm 1$ *where*

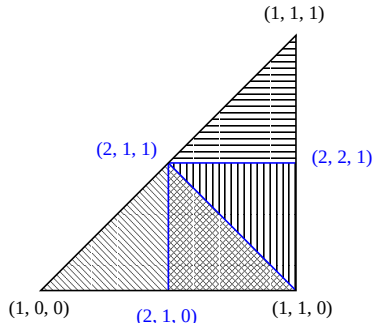
$$\Delta_{i,j,k} = \begin{vmatrix} x_{\vec{v_i}} & x_{\vec{v_j}} & (x_{\vec{v_k}}) \\ y_{\vec{v_i}} & y_{\vec{v_j}} & (y_{\vec{v_k}}) \\ (z_{\vec{v_i}}) & (z_{\vec{v_j}}) & (z_{\vec{v_k}}) \end{vmatrix}$$

Our aim is to compute chamfer mask coefficients to induce a chamfer map as close as possible to the Euclidean map. To do so, we must know the behavior of the chamfer map according to the chamfer mask coefficients, that is to say, we need to extend the local properties of the chamfer mask coefficients to the whole chamfer map. Particularly, we would like to deduce the properties of a chamfer mask from the properties of its cones which are easy to manipulate.

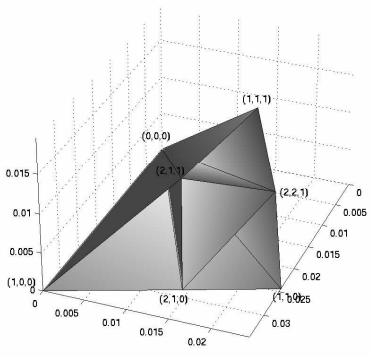
A regular cone presents the interesting property that each point which belongs to it can be reached by an integer linear combination of the vectors delimiting the cone (see [HW78] for the demonstration). Thus, if we choose a mask generator $\mathcal{M}_C^g = \{(\vec{v_i}, \omega_i),\ 1 \leq i \leq m\}$ whose triangulation is regular, for each P in a regular cone $\langle\langle \vec{v_i}, \vec{v_{i+1}}, (\vec{v_{i+2}})\rangle\rangle$ of $\mathcal{M}_C^g$, there exists a couple/triplet $(n_i, n_{i+1}, (n_{i+2})) \in \mathbb{N}^{2,\,3}$ such that $\overrightarrow{OP} = n_i\vec{v_i} + n_{i+1}\vec{v_{i+1}}(+n_{i+2}\vec{v_{i+2}})$ and which induces a minimal distance. This means that the chamfer distance of any point in the regular cone $\langle\langle \vec{v_i}, \vec{v_{i+1}}, (\vec{v_{i+2}})\rangle\rangle$ depends only on the vectors $\vec{v_i}$, $\vec{v_{i+1}}$ (and $\vec{v_{i+2}}$).
On the other hand, if the cone is not regular, there exists at least one point in the discrete cone which can not be reached by a integer linear combination of the delimiting vectors. [Ver91] gives an example (see figure 14) of this kind of cone : the vector $\vec{v}(3, 2, 1) \subset$



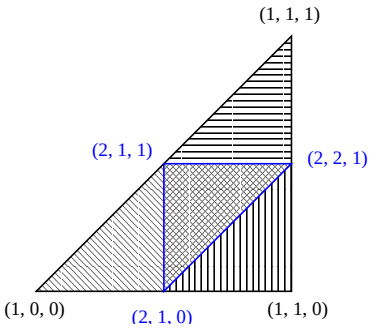
(a) Farey triangulation



(b) Projection of (a) on the Plan $x = 1$



(c) Another triangulation



(d) Projection of (c) on the Plan $x = 1$

Figure 13: Two possible organizations of a 3D isotropic $5 \times 5 \times 5$ chamfer mask generator

$\langle\langle \vec{v_1}(2,1,0), \vec{v_2}(2,1,1), \vec{v_3}(2,2,1)\rangle\rangle$ can not be reached by an integer linear combination of the 3 vectors $\vec{v_1}$, $\vec{v_2}$ and $\vec{v_3}$. We can notice that this cone is not regular since:

$$\Delta_{v_1 v_2 v_3} = \begin{vmatrix} 2 & 2 & 2 \\ 1 & 1 & 2 \\ 0 & 1 & 1 \end{vmatrix} = -2 \neq \pm 1$$

In the following we will only use regular triangulations since we want to predict the behavior of our masks. In dimension 2, if we take the points given by Farey series, ordered

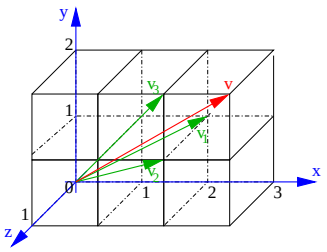


Figure 14: Example of non regular cone

in their angle with the horizontal direction (for example the 7×7 mask shown in figure 12 which generator is $\mathcal{M}_C^g = \{(\vec{a}, \omega_a), (\vec{d}, \omega_d), (\vec{c}, \omega_c), (\vec{e}, \omega_e), (\vec{b}, \omega_b)\}$), we are sure to obtain a regular triangulation (see Appendix A for the demonstration). In dimension 3, we can also build regular triangulations based on Farey series, as shown in Appendix A.

3 How to induce a norm ?

3.1 Conditions on the chamfer mask to induce a *distance*

In [Rem01], E. Remy resumes Verwer's demonstration to show that

Theorem 3.1 (distance condition)
The chamfer distance d_C induced by any chamfer mask $\mathcal{M}_C$ is a discrete distance .

However he also gives an example to illustrate that when the chamfer distance induces a discrete distance, but not a discrete norm, it can lead to unpredictable results. This example (see figure 15) is a section of a chamfer map calculated with a 2D 3×3 isotropic mask which

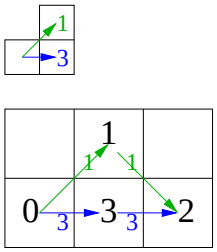


Figure 15: Example of discrete distance where the straight line is not the shortest path between to points

generator is : $\{(\vec{a}(1,0), \omega_a = 3),\ (\vec{b}(1,1), \omega_b = 1)\}$. The conditions of definition, positivity,

symmetry and triangular inequality of a discrete distance are respected. Yet we do not have the homogeneity property (this is why it is not a norm). We can notice that the straight line is not the shortest path between two points. Moreover, in a distance map which does not verifies the conditions of a norm is not invariant by scaling. This means that considering two paths, if the first one is twice larger as the second one, it may not have a distance value twice larger as the first one. Moreover, a distance that is not a norm is not scale invariant. Joint with skeletonization for example, it could lead to skeletons depending on the scale of an object. These drawbacks will lead us to consider chamfer map inducing not only a distance, but also a norm.

3.2 Conditions on the chamfer mask to induce a *norm*

In this section, we present conditions for the chamfer mask to produce chamfer maps inducing a norm on $(\mathbb{E}, \mathbb{Z})$. In fact, a function is a norm if and only if its ball is convex, symmetric and homogeneous. In [Rem01] E. Remy characterized the ball of the chamfer distance in the following way:

DEFINITION **3.2 (Equivalent rational mask)**
The equivalent rational mask $\mathcal{M}'_C$ *of a chamfer mask* $\mathcal{M}_C = \{(\vec{v_i}, \omega_i),\ 1 \leq i \leq m\}$, *is defined by:*

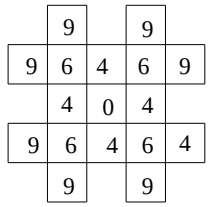
$$\mathcal{M}'_C = \left\{ \left(\frac{x_i}{\omega_i}, \frac{y_i}{\omega_i}, \frac{z_i}{\omega_i} \right) \in \mathbb{Q}^3 : (x_i, y_i, z_i, \omega_i) \in \mathcal{M}_C \right\}$$

Normalizing each weighted vector $(\vec{v_i}, \omega_i) \in \mathcal{M}_C$ by its local distance value ω_i, we obtain the following weighting: $(\frac{\vec{v_i}}{\omega_i}, \frac{\omega_i}{\omega_i}) = (\frac{\vec{v_i}}{\omega_i}, 1)$. According to the chamfer map metric, the points of the rational equivalent mask are also located at a unit distance from the origin, are they are weighted by 1. This is why we call *equivalent rational ball* the polyhedron formed by a regular triangulation of the rational equivalent mask. Figure 16 shows two examples of rational balls.

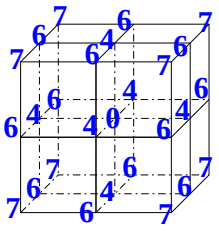
Theoreme 3.3 (Norm condition)
A chamfer mask built up with visible points and regular triangulation induces a norm *if and only if its rational equivalent ball is convex.*

To check whether an equivalent rational ball is convex or not, we can use a local convexity criterion. Indeed, the ball is convex if and only if the following local convexity criterion is verified for each edge of its polyhedron.

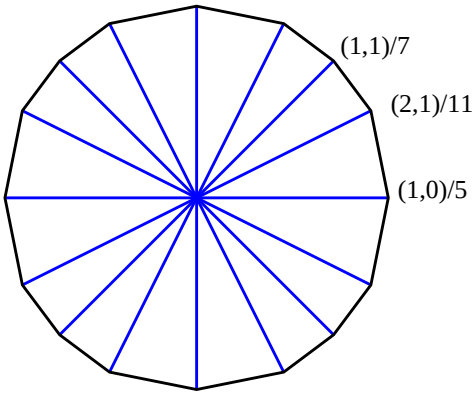
DEFINITION **3.4 (local convexity criterion)**
Two faces (P, Q, S) *and* (Q, R, S) *of a triangulation (see figure 17) verify the local convexity*



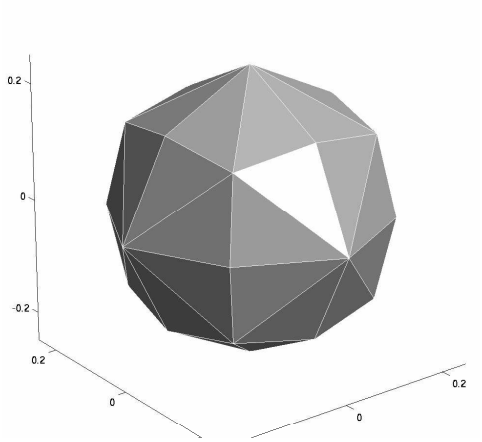
(a) 5×5 mask



(c) $3 \times 3 \times 3$ mask



(d) its equivalent rational ball



(f) its equivalent rational ball

Figure 16: Examples of equivalent rational balls

criterion (named $LCC(P,Q,R,S)$) if and only if:

$$LCC(P,Q,R,S) = \frac{1}{\omega_P.\omega_Q.\omega_R.\omega_S}. \begin{vmatrix} x_Q & x_R & x_S & x_P \\ y_Q & y_R & y_S & y_P \\ z_Q & z_R & z_S & z_P \\ \omega_Q & \omega_R & \omega_S & \omega_P \end{vmatrix} \geq 0.$$

For symmetry reasons, we come down to the chamfer mask generator only.

As the chamfer mask generator is organized in cones, we would rather express the local convexity criterion for each cone of the chamfer mask generator. Figure 18 shows the projection of a cone of the chamfer mask generator and its neighbors. We will then apply the

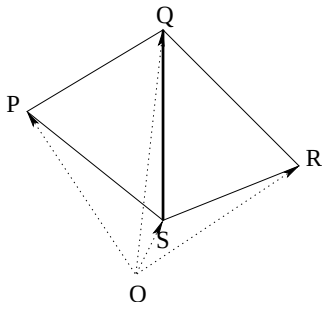


Figure 17: (P, Q, S) and (Q, R, S) are 2 faces of a triangulation

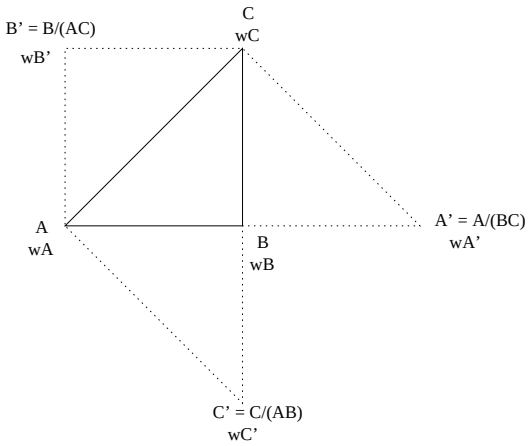


Figure 18: Geometry of LCC for chamfer mask generator cones

local convexity criterion on the three edges of the cone basis. We call A', B' and C' the symmetric points of A, B and C in the chamfer mask, that is to say, the third vertex of the neighboring triangle. If the symmetric of a point belongs to the chamfer mask generator, its weight ($\omega_{A'}$, or $\omega_{B'}$ or $\omega_{C'}$) is its corresponding one. If one of the symmetric point, A' for example is not in the generator, then, as the mask is built by symmetry around the generator, the weight $\omega_{A'}$ will be the weight of A.

We apply the LLC for each edge of the triangle ABC:

- On the edge $[BC]$, we consider the tetrahedron $ABA'C$ to obtain:

$$* \omega_A \, \Delta_{BA'C} - \omega_B \, \Delta_{A'CA} + \omega_{A'} \, \Delta_{BCA} - \omega_C \, \Delta_{ABA'} \geq 0 \tag{1}$$

  i.e.

$$\Delta_{ABC} \, \omega_{A'} - \Delta_{A'BC} \, \omega_A - \Delta_{AA'C} \, \omega_B - \Delta_{ABA'} \, \omega_C \geq 0 \tag{2}$$

- On the edge $[AC]$, we consider the tetrahedron $B'ABC$ to obtain:

$$* \omega_{B'} \Delta_{ABC} - \omega_A \Delta_{BCB'} + \omega_B \Delta_{CB'A} - \omega_C \Delta_{B'AB} \geq 0 \quad (3)$$

  i.e.

$$\Delta_{ABC} \omega_{B'} - \Delta_{B'BC} \omega_A - \Delta_{AB'C} \omega_B - \Delta_{ABB'} \omega_C \geq 0 \quad (4)$$

- On the edge $[AB]$, we consider the tetrahedron $CAC'B$ to obtain:

$$* \omega_C \Delta_{AC'B} - \omega_A \Delta_{C'BC} + \omega_{C'} \Delta_{BCA} - \omega_B \Delta_{CAC'} \geq 0 \quad (5)$$

  i.e

$$\Delta_{ABC} \omega_{C'} - \Delta_{C'BC} \omega_A - \Delta_{AC'C} \omega_B - \Delta_{ABC'} \omega_C \geq 0 \quad (6)$$

When we put together the equations 2, 4, 6, we obtain the following constraints system of inequalities:

$$\Delta_{ABC} \begin{pmatrix} \omega_{A'} \\ \omega_{B'} \\ \omega_{C'} \end{pmatrix} - \left[\begin{pmatrix} \Delta_{A'BC} & \Delta_{AA'C} & \Delta_{ABA'} \\ \Delta_{B'BC} & \Delta_{AB'C} & \Delta_{ABB'} \\ \Delta_{C'BC} & \Delta_{AC'C} & \Delta_{ABC'} \end{pmatrix} \begin{pmatrix} \omega_A \\ \omega_B \\ \omega_C \end{pmatrix} \right] \geq 0 \quad (7)$$

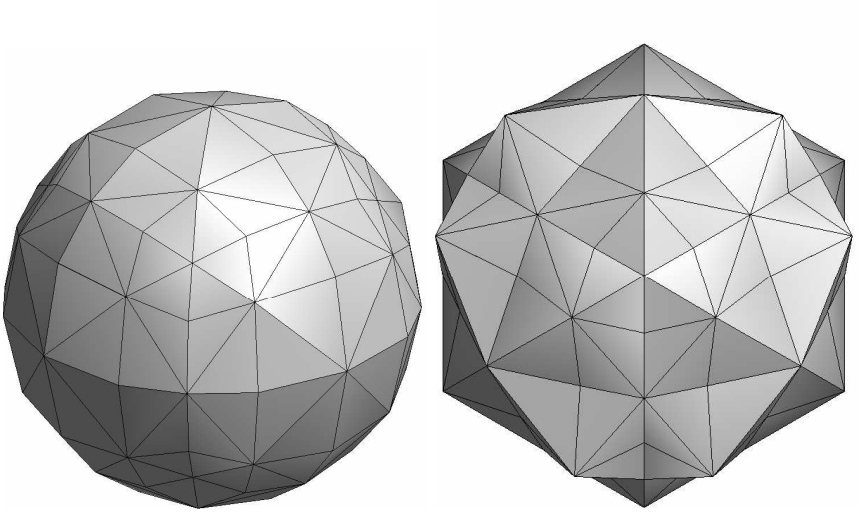
Figure 19 shows the equivalent rational balls of two different masks, one which induces a norm, and another one which does not.

4 How to be as close as possible to the Euclidean norm?

In the previous section, we saw conditions to ensure that a chamfer mask induces a norm. Now, we focus on the weights of the chamfer mask to obtain a chamfer map close to the Euclidean one. For a chamfer norm, being close the the Euclidean norm means minimizing the error between the map induced by the chamfer mask and a map induced by the Euclidean norm. There are several ways to define the error of the chamfer map with respect to the Euclidean map. In the following section, we present a short review of the state of the art in terms of error computation, before introducing our approach.

4.1 State of the art for the error computation

The first approaches to compute optimal chamfer mask coefficients consisted in searching "manually" the optimal real coefficients for a given mask, and then to adapt them for integer values.



(a) Rational ball of a 3D $5 \times 5 \times 5$ isotropic chamfer mask inducing a norm

(b) Rational ball of a 3D $5 \times 5 \times 5$ isotropic chamfer mask not inducing a norm

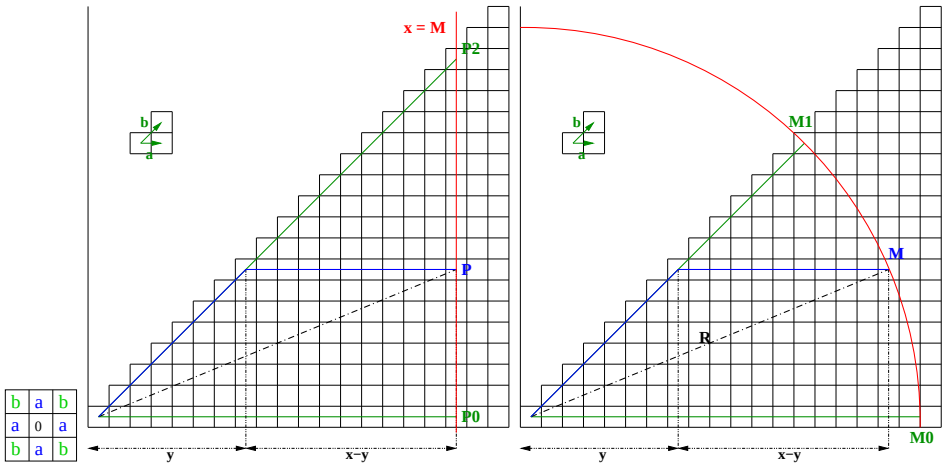
Figure 19: Convex and non convex equivalent rational balls

4.1.1 An example of 2D 3×3 isotropic real mask coefficients calculation

G. Borgefors was the first one to propose a method to calculate the optimal chamfer mask coefficients for the dimensions 2, 3 or more in [Bor84], [Bor86] and [Bor96]. Here is an example of coefficients calculation for a 2D 3×3 isotropic chamfer mask. For symmetrical reasons, there are only two coefficients to calculate : a (the horizontal one), and b (the diagonal one), as shown in figure 20 (a). Moreover, we can consider only the points such that $0 \leq y \leq x$. To find optimal coefficients, G. Borgefors minimizes the maximal error induced by the chamfer map calculation in relation with the Euclidean map. She performs the calculation on the points located on a straight line $x = M$, and minimizes the maximum error on this line. Figure 20 (b) represents the geometry of the calculation. For each point $P(M, y)$ located on the line $x = M$, we get:

- Euclidean distance : $d_{euclidian}(O, P) = \sqrt{M^2 + y^2}$
- Chamfer distance : $d_{chamfer}(O, P) = y.b + (M - y).a = y(b - a) + Ma$
- absolute error at P : $AE(y) = d_{chamfer}(y) - d_{euclidian}(y) = y(b-a) + Ma - \sqrt{M^2 + y^2}$

To obtain these values, we must have



(a) (b) on a straight line $x = M$ (c) on a circle $R = M$

Figure 20: Chamfer mask error calculation

- $b > a$ and $b < 2a$ to consider the shortest path
- $0 \leq y \leq M$ (for the other cases, we use the symmetry)

The derivative of the function
$AE : \begin{cases} [0, M] \longrightarrow \mathbb{R} \\ y \longmapsto y(b-a) + Ma - \sqrt{M^2 + y^2} \end{cases}$
is $AE' : y \longmapsto (b-a) - \frac{y}{\sqrt{M^2+y^2}}$
and is null for $y_{max} = \frac{M(b-a)}{\sqrt{1-(b-a)^2}}$. The error can thus be maximal for:

- $(P0)$ $y = 0$: $EA0_{max} = (a-1)M$
- (P) $y = y_{max}$: $EA1_{max} = (a - \sqrt{1-(b-a)^2})M$
- $(P2)$ $y = x$: $EA2_{max} = (b - \sqrt{2})M$

G. Borgefors minimizes the maximum of these three errors, and obtains optimal values fo a and b : $a_{opt} = \frac{1+\sqrt{2\sqrt{2}-2}}{2} \approx 0.95509...$ and $b_{opt} = \frac{2\sqrt{2}-1+\sqrt{2\sqrt{2}-2}}{2} \approx 1.36930....$ Note that $b_{opt} < \sqrt{2}$, indeed, taking $b = \sqrt{2}$ corresponds to consider the error only on the horizontal and diagonal directions. In fact, the maximum error comes from a direction located at about 23 degrees of the horizontal direction.

4.1.2 Approximation of real optimal values to integer chamfer mask coefficients

Now that we have optimal real coefficients, we want to approximate them by integers for homogeneity and computational reasons. To do so, we search to obtain a rational number $\frac{b_{int}}{a_{int}}$ as close as possible to $\frac{b_{opt}}{a_{opt}}$. For example, for $a_{int} = 1$, $b_{int} = round(\frac{b_{opt}}{a_{opt}} \times a_{int}) = 1$, for $a_{int} = 2$, $b_{int} = round(\frac{b_{opt}}{a_{opt}} \times a_{int}) = 3$, and for $a_{int} = 3$, $b_{int} = round(\frac{b_{opt}}{a_{opt}} \times a_{int}) = 4$... The latter one is the most popular couple of coefficients for 2D 3×3 isotropic chamfer mask. When computing a chamfer map with the chamfer mask $\mathcal{M}_C^g = \{(\overrightarrow{(1,0)}, 3), (\overrightarrow{(1,1)}, 4)\}$, one must divide it by 3 to obtain the real values. In the following, we will note $\varepsilon = a_{int}$ the scale coefficient of the chamfer map. To obtain the error induced by the map on a straight line $x = M$, one must divide $d_{chamfer}$ by ε.

4.1.3 Other error calculations

B.J.H. Verwer in [Ver91] criticized the definition of the error used, because the calculation is done on a straight line. Indeed, on a straight line, the error committed in a 45 degrees direction is bigger than the one committed in the horizontal direction. The error committed in the diagonal direction influences too much the error minimization by inducing an anisotropy in the error minimization scheme. Verwer therefore proposed to calculate this error on a unit circle. In fact, calculating an absolute error on a unit circle is equivalent to calculating a relative error on a straight line or on a circle of any radius. We summarize in the following table (see figure 1) the various error definitions and the maxima obtained by the approach introduced in the previous section.

These methods can be extended for wider 2D masks [Bor86], [Ver91], for 3D masks see [Bor96], for other dimensions [Bor84], and even for some anisotropic images [Che97], [SB02]. But they remain tedious and can not be systematized. When we get anisotropic images, their anisotropy can vary from one to the other. This is why we want an automatic method to compute chamfer mask coefficients.

4.2 Analytical expression of the error to the Euclidean distance induced by the chamfer mask

4.2.1 Which error to use ?

We want to calculate chamfer mask coefficients which will induce a chamfer map close to the Euclidean one. To do so, we choose coefficients which minimize the error with respect to the Euclidean norm. In the previous section, we saw some calculation of absolute and relative maximum error to the Euclidean norm. But these calculations were dedicated to one kind of chamfer mask. We now consider a more general expression of the error against the Euclidean norm. Thus we consider an anisotropic grid, with a unit horizontal displacement and with anisotropy in the vertical and depth direction. Figure 21 represents an element of this grid.

	For the line $x = M$	For the circle $R = M$
Point coordinates	$P(M, y)$	$P(\sqrt{M^2 - y^2}, y)$
Work Area Boundaries $(y = x)$	$y_0 = 0$ $y_1 = M$	$y_0 = 0$ $y_1 = \frac{M}{\sqrt{2}}$
$d_{euclidian}(O, P)$	$\sqrt{M^2 + y^2}$	M
$d_{chamfer}(O, P)$	$y.b + (M - y).a$ i.e. $(b - a).y + Ma$	$y.b + (\sqrt{M^2 - y^2} - y).a$ i.e. $(b - a).y + a.\sqrt{M^2 - y^2}$
	Absolute error	
$F^{abs}(y)$	$(b - a).y + Ma - \sqrt{M^2 + y^2}$	$(b - a).y + a.\sqrt{M^2 - y^2} - M$
$\frac{\partial F^{abs}}{\partial y}(y)$	$(b - a) - \frac{y}{\sqrt{M^2+y^2}}$	$(b - a) - \frac{ay}{\sqrt{M^2-y^2}}$
$\frac{\partial F^{abs}}{\partial y}(y_{Max}) = 0$	$y_{max} = (b - a)M\sqrt{\frac{1}{1-(b-a)^2}}$	$y_{max} = \frac{(b-a)M}{\sqrt{a^2+(b-a)^2}}$
Possible maxima $y = y_0 \Longrightarrow F^{abs}_{max1}(y)$ $y = y_{max} \Longrightarrow F^{abs}_{max2}(y)$ $y = y_1 \Longrightarrow F^{abs}_{max3}(y)$	 $(a - 1)M$ $(a - \sqrt{1 - (b - a)^2})M$ $(b - \sqrt{2})M$	 $(a - 1)M$ $(\sqrt{(b - a)^2 + a^2} - 1)M$ $(\frac{b}{\sqrt{2}} - 1)M$
	Relative error	
$F^{rel}(y)$	$F^{rel}_{line}(y) = \frac{F^{abs}_{line}(y)}{d_{euclidian}(y)}$ $\frac{(b-a).y+Ma}{\sqrt{M^2+y^2}} - 1$	$F^{rel}_{circle}(y) = \frac{F^{abs}_{circle}(y)}{d_{euclidian}(y)}$ $\frac{(b-a).y+a.\sqrt{M^2-y^2}}{M} - 1$
$\frac{\partial F^{rel}}{\partial y}(y)$	$\frac{M(M(b-a)-ya)}{\sqrt{M^2+y^2}^3}$	$(b - a) - \frac{ay}{M\sqrt{M^2+y^2}}$
$\frac{\partial F^{rel}}{\partial y}(y_{Max}) = 0$	$y_{max} = M\frac{b-a}{a}$	$y_{max} = \frac{M(b-a)}{\sqrt{(b-a)^2+a^2}}$
Possible maxima $y = y_0 \Longrightarrow F^{abs}_{max1}(y)$ $y = y_{max} \Longrightarrow F^{abs}_{max2}(y)$ $y = y_1 \Longrightarrow F^{abs}_{max3}(y)$	 $(a - 1)$ $\sqrt{(b - a)^2 + a^2} - 1$ $\frac{\sqrt{2}}{2}(\sqrt{2} - b)$	 $(a - 1)$ $\sqrt{(b - a)^2 + a^2} - 1$ $\frac{b}{\sqrt{2}} - 1$

Table 1: Several computation error schemes

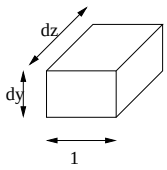


Figure 21: Anisotropic grid element

As seen earlier, the absolute error can induce an irregularity of the distance around a circle. Moreover, calculating the relative error on a plane is equivalent to calculating the absolute error on the unit circle. We choose the projection on planes as the associated calculation is easier.

4.2.2 General expression

For symmetrical reasons, the calculation of the maximum and minimum error to the Euclidean distance is made only for the generator of the mask. Moreover, as seen in section 2.3.3 the generator of a mask is composed of regular cones. Let $\langle\langle \vec{v}_i, \vec{v}_j, \vec{v}_k \rangle\rangle$ be a discrete regular cone of a chamfer mask. For each point $P(x, y, z)$ of this discrete cone, the chamfer norm n_C at the point P depends only on ω_i, ω_j and ω_k. This means that threre exist $a, b, c \in \mathbb{Z}$ such that

$$n_C(P) = a.\omega_i + b.\omega_j + c.\omega_k \tag{8}$$

with

$$\overrightarrow{OP} = a.\vec{v}_i + b.\vec{v}_j + c.\vec{v}_k$$

We obtain the following system:

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} x_i & x_j & x_k \\ y_i & y_j & y_k \\ z_i & z_j & z_k \end{pmatrix} \begin{pmatrix} a \\ b \\ c \end{pmatrix}$$

i.e.

$$\begin{pmatrix} a \\ b \\ c \end{pmatrix} = \begin{pmatrix} x_i & x_j & x_k \\ y_i & y_j & y_k \\ z_i & z_j & z_k \end{pmatrix}^{-1} \begin{pmatrix} x \\ y \\ z \end{pmatrix}$$

As we considerer regular cones, $\Delta_{i,j,k} = \pm 1$ and the matrix is reversal. We obtain:

$$a = \frac{1}{\Delta_{i,j,k}} \begin{vmatrix} x & x_j & x_k \\ y & y_j & y_k \\ z & z_j & z_k \end{vmatrix}$$

$$b = \frac{1}{\Delta_{i,j,k}} \begin{vmatrix} x_i & x & x_k \\ y_i & y & y_k \\ z_i & z & z_k \end{vmatrix}$$

$$c = \frac{1}{\Delta_{i,j,k}} \begin{vmatrix} x_i & x_j & x \\ y_i & y_j & y \\ z_i & z_j & z \end{vmatrix}$$

$$a = \frac{1}{\Delta_{i,j,k}} \Big((y_j z_k - y_k z_j).x + (x_k z_j - x_j z_k).y + (x_j y_k - x_k y_j).z \Big) \tag{9}$$

$$b = \frac{1}{\Delta_{i,j,k}}\Big((y_k z_i - y_i z_k).x + (x_i z_k - x_k z_i).y + (x_k y_i - x_i y_k).z\Big) \tag{10}$$

$$c = \frac{1}{\Delta_{i,j,k}}\Big((y_i z_j - y_j z_i).x + (x_j z_i - x_i z_j).y + (x_i y_j - x_j y_i).z\Big) \tag{11}$$

$$\tag{12}$$

If we replace a, b, c by their values ginen by (9), (10), and 11 in 8, we obtain:

$$\begin{aligned} n_C &= \tfrac{1}{\Delta_{i,j,k}} \Big(\Big((y_j z_k - y_k z_j).x + (x_k z_j - x_j z_k).y + (x_j y_k - x_k y_j).z\Big).\omega_i \\ &\quad + \Big((y_k z_i - y_i z_k).x + (x_i z_k - x_k z_i).y + (x_k y_i - x_i y_k).z\Big).\omega_j \\ &\quad + \Big((y_i z_j - y_j z_i).x + (x_j z_i - x_i z_j).y + (x_i y_j - x_j y_i).z\Big).\omega_k \Big) \\ &= \tfrac{1}{\Delta_{i,j,k}} \Big(\Big((y_j z_k - y_k z_j).\omega_i + (y_k z_i - y_i z_k).\omega_j + (y_i z_j - y_j z_i).\omega_k\Big).x \\ &\quad + \Big((x_k z_j - x_j z_k).\omega_i + (x_i z_k - x_k z_i).\omega_j + (x_j z_i - x_i z_j).\omega_k\Big).y \\ &\quad + \Big((x_j y_k - x_k y_j).\omega_i + (x_k y_i - x_i y_k).\omega_j + (x_i y_j - x_j y_i).\omega_k\Big).z \Big) \end{aligned}$$

This can be expressed by:

$$n_C(x, y, z) = \alpha.x + \beta.y + \gamma.z \tag{13}$$

with

- $\alpha = (y_j z_k - y_k z_j).\omega_i + (y_k z_i - y_i z_k).\omega_j + (y_i z_j - y_j z_i).\omega_k$
- $\beta = (x_k z_j - x_j z_k).\omega_i + (x_i z_k - x_k z_i).\omega_j + (x_j z_i - x_i z_j).\omega_k$
- $\gamma = (x_j y_k - x_k y_j).\omega_i + (x_k y_i - x_i y_k).\omega_j + (x_i y_j - x_j y_i).\omega_k$

The Euclidean norm at this point is:

$$n_E(x, y, z) = \sqrt{d_x^2 x^2 + d_y^2 y^2 + d_z^2 z^2} \tag{14}$$

The expressions 13 and 14 are also verified in the continuous cone.

The chamfer map is filled with integers and scaled with the weight of the unit displacement (usually, the horizontal displacement). To obtain the real norm map value, we must scale the map by this factor (called ε). This leads us to consider the following relative error:

$$E_{relative} = \frac{\frac{n_C}{\varepsilon} - n_E}{n_E}$$

$$E_{relative}(x, y, z) = \frac{\alpha x + \beta y + \gamma z}{\varepsilon\sqrt{d_x^2 x^2 + d_y^2 y^2 + d_z^2 z^2}} - 1 \tag{15}$$

Figure 22 shows the cones of a $5 \times 5 \times 5$ mask generator. Depending on the orientation of the cone, this error has to be minimized on either the plane $x = M$, or $y = M$, or $z = M$.

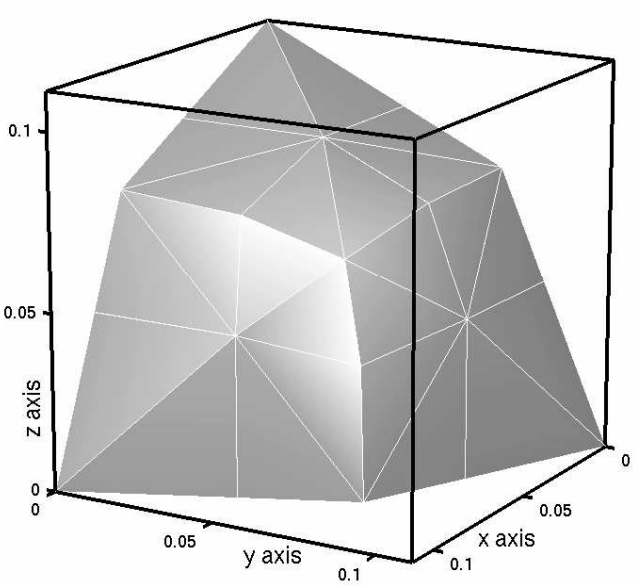


Figure 22: Cones of a $5 \times 5 \times 5$ mask generator

4.2.3 Projection on the plan $x = M, M \neq 0$

In this section we consider the points on the intersection of a cone $\langle\langle \overrightarrow{v_i}, \overrightarrow{v_j}, \overrightarrow{v_k} \rangle\rangle$ and the plan $x = M$. The vertices of this triangle are the points $V_l = (M, M\frac{y_l}{x_l}, M\frac{z_l}{x_l}) = (M, My'_l, Mz'_l)$ for $l = i, j, k$. The coordinates of $P(x, y, z)$ become $P_{px}(Mx', My', Mz')$

- $x' = \frac{x}{M} = 1$,
- $y' = \frac{y}{M}$ and
- $z' = \frac{z}{M}$.

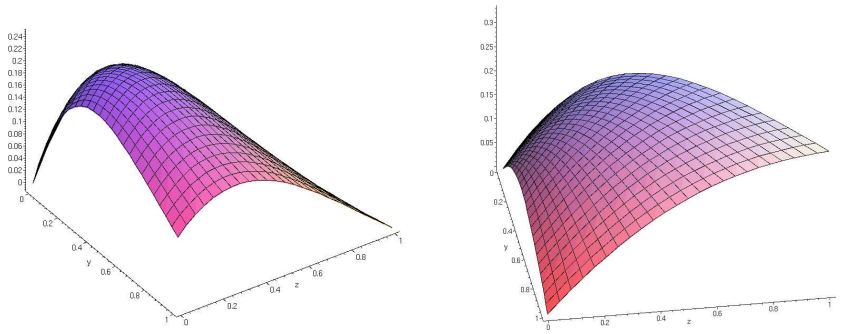
On this plan, the error function becomes:

$$F_x : (y', z') \longrightarrow \frac{1}{\varepsilon} \frac{\alpha + \beta y' + \gamma z'}{\sqrt{d_x^2 + d_y^2 y'^2 + d_z^2 z'^2}} - 1 \tag{16}$$

This function is defined and continuous on the triangle $(V_i V_j V_k)$.

Figure 23 shows a 3D representation of this function for the unique cone of the following chamfer mask generators:

(a) $\mathcal{M}_C^g = \{(\overrightarrow{(1,0,0)}, 3), (\overrightarrow{(1,1,0)}, 6), (\overrightarrow{(1,1,1)}, 8)\}$ in an anisotropic image: $d_x = 1$, $d_y = 1.5$, $d_z = 2$



(a) $d_x = 1$, $d_y = 1.5$, $d_z = 2$, $\alpha = 3$, $\beta = 3$, $\gamma = 2$

(b) $d_x = 1$, $d_y = 1$, $d_z = 1$, $\alpha = 1$, $\beta = \sqrt{2} - 1$, $\gamma = \sqrt{5} - \sqrt{2}$

Figure 23: $F_x(y', z')$

(b) $\mathcal{M}_C^g = \{(\overrightarrow{(1,0,0)}, 1), (\overrightarrow{(1,1,0)}, \sqrt{2}), (\overrightarrow{(1,1,1)}, \sqrt{5})\}$ in an isotropic image.

As F_x is continuous on a closed and bounded interval, it is bounded and reaches its bounds. Its extrema can be located inside the triangle (case (a)), on the edges of the triangle (case (b)), or on the summit of the triangle (case (c)).

a The extreme value is inside the triangle

$$\frac{\partial F_x}{\partial y'}(y', z') = \frac{\beta d_x^2 - (\alpha + \gamma z') y' d_y^2 + \beta z'^2 d_z^2}{\varepsilon (d_x^2 + d_y^2 y'^2 + d_z^2 z'^2)^{\frac{3}{2}}} \tag{17}$$

$$\frac{\partial F_x}{\partial z'}(y', z') = \frac{\gamma d_x^2 + \gamma y'^2 d_y^2 - (\alpha + \beta y') z' d_z^2}{\varepsilon (d_x^2 + d_y^2 y'^2 + d_z^2 z'^2)^{\frac{3}{2}}} \tag{18}$$

(17) and (18) are both null for $(y_{xMax}, z_{xMax}) = (\frac{\beta d_x^2}{\alpha d_y^2}, \frac{\gamma d_x^2}{\alpha d z^2})$

The extreme value of the function becomes :

$$F_{xMax} = F_x(\frac{\beta d_x^2}{\alpha d y^2}, \frac{\gamma d_x^2}{\alpha d z^2}) = \frac{1}{\varepsilon} \sqrt{\frac{\alpha^2}{d_x^2} + \frac{\beta^2}{d_y^2} + \frac{\gamma^2}{d_z^2}} - 1 \tag{19}$$

b The extreme is on an edge

There are three edges, but we will only present the calculation for $[V_iV_j]$. In this case, a point P belonging to the edge can be represented by $P = aV_i + (1-a)V_j$, which means:

- $y' = ay'_i + (1-a)y'_j = (y'_i - y'_j)a + y'_j$
- $z' = az'_i + (1-a)z'_j = (z'_i - z'_j)a + z'_j$

The error function becomes:

$$F_{xEdge}(a) = \frac{1}{\varepsilon}\frac{\alpha + \beta(a(y_i - y_j) + y_j) + \gamma(a(z_i - z_j) + z_j)}{\sqrt{1 + dy^2(a(y_i - y_j) + y_j)^2 + dz^2(a(z_i - z_j) + z_j)^2}} - 1 \qquad (20)$$

which is extremal for:

$$a_{max} = \frac{(\beta(y'_i - y'_j) + \gamma(z'_i - z'_j))d_x^2 - y'_j(\alpha(y'_i - y'_j) + \gamma(y'_iz'_j - y'_jz'_i))d_y^2 - z'_j(\alpha(z'_i - z'_j) + \beta(z'_iy'_j - z'_jy'_i))d_z^2}{(y'_i - y'_j)(\alpha(y'_i - y'_j) + \gamma(y'_iz'_j - y'_jz'_i))d_y^2 + (z'_i - z'_j)(\alpha(z'_i - z'_j) + \beta(z'_iy'_j - z'_jy'_i))d_z^2}$$

If $0 \leq a_{max} \leq 1$, the extreme value $F_{xEdgeMax}$ of F_{xEdge} is given by $F_{xEdgeMax}(a_{max})$ whose form is not simple enough to be displayed here.

c The extremum is reached on one of the triangle's vertices V_l, where the relative error is given by:

$$F_{xVertex} = \frac{1}{\varepsilon}\frac{\omega_l}{\sqrt{x_l^2d_x^2 + y_l^2d_y^2 + z_l^2d_z^2}} - 1$$

4.2.4 Projection on the plan $y = M, M \neq 0$

In this section we consider the points on the intersection of a cone $\langle\langle \overrightarrow{v_i}, \overrightarrow{v_j}, \overrightarrow{v_k}\rangle\rangle$ and the plan $y = M$.

The vertices of this triangle are the points $V_l = (M\frac{x_l}{y_l}, M, M\frac{z_l}{y_l}) = (Mx'_l, M, Mz'_l)$ for $l = i, j, k$. The coordinates of $P(x, y, z)$ become $P_{px}(Mx', My', Mz')$

- $x' = \frac{x}{M}$,
- $y' = \frac{y}{M} = 1$ and
- $z' = \frac{z}{M}$.

On this plan, the error function becomes:

$$F_y : (x', z') \longrightarrow \frac{1}{\varepsilon}\frac{\alpha x' + \beta + \gamma z'}{\sqrt{d_x^2x'^2 + d_y^2 + d_z^2z'^2}} - 1 \qquad (21)$$

a The extreme is inside the triangle

$$\frac{\partial F_y}{\partial x'}(x',z') = \frac{-(\beta+\gamma z')x' d_x^2 + \alpha d_y^2 + \alpha z'^2 d_z^2}{\varepsilon(d_x^2 x'^2 + d_y^2 + d_z^2 z'^2)^{\frac{3}{2}}} \quad (22)$$

$$\frac{\partial F_y}{\partial z'}(x',z') = \frac{\gamma x'^2 d_x^2 + \gamma d_y^2 - (\beta+\alpha x')z' d_z^2}{\varepsilon(d_x^2 x'^2 + d_y^2 + d_z^2 z'^2)^{\frac{3}{2}}} \quad (23)$$

(22) and (23) are both null for $(x_{yMax}, z_{yMax}) = \left(\frac{\alpha d_y^2}{\beta d_x^2}, \frac{\gamma d_y^2}{\beta d_z^2}\right)$.

The extreme value of the function becomes :

$$F_{yMax} = \frac{1}{\varepsilon}\sqrt{\frac{\alpha^2}{d_x^2} + \frac{\beta^2}{dy^2} + \frac{\gamma^2}{dz^2}} - 1 \quad (24)$$

b The extreme is on an edge

There are three edges, but we will only present the calculation for $[V_i V_j]$. In this case, a point P belonging to the edge can be represented by $P = aV_i + (1-a)V_j$, which means:

- $x' = ax'_i + (1-a)x'_j = (x'_i - x'_j)a + x'_j$
- $z' = az'_i + (1-a)z'_j = (z'_i - z'_j)a + z'_j$

The error function becomes:

$$F_{yEdge}(a) = \frac{1}{\varepsilon}\frac{\alpha(a(x'_i - x'_j) + x'_j) + \beta + \gamma(a(z'_i - z'_j) + z'_j)}{\sqrt{dx^2(a(x'_i - x'_j) + x'_j)^2 + d_y^2 + d_z^2(a(z'_i - z'_j) + z'_j)^2}} - 1 \quad (25)$$

which is extremal for:

$$a_{max} = \frac{-x'_j(\beta(x'_i - x'_j) + \gamma(x'_i z'_j - x'_j z'_i))d_x^2 + (x'_i - x'_j) + \gamma(z'_i - z'_j))d_y^2 - z'_j(\beta(z'_i - z'_j) + \alpha(z'_i x'_j - z'_j x'_i))d_z^2}{(x'_i - x'_j)(\beta(x'_i - x'_j) + \gamma(x'_i z'_j - x'_j z'_i))d_x^2 + (z'_i - z'_j)(\beta(z'_i - z'_j) + \alpha(z'_i x'_j - z'_j x'_i))d_z^2}$$

If $0 \leq a_{max} \leq 1$, the extreme value $F_{xEdgeMax}$ of F_{xEdge} is given by $F_{xEdgeMax}(a_{max})$ whose form is not simple enough to be displayed here.

c The extremum is reached on one of the triangle's vertices V_l, where the relative error is given by:

$$F_{yVertex} = \frac{1}{\varepsilon}\frac{\omega_l}{\sqrt{x_l^2 d_x^2 + y_l^2 d_y^2 + z_l^2 d_z^2}} - 1$$

4.2.5 Projection on the plan $z = M, M \neq 0$

In this section we will consider the points on the intersection of a cone $\langle\langle \overrightarrow{v_i}, \overrightarrow{v_j}, \overrightarrow{v_k} \rangle\rangle$ and the plan $z = M$.

The vertices of this triangle are the points $V_l = (M\frac{x_l}{z_l}, M\frac{y_l}{z_l}, M) = (Mx'_l, My'_l, M)$ for $l = i, j, k$. The coordinates of $P(x, y, z)$ become $P_{px}(Mx', My', Mz')$

- $x' = \frac{x}{M}$,
- $y' = \frac{y}{M}$ and
- $z' = \frac{z}{M} = 1$.

On this plan, the error function becomes:

$$F_z : (x', y') \longrightarrow \frac{1}{\varepsilon} \frac{\alpha x' + \beta y' + \gamma}{\sqrt{d_x^2 x'^2 + d_y^2 y'^2 + d_z^2}} - 1 \tag{26}$$

a The extreme is inside the triangle

$$\frac{\partial F_z}{\partial x'}(x', y') = \frac{-(\gamma + \beta y') x' d_x^2 + \alpha y'^2 d_y^2 + \alpha d_z^2}{\varepsilon (d_x^2 x'^2 + d_y^2 y'^2 + dz^2)^{\frac{3}{2}}} \tag{27}$$

$$\frac{\partial F_z}{\partial y'}(x', y') = \frac{\beta x'^2 d_x^2 - (\gamma + \alpha x') y' d_y^2 + \beta d_z^2}{\varepsilon (d_x^2 x'^2 + d_y^2 y'^2 + dz^2)^{\frac{3}{2}}} \tag{28}$$

(27) and (28) are both null for $(x_{zMax}, y_{zMax}) = \left(\frac{\alpha d_z^2}{\gamma d_x^2}, \frac{\beta d_z^2}{\gamma d_y^2}\right)$ The extreme value of the function becomes :

$$F_{zMax} = \frac{1}{\varepsilon} \sqrt{\frac{\alpha^2}{d_x^2} + \frac{\beta^2}{dy^2} + \frac{\gamma^2}{dz^2}} - 1 \tag{29}$$

b The extreme is on an edge There are three edges, but we will only present the calculation for $[V_i V_j]$. In this case, a point P belonging to the edge can be represented by $P = aV_i + (1 - a)V_j$, which means:

- $x' = ax'_i + (1 - a)x'_j = (x'_i - x'_j)a + x'_j$
- $y' = ay'_i + (1 - a)y'_j = (y'_i - y'_j)a + y'_j$

The error function becomes:

$$F_{yEdge}(a) = \frac{1}{\varepsilon} \frac{\alpha(a(x'_i - x'_j) + x'_j) + \beta(a(y'_i - y'_j) + y'_j)\gamma}{\sqrt{dx^2(a(x'_i - x'_j) + x'_j)^2 + dy^2(a(y'_i - y'_j) + y'_j)^2 + d_z^2}} - 1 \tag{30}$$

which is extremal for:

$$a_{max} = \frac{-x'_j(\gamma(x'_i - x'_j) + \beta(x'_i y'_j - x'_j y'_i))d_x^2 - y'_j(\gamma(y'_i - y'_j) + \alpha(y'_i x'_j - y'_j x'_i))d_y^2 + (\alpha(x'_i - x'_j) + \beta(y'_i - y'_j))d_z^2}{(x'_i - x'_j)(\gamma(x'_i - x'_j) + \beta(x'_i y'_j - x'_j y'_i))d_x^2 + (y'_i - y'_j)(\gamma(y'_i - y'_j) + \alpha(y'_i x'_j - y'_j x'_i))d_y^2}$$

If $0 \leq a_{max} \leq 1$, the extreme value $F_{xEdgeMax}$ of F_{xEdge} is given by $F_{xEdgeMax}(a_{max})$ whose form is not simple enough to be displayed here.

c The extremum is reached on one of the triangle's vertices V_l, where the relative error is given by:

$$F_{zVertex} = \frac{1}{\varepsilon} \frac{\omega_l}{\sqrt{x_l^2 d_x^2 + y_l^2 d_y^2 + z_l^2 d_z^2}} - 1$$

4.3 Optimization of the error rate

Let be $F_{\mathcal{M}}$, the global error function of the mask generator. For each $(x, y, z) \in \frac{1}{8}\mathbb{Z}^3 = \{(x, y, z) \in \mathbb{Z}^3 \ : \ 0 \leq x, 0 \leq y, 0 \leq z\}$ $F_{\mathcal{M}}$ is the projection of $E_{relative}$ on the plan corresponding to the discrete cone where (x, y, z) lies. The extrema of $F_{\mathcal{M}}$ are obtained by taking the extrema of the projection of the error fuction for each cone of the mask generator. The absolute values of these extrema are called *error rate* and write :

- $\tau_{min} = min(F_p)$
- $\tau_{max} = max(F_p)$
- $\tau = max(|\tau_{min}|, |\tau_{max}|)$

The *error amplitude* is defined as : $amp = |\tau_{max} - \tau_{min}|$.

In the last section, we saw that the use of the scale factor ε is essential to compare a chamfer distance $\frac{d_C}{\varepsilon}$ with the Euclidean distance d_E. We first fixed it to the littlest coefficient corresponding to the horizontal unit displacement.

In [Thi94], E. Thiel computes an optimized scale factor to reduce the error rate τ to the half of the error amplitude. Let us write τ_ε the error rate obtained with a scale ε. He showed that by choosing

$$\varepsilon_{opt} = \varepsilon \left(\frac{|\tau_{min}| + |\tau_{max}|}{2} + 1 \right) \tag{31}$$

we obtain

$$\tau_{opt} = \frac{|\tau_{min}| + |\tau_{max}|}{2} \tag{32}$$

In the following, we will choose the chamfer mask coefficients that minimize τ_{opt}.

5 How to compute automatically chamfer map coefficients ?

The computation of optimal coefficients for a mask of size $(2n + 1)^3$ is done in three steps: generation of the Farey triangulation, generation of the norm constraints, and iterative computation of the optimal sets of weights.

5.1 Building the Farey triangulation

The recursive automated construction of the Farey triangulation of order n is described in appendix A.2.2. This triangulation $\mathcal{T}_C^g$ corresponds to isotropic chamfer mask generator $\mathcal{M}_C^g$. When dealing with anisotropic lattice, one have to add extra vectors to the mask generator and extra cones to the triangulation.

This is achieved by symmetry considerations. For instance, for a 3^3 mask, if the voxel size d_z along z is different from the ones along x and y, d_x and d_y, we have to consider in the mask generator, in addition to the vectors $\{(1,\vec{0},0),(1,\vec{1},0),(1,\vec{1},1)\}$, the two extra vectors $\{(0,\vec{0},1),(1,\vec{0},1)\}$ that corresponds to weights induced by the anisotropy. These extra vectors belongs to the two extra cones, $\langle\langle(1,\vec{0},0),(1,\vec{0},1),(1,\vec{1},1)\rangle\rangle$ and $\langle\langle(0,\vec{0},1),(1,\vec{0},1),(1,\vec{1},1)\rangle\rangle$, that are to be considered for the error computation and the local convexity constraints.

5.2 Generating convexity criteria

The triangulation $\mathcal{T}_C^g$ has been built as described above. It allows us to generate all the local convexity constraints (equation 7) that are to be verified. They have to be generated for every edge inside the mask generator, but also for the edges that are at the border of the mask generator. For the latter, the fourth point (see figure 18) is derived from symmetry considerations.

Please notice that each of the generated LCC depends on 4 weights ω_i.

5.3 Finding the optimal coefficients

This is the tough part. We have to identify the m-tuples $(\omega_1 \ldots \omega_m)$ of weights corresponding to the chamfer mask generator $\mathcal{M}_C^g = \{\vec{v}_i, 1 \leq i \leq m\}$ to find the optimal ones that yield optimal error.

These sets of optimal coefficients are searched by a brute-force method. However, we try to reduce this computationally expensive search by throwing away m-tuples $(\omega_1 \ldots \omega_m)$ as soon as part of them do not satisfy the local convexity constraints (as sketched by below recursive algorithm[1]).

```
1: procedure TEST( n )
2: if some LCCs can be verified with (ω_i, ... ω_n) then
3:    test these LCCs and return if one of them is not verified
4: if n equals to m then %All ω_i are set.
5:    Compute the error τ_opt
6:    if this τ_opt is smaller than the previous one then
7:       (ω_i, ..., ω_m) is an optimal set of coefficients
8:    return
9: for ω_{n+1} from ω_1||v_i||_∞ to ω_1||v_i||_1 do %Iteratively set a value to ω_{n+1}.
10:   TEST( n + 1 )
```

[1] Java code is available from `http://www-sop.inria.fr/epidaure/personnel/Celine.Fouard/`.

11: %**Main Program**
12: **for** ω_1 from 1 to some user provided value **do**
13: TEST(1)

ω_1, the coefficient corresponding to the direction of smallest voxel size, varies from 1 to some maximal value provided by the user, while the other coefficients are searched in the interval $[\omega_1||\vec{v_i}||_\infty, \omega_1||\vec{v_i}||_1]$. Error computation is only performed on coefficients sets that verify all the local convexity constraints. As a result, this algorithm gives all the optimal m-tuples in lexicographical order.

6 Results

6.1 wide isotropic masks

6.1.1 $3 \times 3 \times 3$ isotropic mask

The program gives the following properties of the $3 \times 3 \times 3$ chamfer mask generator:

```
Mask Points:
0: [(1, 0, 0), 0]
1: [(1, 1, 0), 0]
2: [(1, 1, 1), 0]

Mask Cones:
{[(1, 0, 0), 0], [(1, 1, 0), 0], [(1, 1, 1), 0]}

Cones Symetrics
{[(0, 1, 0), 0], [(1, 0, 1), 0], [(1, 1, -1), 0]}
```

These properties can be illustrated by figure 24. The local convexity criteria calculated with the symetrics for each edge of the mask generator are:

```
Local Convexity Criteria
w1 <= 2w0
w0 + w2 <= 2w1
2w1 <= 2w2
```

If we write the chamfer mask generator points with their lexicographic order ($(1,0,0) \longmapsto a$, $(1,1,0) \longmapsto b$ and $(1,1,1) \longmapsto c$), we obtain the following properties:

$$\omega_b \leq 2\omega_a \tag{33}$$
$$\omega_a + \omega_c \leq 2\omega_b \tag{34}$$
$$\omega_b \leq \omega_c \tag{35}$$

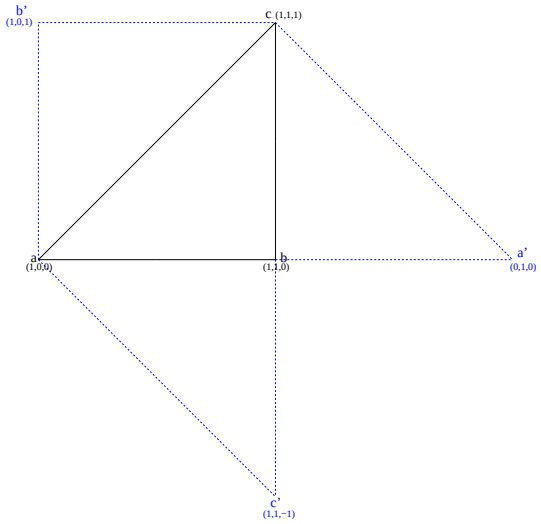


Figure 24: Projection of the geometry of the $3 \times 3 \times 3$ chamfer mask generator

The following table summurize the first optimal $3 \times 3 \times 3$ chamfer mask generator coefficients.

$(1,0,0)$ a	$(1,1,0)$ b	$(1,1,1)$ c	ϵ_{opt}	τ_{opt}
1	1	1	1.211	0.211
1	2	2	1.207	0.207
2	3	3	2.252	0.126
2	3	4	2.225	0.112
3	4	5	3.244	0.081
4	6	7	4.291	0.073
7	10	12	7.473	0.068
11	16	19	11.740	0.067
12	17	21	12.801	0.067
19	27	33	20.235	0.065
22	31	38	23.429	0.065
26	37	45	27.681	0.065
41	58	71	43.629	0.064
123	174	213	130.886	0.064
198	280	343	210.693	0.064
224	317	388	238.359	0.064
239	338	414	254.313	0.064

6.1.2 $3 \times 3 \times 5$ isotropic mask

The program gives the following properties of the $3 \times 3 \times 5$ chamfer mask generator:

```
Mask Points:
0: [(1, 0, 0), 0]
1: [(1, 1, 0), 0]
2: [(1, 1, 1), 0]
3: [(2, 1, 1), 0]

Mask Cones:
{[(1, 0, 0), 0], [(1, 1, 0), 0], [(2, 1, 1), 0]}
{[(2, 1, 1), 0], [(1, 1, 0), 0], [(1, 1, 1), 0]}

Cones Symetrics
{[(1, 1, 1), 0], [(1, 0, 1), 0], [(2, 1, -1), 0]}
{[(1, 2, 1), 0], [(1, 0, 1), 0], [(1, 0, 0), 0]}
```

These properties can be illustrated by figure 25. The local convexity criteria calculated with

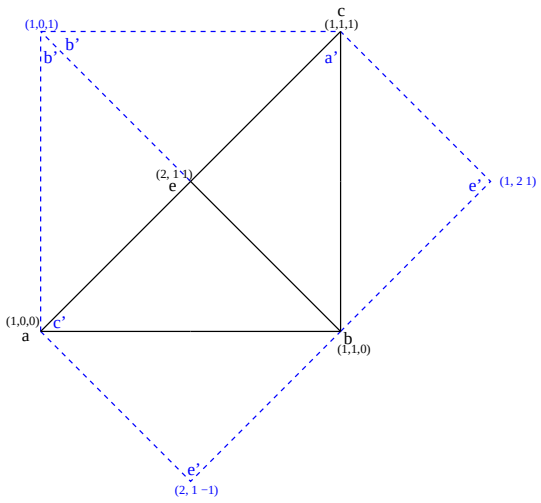


Figure 25: Projection of the geometry of the $3 \times 3 \times 5$ chamfer mask generator

the symetrics for each edge of the mask generator are:

```
Local Convexity Criteria
w3 <= w2 + w0
w3 <= 2w1
```

```
2w0 + 2w1 <= 2w3
w1 + 2w2 <= 2w3
```

The following table summurize the first optimal $3 \times 3 \times 5$ chamfer mask generator coefficients.

$(1,0,0)$ a	$(1,1,0)$ b	$(1,1,1)$ c	$(2,1,1)$ e	ϵ_{opt}	τ_{opt}
1	1	1	2	1.211	0.211
1	2	2	3	1.207	0.207
2	2	3	4	2.293	0.146
2	3	3	5	2.252	0.146
2	3	4	6	2.225	0.112
3	4	5	7	3.167	0.056
5	7	9	13	5.264	0.053
7	10	12	17	7.344	0.049
7	10	13	18	7.341	0.049
10	14	17	24	10.486	0.049
10	14	18	25	10.459	0.046
12	17	21	30	12.519	0.043
17	24	30	42	17.721	0.042
24	34	42	59	25.010	0.042
41	58	71	101	42.719	0.042
41	58	72	101	42.715	0.042
46	65	80	113	47.924	0.042
53	75	92	130	55.210	0.042
58	82	101	142	60.415	0.042
70	99	122	172	72.914	0.042
82	116	143	201	85.410	0.042
99	140	172	243	103.117	0.042
111	157	193	272	115.610	0.042
140	198	243	343	145.811	0.042
169	239	293	414	176.012	0.041

6.1.3 $3 \times 5 \times 5$ isotropic mask

The program gives the following properties of the $3 \times 5 \times 5$ chamfer mask generator:

```
Mask Points:
0: [(1, 0, 0), 0]
1: [(1, 1, 0), 0]
2: [(1, 1, 1), 0]
3: [(2, 1, 1), 0]
```

```
4: [(2, 1, 0), 0]

Mask Cones:
{[(2, 1, 1), 0], [(1, 1, 0), 0], [(1, 1, 1), 0]}
{[(1, 0, 0), 0], [(2, 1, 0), 0], [(2, 1, 1), 0]}
{[(2, 1, 0), 0], [(1, 1, 0), 0], [(2, 1, 1), 0]}

Cones Symetrics
{[(1, 2, 1), 0], [(1, 0, 1), 0], [(2, 1, 0), 0]}
{[(1, 1, 0), 0], [(2, 0, 1), 0], [(2, 1, -1), 0]}
{[(1, 1, 1), 0], [(1, 0, 0), 0], [(2, 1, -1), 0]}
```

These properties can be illustrated by figure 26. The local convexity criteria calculated with

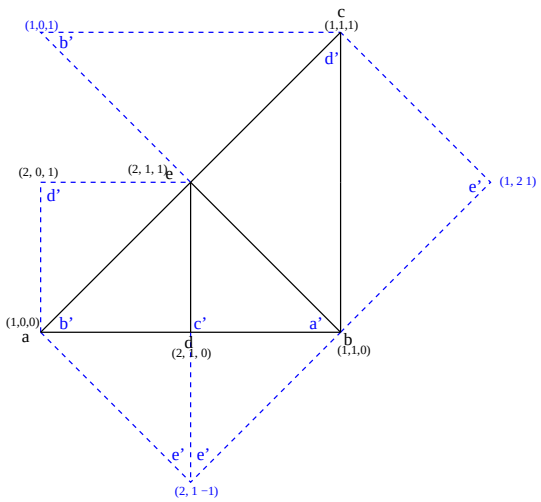


Figure 26: Projection of the geometry of the $3 \times 5 \times 5$ chamfer mask generator

the symetrics for each edge of the mask generator are:

```
Local Convexity Criteria
w1 + 2w2 <= 2w3
w3 <= 2w1
w3 + w1 <= w4 + w2
w4 <= w1 + w0
2w0 + w3 <= 2w4
2w4 <= 2w3
```

%

The following table summurize the first optimal $3 \times 5 \times 5$ chamfer mask generator coefficients.

$(1,0,0)$ a	$(1,1,0)$ b	$(1,1,1)$ c	$(2,1,0)$ d	$(2,1,1)$ e	ϵ_{opt}	τ_{opt}
1	1	1	2	2	1.211	0.211
1	2	2	3	3	1.211	0.211
2	2	3	4	4	2.293	0.146
2	3	3	5	5	2.252	0.126
2	3	4	5	6	2.225	0.112
3	4	5	7	7	3.167	0.056
4	6	7	9	10	4.179	0.045
6	8	10	13	14	6.254	0.042
7	10	12	16	17	7.210	0.030
10	14	17	22	24	10.297	0.030
17	24	29	38	41	17.491	0.029
19	27	33	43	47	19.525	0.028
22	31	38	49	54	22.597	0.027
26	47	45	59	64	26.703	0.027
29	41	50	65	71	29.781	0.027
34	48	59	76	83	34.897	0.026
41	58	71	92	101	42.080	0.026
46	65	80	103	113	47.195	0.026
53	75	92	119	130	54.348	0.025
60	85	104	135	147	61.520	0.025
63	89	109	141	154	64.591	0.025
82	116	142	184	201	84.058	0.025
116	164	201	260	284	118.895	0.025
145	205	251	324	355	148.614	0.025
198	280	343	443	485	202.900	0.025

6.1.4 $5 \times 5 \times 5$ isotropic mask

The program gives the following properties of the $5 \times 5 \times 5$ chamfer mask generator:

```
Mask Points:
0: [(1, 0, 0), 0]
1: [(1, 1, 0), 0]
```

```
2: [(1, 1, 1), 0]
3: [(2, 1, 1), 0]
4: [(2, 1, 0), 0]
5: [(2, 2, 1), 0]

Mask Cones:
{[(1, 0, 0), 0], [(2, 1, 0), 0], [(2, 1, 1), 0]}
{[(2, 1, 0), 0], [(1, 1, 0), 0], [(2, 1, 1), 0]}
{[(2, 1, 1), 0], [(1, 1, 0), 0], [(2, 2, 1), 0]}
{[(2, 1, 1), 0], [(2, 2, 1), 0], [(1, 1, 1), 0]}

Cones Symetrics
{[(1, 1, 0), 0], [(2, 0, 1), 0], [(2, 1, -1), 0]}
{[(2, 2, 1), 0], [(1, 0, 0), 0], [(2, 1, -1), 0]}
{[(1, 2, 1), 0], [(1, 1, 1), 0], [(2, 1, 0), 0]}
{[(1, 2, 1), 0], [(2, 1, 2), 0], [(1, 1, 0), 0]}
```

These properties can be illustrated by figure 27. The local convexity criteria calculated with

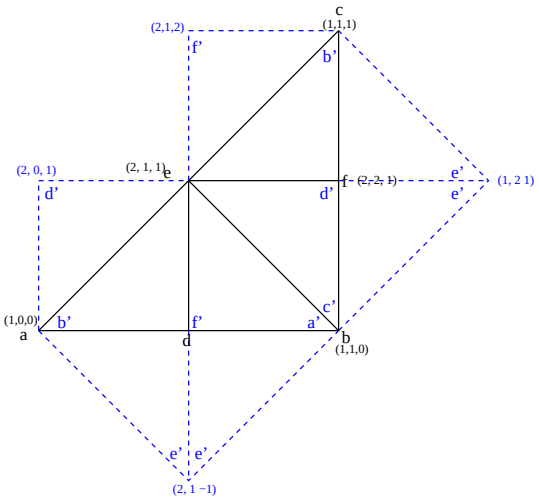


Figure 27: Projection of the geometry of the $5 \times 5 \times 5$ chamfer mask generator

the symetrics for each edge of the mask generator are:

```
Local Convexity Criteria
w4 <= w1 + w0
2w0 + w3 <= 2w4
```

```
2w4 <= 2w3
2w1 + w3 <= w5 + w4
2w4 <= 2w3
2w5 <= 2w3 + w1
w5 <= w2 + w1
w5 + w2 <= 2w3
w3 + 2w2 <= 2w5
```

The following table summurize the first optimal $5 \times 5 \times 5$ chamfer mask generator coefficients.

$(1,0,0)$ a	$(1,1,0)$ b	$(1,1,1)$ c	$(2,1,0)$ d	$(2,1,1)$ e	$(2,2,1)$ f	ϵ_{opt}	τ_{opt}
1	1	1	2	2	2	1.211	0.211
1	2	2	3	3	4	1.207	0.207
2	2	3	4	4	5	1.207	0.207
2	3	3	5	5	6	2.252	0.126
2	3	4	5	6	7	2.225	0.112
3	4	5	7	7	9	3.167	0.056
4	6	7	9	10	13	4.179	0.045
5	7	9	11	12	15	5.149	0.030
9	13	16	20	22	28	9.245	0.027
11	16	20	25	27	34	11.288	0.026
20	29	35	45	49	62	20.500	0.025
29	41	51	65	71	88	29.722	0.025
49	70	85	110	120	150	50.220	0.025
69	98	120	155	169	210	70.718	0.025
71	101	123	159	174	217	72.766	0.025
80	114	139	179	196	245	81.982	0.025
89	126	155	199	218	271	91.201	0.025

Table 2:

6.1.5 $7 \times 7 \times 7$ isotropic mask

The program gives the following properties of the $7 \times 7 \times 7$ chamfer mask generator:

```
Mask Points:
0: [(1, 0, 0), 0]
1: [(1, 1, 0), 0]
```

```
2: [(1, 1, 1), 0]
3: [(2, 1, 1), 0]
4: [(2, 1, 0), 0]
5: [(2, 2, 1), 0]
6: [(3, 1, 1), 0]
7: [(3, 2, 1), 0]
8: [(3, 3, 1), 0]
9: [(3, 3, 2), 0]
10: [(3, 1, 0), 0]
11: [(3, 2, 0), 0]
12: [(3, 2, 2), 0]

Mask Cones:
{[(3, 1, 1), 0], [(2, 1, 0), 0], [(2, 1, 1), 0]}
{[(2, 1, 0), 0], [(3, 2, 1), 0], [(2, 1, 1), 0]}
{[(2, 1, 1), 0], [(3, 3, 1), 0], [(2, 2, 1), 0]}
{[(2, 1, 1), 0], [(2, 2, 1), 0], [(3, 3, 2), 0]}
{[(1, 0, 0), 0], [(3, 1, 0), 0], [(3, 1, 1), 0]}
{[(3, 1, 0), 0], [(2, 1, 0), 0], [(3, 1, 1), 0]}
{[(2, 1, 0), 0], [(3, 2, 0), 0], [(3, 2, 1), 0]}
{[(3, 2, 0), 0], [(1, 1, 0), 0], [(3, 2, 1), 0]}
{[(2, 1, 1), 0], [(3, 2, 1), 0], [(3, 3, 1), 0]}
{[(3, 2, 1), 0], [(1, 1, 0), 0], [(3, 3, 1), 0]}
{[(2, 1, 1), 0], [(3, 3, 2), 0], [(3, 2, 2), 0]}
{[(3, 2, 2), 0], [(3, 3, 2), 0], [(1, 1, 1), 0]}

Cones Symetrics
{[(3, 2, 1), 0], [(2, 0, 1), 0], [(3, 1, 0), 0]}
{[(3, 3, 1), 0], [(3, 1, 1), 0], [(3, 2, 0), 0]}
{[(1, 2, 1), 0], [(3, 3, 2), 0], [(3, 2, 1), 0]}
{[(1, 2, 1), 0], [(3, 2, 2), 0], [(3, 3, 1), 0]}
{[(2, 1, 0), 0], [(3, 0, 1), 0], [(3, 1, -1), 0]}
{[(2, 1, 1), 0], [(1, 0, 0), 0], [(3, 1, -1), 0]}
{[(1, 1, 0), 0], [(2, 1, 1), 0], [(3, 2, -1), 0]}
{[(3, 3, 1), 0], [(2, 1, 0), 0], [(3, 2, -1), 0]}
{[(1, 1, 0), 0], [(2, 2, 1), 0], [(2, 1, 0), 0]}
{[(2, 3, 1), 0], [(2, 1, 1), 0], [(3, 2, 0), 0]}
{[(1, 1, 1), 0], [(3, 2, 3), 0], [(2, 2, 1), 0]}
{[(2, 3, 2), 0], [(3, 2, 3), 0], [(2, 1, 1), 0]}
```

These properties can be illustrated by figure 28. The local convexity criteria calculated with the symetrics for each edge of the mask generator are:

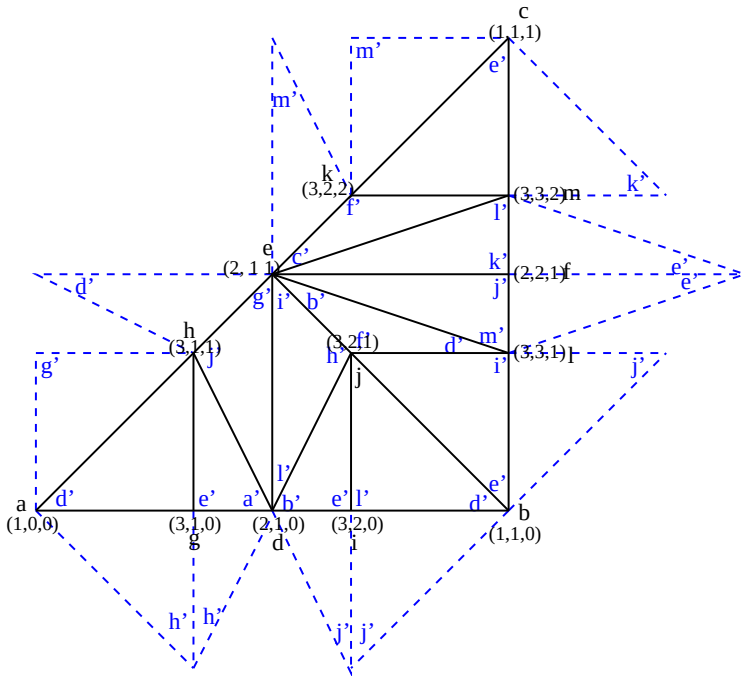


Figure 28: Projection of the geometry of the $7 \times 7 \times 7$ chamfer mask generator

```
Local Convexity Criteria
w4 + 2w3 <= w7 + w6
2w6 <= 2w4 + w3
w6 + w4 <= w10 + w3
3w7 <= w8 + w4 + 2w3
w4 + w7 <= w11 + w3
3w5 <= 2w3 + w8
3w5 <= w9 + w8
w3 + w8 <= w7 + w5
w9 <= 2w3
w3 + w9 <= w12 + w5
w10 <= w4 + w0
3w0 + w6 <= 2w10
2w10 <= 2w6
2w10 <= 2w6
w11 <= w1 + w4
2w11 <= 2w7
3w1 + w7 <= w8 + w11
```

```
2w11 <= 2w7
w7 <= w1 + w3
2w8 <= 2w7 + w1
w12 <= w2 + w3
4w12 <= 3w3 + 2w9
w9 + 2w2 <= 2w12
w12 + 3w2 <= 2w9
```

The following 3 summurize the first optimal $7 \times 7 \times 7$ chamfer mask generator coefficients.

a	b	c	d	e	f	g	h	i	j	k	l	m	ϵ_{opt}	τ_{opt}
1	1	1	2	2	2	3	3	3	3	3	3	3	1.211	0.211
1	2	2	3	3	4	4	4	5	5	5	6	6	1.207	0.207
2	2	3	4	4	5	6	6	6	6	7	7	8	2.293	0.146
2	3	3	5	5	6	7	7	8	8	8	9	9	2.252	0.126
2	3	4	5	6	7	7	8	8	9	10	10	11	2.225	0.112
3	4	5	6	7	9	9	9	10	11	12	13	14	3.158	0.053
4	6	7	9	10	13	13	14	15	16	17	19	20	4.179	0.045
5	7	9	11	12	15	16	16	18	19	21	22	24	5.186	0.037
5	7	9	11	12	15	16	17	18	19	21	22	24	5.149	0.030
7	10	12	16	17	21	22	23	26	27	29	31	33	7.176	0.025
8	11	14	18	19	24	25	26	29	30	33	34	38	8.184	0.023
10	14	17	22	24	30	32	33	36	37	41	43	47	10.224	0.022
11	16	19	25	27	34	35	37	41	42	46	49	53	11.238	0.022
12	17	21	27	29	36	38	40	44	45	49	52	56	12.245	0.020
14	20	24	31	34	34	44	46	51	53	58	62	67	14.248	0.018
17	24	30	38	42	52	54	57	62	65	71	75	81	17.275	0.016

Table 3:

6.2 anisotropic masks

Table 4 presents optimal sets of weights of a $3 \times 3 \times 3$ chamfer mask for an anisotropic grid with $dx = 1, dy = 1.5, dz = 3.0$. The points belonging to this mask are: $aX(1,0,0)$, $aY(0,1,0)$, $aZ(0,0,1)$, $bYZ(0,1,1)$, $bXZ(1,0,1)$, $bXY(1,1,0)$, and $c(1,1,1)$.

```
Mask Points:
0: [(1, 0, 0), 0]
1: [(0, 1, 0), 0]
2: [(0, 0, 1), 0]
3: [(0, 1, 1), 0]
4: [(1, 0, 1), 0]
```

```
5: [(1, 1, 0), 0]
6: [(1, 1, 1), 0]

Mask Cones:
{[(1, 0, 0), 0], [(1, 1, 0), 0], [(1, 1, 1), 0]}
{[(1, 0, 0), 0], [(1, 1, 1), 0], [(1, 0, 1), 0]}
{[(0, 0, 1), 0], [(1, 0, 1), 0], [(1, 1, 1), 0]}
{[(0, 0, 1), 0], [(1, 1, 1), 0], [(0, 1, 1), 0]}
{[(0, 1, 1), 0], [(1, 1, 1), 0], [(0, 1, 0), 0]}
{[(0, 1, 0), 0], [(1, 1, 1), 0], [(1, 1, 0), 0]}

Cones Symetrics:
{[(0, 1, 0), 0], [(1, 0, 1), 0], [(1, 1, -1), 0]}
{[(0, 0, 1), 0], [(1, -1, 1), 0], [(1, 1, 0), 0]}
{[(1, 0, 0), 0], [(0, 1, 1), 0], [(1, -1, 1), 0]}
{[(0, 1, 0), 0], [(-1, 1, 1), 0], [(1, 0, 1), 0]}
{[(1, 1, 0), 0], [(-1, 1, 1), 0], [(0, 0, 1), 0]}
{[(1, 0, 0), 0], [(1, 1, -1), 0], [(0, 1, 1), 0]}

Local Convexity Criteria:
w5 <= w1 + w0
w0 + w6 <= w4 + w5
2w5 <= 2w6
w4 <= w2 + w0
2w4 <= 2w6
w2 + w6 <= w3 + w4
w3 <= w1 + w2
2w3 <= 2w6
w6 + w1 <= w5 + w3
```

7 Conclusion

We have proposed an automated approach to compute optimal chamfer norm coefficients for mask of any size and for lattice of any anisotropy. It is based on the Farey triangulation that permits us to recursively build large masks while ensuring a regular triangulation of the chamfer mask generators. It allows us to automatically compute the error of any mask, thanks to analytical expressions of errors we can derive on regular cones. In addition, the coefficients we calculate verify norm constraints, thus yields scale invariant chamfer maps.

aX	aY	aZ	bYZ	bXZ	bXY	c	ε_{opt}	$\tau_{opt}(\%)$
1	2	3	3	3	2	3	1.257	25.66
1	2	3	4	4	2	4	1.238	23.79
2	3	6	6	6	3	6	2.370	18.49
2	3	6	7	6	4	7	2.353	17.65
2	3	6	7	7	4	7	2.302	15.09
4	6	12	13	12	7	13	4.592	14.81
4	6	12	13	13	7	14	4.584	14.60
4	6	12	14	13	7	14	4.581	14.52
5	8	15	17	16	9	17	5.703	14.06
6	9	18	20	19	11	21	6.834	13.90
6	9	18	21	19	11	21	6.815	13.59
10	15	30	34	32	18	35	11.343	13.43

Table 4: $3 \times 3 \times 3$ chamfer mask coefficients for anisotropic grid.

A Farey Series/Sets

A.1 Farey Series

A.1.1 Farey Series Points

DEFINITION **A.1 (Farey Series)**
A Farey series $\mathcal{F}_n$ *of level* n *is an increasing sequence of irreducible fractions between* 0 *and* 1 *and whose denominator does not exceed* n*. i.e.:*

$$\frac{y}{x} \in \mathcal{F}_n \; iff \begin{cases} 0 \leq y \leq x \leq n & and \\ pgcd(y,x) = 1 \end{cases}$$

0 and 1 are included in $\mathcal{F}_n$ as $\frac{0}{1}$ and $\frac{1}{1}$.
For example the Farey series at the level 5 writes:

$$\mathcal{F}_5 = \left\{ \frac{0}{1}, \frac{1}{5}, \frac{1}{4}, \frac{1}{3}, \frac{2}{5}, \frac{1}{2}, \frac{3}{5}, \frac{2}{3}, \frac{3}{4}, \frac{4}{5}, \frac{1}{1} \right\}$$

The Farey series elements verify the following properties (see [HW78] for demonstration):

- If $\frac{y}{x}$ and $\frac{y'}{x'}$ are two successive terms of $\mathcal{F}_n$, then $xy' - yx' = 1$.
- If $\frac{y}{x}$, $\frac{y''}{x''}$ and $\frac{y'}{x'}$ are tree successive terms of $\mathcal{F}_n$, then: $\frac{y''}{x''} = \frac{y+y'}{x+x'}$.
- If $\frac{y}{x}$ and $\frac{y'}{x'}$ are two successive terms of $\mathcal{F}_n$, then:
  - $x + x' > n$

– The median $\frac{y+y'}{x+x'}$ of $\frac{y}{x}$ and $\frac{y'}{x'}$ is in the interval $[\frac{y}{x},\ \frac{y'}{x'}]$.

We can thus build $\mathcal{F}_{n+1}$ from $\mathcal{F}_n$ by inserting the new term $\frac{y+y'}{x+x'}$ in each couple $(\frac{y}{x},\ \frac{y'}{x'})$ if $x + x' \leq n + 1$. For example, starting form $\mathcal{F}_1 = \{\frac{0}{1}, \frac{1}{1}\}$, we can build $\mathcal{F}_2 = \{\frac{0}{1}, \frac{1}{2}, \frac{1}{1}\}$ by adding the term $\frac{0+1}{1+1} = \frac{1}{2}$ between $\frac{0}{1}$ and $\frac{1}{1}$. And so on... Here are the first Farey series:
$\mathcal{F}_1 = \{\frac{0}{1},\ \frac{1}{1}\}$
$\mathcal{F}_2 = \{\frac{0}{1},\ \frac{1}{2},\ \frac{1}{1}\}$
$\mathcal{F}_3 = \{\frac{0}{1},\ \frac{1}{3},\ \frac{1}{2},\ \frac{2}{3},\ \frac{1}{1}\}$
$\mathcal{F}_4 = \{\frac{0}{1},\ \frac{1}{4},\ \frac{1}{3},\ \frac{1}{2},\ \frac{2}{3},\ \frac{3}{4},\ \frac{1}{1}\}$
$\mathcal{F}_5 = \{\frac{0}{1},\ \frac{1}{5},\ \frac{1}{4},\ \frac{1}{3},\ \frac{2}{5},\ \frac{1}{2},\ \frac{3}{5},\ \frac{2}{3},\ \frac{3}{4},\ \frac{4}{5},\ \frac{1}{1}\}$
$\mathcal{F}_6 = \{\frac{0}{1},\ \frac{1}{6},\ \frac{1}{5},\ \frac{1}{4},\ \frac{1}{3},\ \frac{2}{5},\ \frac{1}{2},\ \frac{3}{5},\ \frac{2}{3},\ \frac{3}{4},\ \frac{4}{5},\ \frac{5}{6},\ \frac{1}{1}\}$
We can see these fractions $\frac{y}{x}$ as the coordinates of a point (x, y) of the fundamental network of $\mathbb{Z}^2$. Figure 29 shows the representation of the 2D points of $\mathcal{F}_6$.

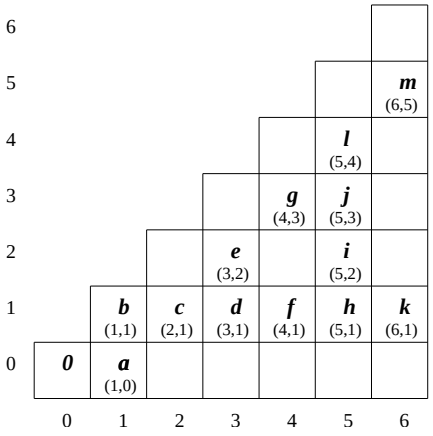


Figure 29: first 2D visible points

DEFINITION **A.2 (Visible point)**
A point $P(x, y) \in \mathbb{Z}^2$ is said to be visible *from the origin, if there is no point on the fundamental network of $\mathbb{Z}^2$ located on (OP) between O and P. A necessary and sufficient condition for P to be visible is $gcd(x, y) = 1$. (see [HW78] for a demonstration).*

Let us write a, b, c, ... the visible points in the first eighth of the plan, ordered by their distance to the origin. They are represented in figure 29. Now if we order the first visible points by their angle to the horizontal direction, we obtain:

$$a(1,0),\ k(6,1),\ h(5,1),\ f(4,1),\ d(3,1),\ i(5,2),\ c(2,1),\ j(5,3),\ e(3,2),\ g(4,3),\ l(5,4),\ m(6,5),\ b(1,1) \tag{36}$$

which exactly correspond with:

$$\mathcal{F}_6 = \left\{ \frac{0}{1}, \frac{1}{6}, \frac{1}{5}, \frac{1}{4}, \frac{1}{3}, \frac{2}{5}, \frac{1}{2}, \frac{3}{5}, \frac{2}{3}, \frac{3}{4}, \frac{4}{5}, \frac{5}{6}, \frac{1}{1} \right\} \tag{37}$$

Theoreme A.3
Let be V_n the set of visible points $M_i(x_i, y_i)$ ordered by their angle in relation to the horizontal direction, and such that $x_i \leq n$. Then, the rational numbers $\frac{y_i}{x_i}$ are exactly the Farey series of level n.
(see [Thi94] for the demonstration).
We now have a mean to build the visible points which will be used in a chamfer mask, by building the corresponding Farey series.

A.1.2 Farey Series Triangulation

The triangulation formed with the Farey points in the increasing order (see (36) and (37) for examples) is regular. Let $\frac{y_i}{x_i}$ and $\frac{y_j}{x_j}$ be two successive rational numbers of a Farey series, and $\overrightarrow{v_i}$ and $\overrightarrow{v_j}$ the corresponding 2D vectors. Then, the discrete cone $\langle\langle \overrightarrow{v_i}, \overrightarrow{v_j} \rangle\rangle$ is regular. Indeed,

$$\Delta_{i,j} = \begin{vmatrix} x_i & x_j \\ y_i & y_j \end{vmatrix} = x_i y_j - y_i x_j = 1$$

(see the properties of Farey series).

A.2 Farey Sets

Farey series can be extended to 3D: they are called Farey sets.

A.2.1 Farey Sets Points

DEFINITION **A.4 (Farey Set)**
A Farey set $\widehat{\mathcal{F}_n}$ *of level n is a set of irreducible points $(\frac{y}{x}, \frac{z}{x})$ in $[0,1]^2$ whose denominator does net exceed n. i.e.:*

$$\left(\frac{y}{x}, \frac{z}{x}\right) \in \widehat{\mathcal{F}_n} \; iff \begin{cases} x \leq n & and \\ 0 \leq y \leq x & and \\ 0 \leq z \leq x & and \\ pgcd(x, y, z) = 1 & \end{cases}$$

DEFINITION **A.5** $(\widehat{+})$
Let $(\frac{y}{x}, \frac{z}{x})$ and $(\frac{y'}{x'}, \frac{z'}{x'})$ be two points of $\widehat{\mathcal{F}_n}$. $\widehat{+}$ is defined by:

$$\left(\frac{y}{x}, \frac{z}{x}\right) \widehat{+} \left(\frac{y'}{x'}, \frac{z'}{x'}\right) = \left(\frac{y+y'}{x+x'}, \frac{z+z'}{x+x'}\right)$$

The Farey set elements verify the following properties:
Let $\widehat{Q}, \widehat{R}, \widehat{S}$ be 3 points of the Farey set $\widehat{\mathcal{F}_n}$. If $(\widehat{Q}, \widehat{R}, \widehat{S})$ is a regular triangle of $\mathcal{F}_n$, and $\widehat{P} \in \mathcal{F}_{n+1}$ such that $\widehat{P} \subset (\widehat{Q}, \widehat{R}, \widehat{S})$, then

$$\begin{cases} \widehat{P} = \widehat{Q} \hat{+} \widehat{R} & or \\ \widehat{P} = \widehat{R} \hat{+} \widehat{S} & or \\ \widehat{P} = \widehat{S} \hat{+} \widehat{Q} \end{cases}$$

We can thus build $\widehat{\mathcal{F}_{n+1}}$ from $\widehat{\mathcal{F}_n}$ in the same way than for the Farey series. Here are some examples of Farey sets:
$\widehat{\mathcal{F}_1} = \{(\frac{0}{1}, \frac{0}{1}), (\frac{0}{1}, \frac{1}{1}), (\frac{1}{1}, \frac{1}{1})\}$.
$\widehat{\mathcal{F}_2} = \{(\frac{0}{1}, \frac{0}{1}), (\frac{0}{1}, \frac{1}{1}), (\frac{1}{1}, \frac{1}{1}), (\frac{0}{2}, \frac{1}{2}), (\frac{1}{2}, \frac{1}{2}), (\frac{1}{2}, \frac{2}{2})\}$
$\widehat{\mathcal{F}_3} = \{(\frac{0}{1}, \frac{0}{1}), (\frac{0}{1}, \frac{1}{1}), (\frac{1}{1}, \frac{1}{1}), (\frac{0}{2}, \frac{1}{2}), (\frac{1}{2}, \frac{1}{2}), (\frac{1}{2}, \frac{2}{2}), (\frac{0}{3}, \frac{1}{3}), (\frac{1}{3}, \frac{1}{3}), (\frac{1}{3}, \frac{2}{3}), (\frac{0}{3}, \frac{2}{3}), (\frac{1}{3}, \frac{3}{3}), (\frac{2}{3}, \frac{2}{3}), (\frac{2}{3}, \frac{3}{3})\}$
Figure 30 shows a spatial representation of the Farey set $\widehat{\mathcal{F}_3}$. By definition of the Farey

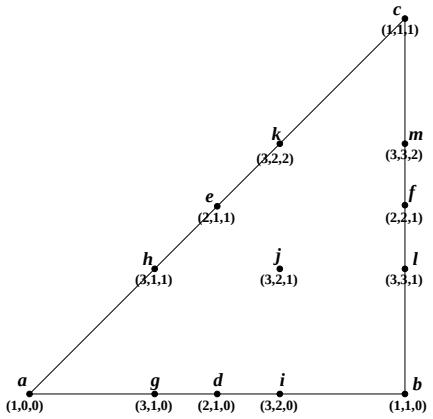


Figure 30: Spatial representation of $\widehat{\mathcal{F}_3}$

sets, Farey sets points are 3D visible points ($pgcd(x, y, z) = 1$).

A.2.2 Farey Sets Triangulation

Here we expose an automatic method to build regular triangulation from Farey sets points.

First step
We start from the cone $\langle\langle \overrightarrow{a}, \overrightarrow{b}, \overrightarrow{c} \rangle\rangle$ with $\overrightarrow{a}(1,0,0)$, $\overrightarrow{b}(1,1,0)$, and $\overrightarrow{c}(1,1,1)$, build with

the points of $\widehat{\mathcal{F}_1}$ (see figure 31 (a)). This cone is regular:

$$\Delta_{\overrightarrow{a},\overrightarrow{b},\overrightarrow{c}} = \begin{vmatrix} 1 & 1 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{vmatrix} = 1.$$

From n to $n+1$
Then, for each direct triangle $(\widehat{A}\widehat{B}\widehat{C})$ of the regular triangulation of $\widehat{\mathcal{F}_n}$, we choose the an edge, for example $[AC]$. We then calculate

$$\widehat{B'} = \widehat{A}\widehat{+}\widehat{C} : \begin{cases} x_{\widehat{B'}} &= x_{\widehat{A}} + x_{\widehat{C}} \\ y_{\widehat{B'}} &= y_{\widehat{A}} + y_{\widehat{C}} \\ z_{\widehat{B'}} &= z_{\widehat{A}} + z_{\widehat{C}} \end{cases}$$

If $x_{\widehat{B'}} > n+1$, we mark the triangle as already done, and we choose another triangle of the triangulation. Else, we obtain 2 new direct triangles $(\widehat{A}\widehat{B}\widehat{B'})$ and $(\widehat{B'}\widehat{B}\widehat{C})$. These triangles are regular:

$$\begin{aligned} \Delta_{\widehat{A}\widehat{B}\widehat{B'}} &= \begin{vmatrix} x_{\widehat{A}} & x_{\widehat{B}} & x_{\widehat{B'}} \\ y_{\widehat{A}} & y_{\widehat{B}} & y_{\widehat{B'}} \\ z_{\widehat{A}} & z_{\widehat{B}} & z_{\widehat{B'}} \end{vmatrix} = \begin{vmatrix} x_{\widehat{A}} & x_{\widehat{B}} & x_{\widehat{A}} + x_{\widehat{C}} \\ y_{\widehat{A}} & y_{\widehat{B}} & y_{\widehat{A}} + y_{\widehat{C}} \\ z_{\widehat{A}} & z_{\widehat{B}} & z_{\widehat{A}} + z_{\widehat{C}} \end{vmatrix} \\ &= x_A y_B (z_A + z_C) + x_B (y_A + y_C) z_A + (x_A + x_C) y_A z_B \\ &\quad -(x_A + x_C) y_B z_A - x_B y_A (z_A + z_C) - x_A (y_A + y_C) z_B \\ &= x_A y_B z_C + x_B y_C z_A + x_C y_A z_B - x_C y_B z_A - x_B y_A z_C - x_A y_C z_B \\ &= \Delta_{\widehat{A}\widehat{B}\widehat{C}} \\ &= 1 \end{aligned}$$

In the same way, we obtain $\Delta_{\widehat{B'}\widehat{B}\widehat{C}} = \Delta_{\widehat{A}\widehat{B}\widehat{C}} = 1$. We repeat the operation with the new triangles until we do not find any point whose x-coordinate is lower or equal to $n+1$. We obtain the points and a regular triangulation of $\widehat{\mathcal{F}_{n+1}}$.

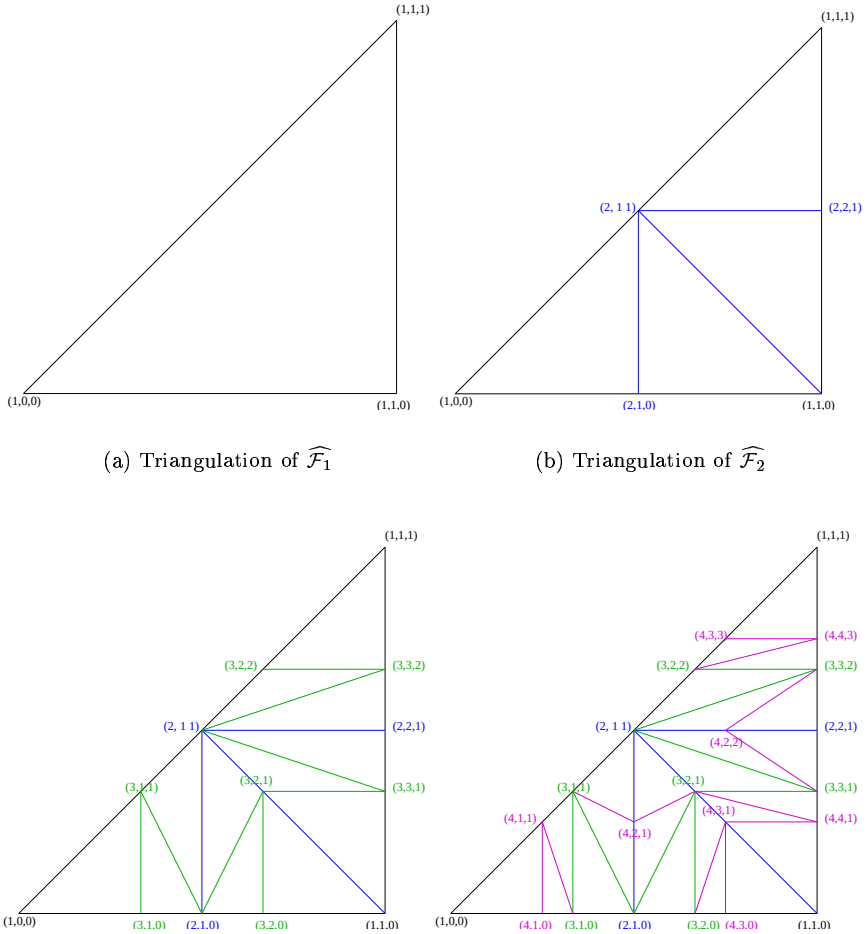
Figure 31 shows the projection on the plan $x = 1$ of the Farey triangulation from $\widehat{\mathcal{F}_1}$ to $\widehat{\mathcal{F}_4}$. For each step we compute the median of the largest edge of the triangle.

If we compute the points and this triangulation of $\widehat{\mathcal{F}_n}$, we obtain the points and the triangulation of a mask generator for a $(2n+1) \times (2n+1) \times (2n+1)$ mask.

B References

[Bor84] G. Borgefors. Distance transformations in arbitrary dimensions. *Computer Vision, Graphics, and Image Processing*, 27:321–345, February 1984.

[Bor86] G. Borgefors. Distance transformations in digital images. *Computer Vision, Graphics, and Image Processing*, 34(3):344–371, February 1986.



(a) Triangulation of $\widehat{\mathcal{F}_1}$

(b) Triangulation of $\widehat{\mathcal{F}_2}$

(c) Triangulation of $\widehat{\mathcal{F}_3}$

(d) Triangulation of $\widehat{\mathcal{F}_4}$

Figure 31: Projection of Farey Triangulation on the plan $x = 1$

[Bor96] G. Borgefors. On digital distance transforms in three dimensions. *Computer Vision and Image Understanding*, 64(3):368–376, November 1996.

[Che97] Y. Chehadeh. *Opérateurs locaux de distance en maillages rectangulaires et parallélepipédique : application à l'analyse d'images*. PhD thesis, Université de Savoie, 1997.

[Dan80] P.E. Danielsson. Euclidean distance mapping. *Computer Graphics and Image Processing*, 1980.

[HM94] C.T. Huang and O.R. Mitchel. A euclidean distance transform using grayscale morphology decomposition. *IEEE transactions on Pattern Analysis and Machine Intelligence*, 1994.

[HW78] G.H. Hardy and E.M. Wright. *An Introduction to the Theory of Numbers.* Oxford University Press, 1978.

[HZB92] G.T. Herman, J. Zheng, and C.A. Bucholtz. Shape-based interpolation. *IEEE Computer Graphics & Applications*, pages 69–79, May 1992.

[MBLKF94] J.F. Mangin, I. Bloch, J. López-Krahe, and V. Frouin. Chamfer distances in anisotropic 3D images. In *VII European Signal Processing Conference*, Edimburgh, UK, September 1994.

[Pud98] C.J. Pudney. Distance-ordered homotopic thinning: A skeletonization algorithm for 3d digital images. *Computer Vision and Image Understanding*, 72(3):404–413, December 1998.

[Rag93] I. Ragnemalm. The euclidean distance transform in arbitrary dimensions. *Pattern Recognition Letters*, 14(11):883–888, November 1993.

[Rem00] E. Remy. Optimizing 3d chamfer masks with norm constraints. In *International Workshop on Combinatorial Image Analysis*, pages 39–56, July 2000.

[Rem01] E. Remy. *Normes de chanfrein et axe médian dans le volume discret.* PhD thesis, Université de la Méditerranée, Marseille, France, December 2001.

[RP66] A. Rosenfeld and J.L. Pfaltz. Sequential operations in digital picture processing. *Journal of the Association for Computing Machinery*, 13(4):471–494, October 1966.

[SB02] I.M. Sintorn and G. Borgefors. Weighted distance transfoms for images using elongated voxel grids. In A. Braquelaire, J.-O. Lachaud, and A. Vialard, editors, *Discrete Geometry for Compueter Imagery (DGCI)*, volume 2301 of *LNCS*, pages 244–254. Springer, 2002.

[SM92] F.Y Shih and O.R. Mitchell. A mathematical morphology approach to euclidean distance transformation. *IEEE Transactions on Image Processing*, 1992.

[Thi94] E. Thiel. *Les distances de chanfrein en analyse d'images : fondements et applications.* PhD thesis, Université Joseph Fourier, Septembre 1994.

[Ver91] B.J.H Verwer. Local distances for distance transformations in two and three dimensions. *Pattern Recognition Letters*, 12:671–682, November 1991.

Unité de recherche INRIA Sophia Antipolis
2004, route des Lucioles - BP 93 - 06902 Sophia Antipolis Cedex (France)

Unité de recherche INRIA Futurs : Domaine de Voluceau - Rocquencourt - BP 105 - 78153 Le Chesnay Cedex (France)
Unité de recherche INRIA Lorraine : LORIA, Technopôle de Nancy-Brabois - Campus scientifique
615, rue du Jardin Botanique - BP 101 - 54602 Villers-lès-Nancy Cedex (France)
Unité de recherche INRIA Rennes : IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex (France)
Unité de recherche INRIA Rhône-Alpes : 655, avenue de l'Europe - 38334 Montbonnot Saint-Ismier (France)
Unité de recherche INRIA Rocquencourt : Domaine de Voluceau - Rocquencourt - BP 105 - 78153 Le Chesnay Cedex (France)

Éditeur
INRIA - Domaine de Voluceau - Rocquencourt, BP 105 - 78153 Le Chesnay Cedex (France)
http://www.inria.fr
ISSN 0249-6399