Second Order Geometric Distance Fields

Anna Lili Horváth

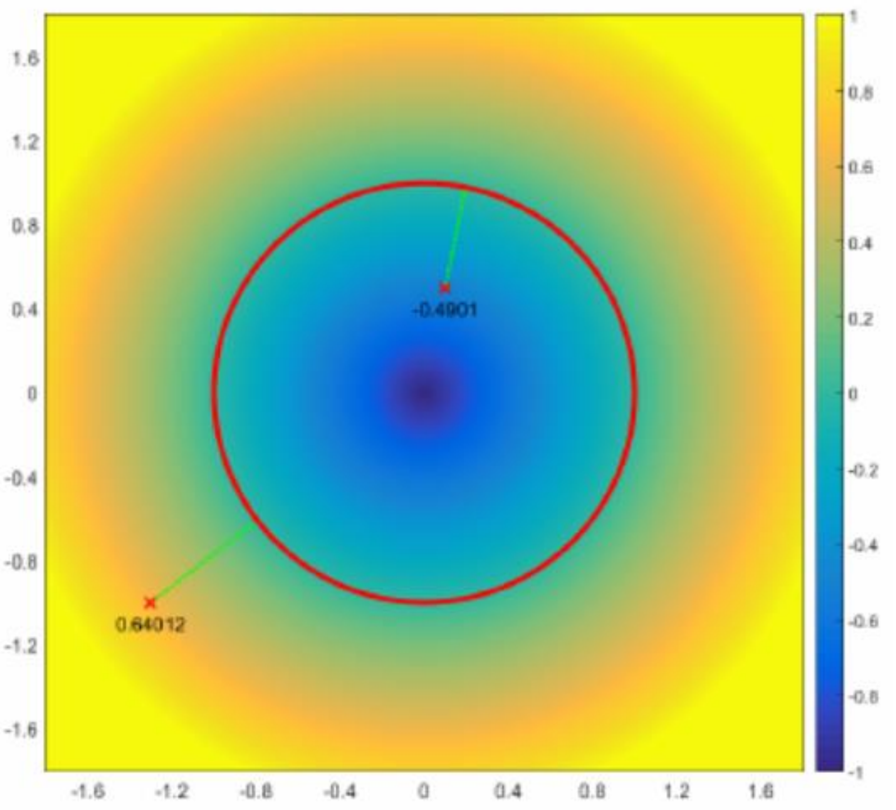
Eötvös Loránd University

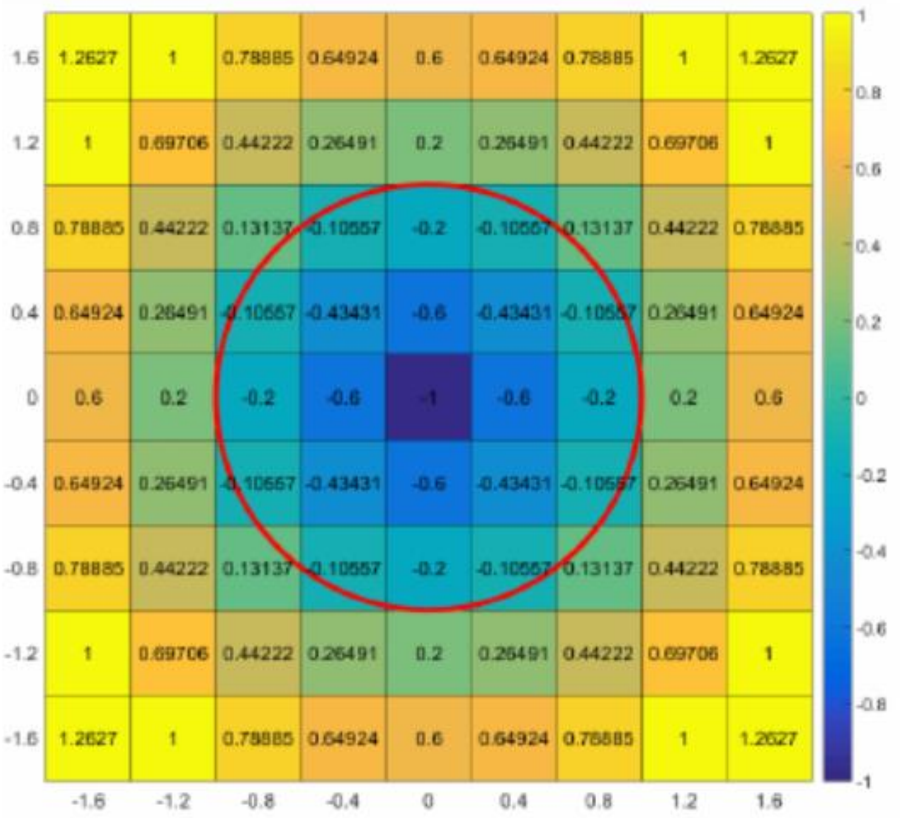
Róbert Bán, Gábor Valasek

CGTA 2023

Distance fields

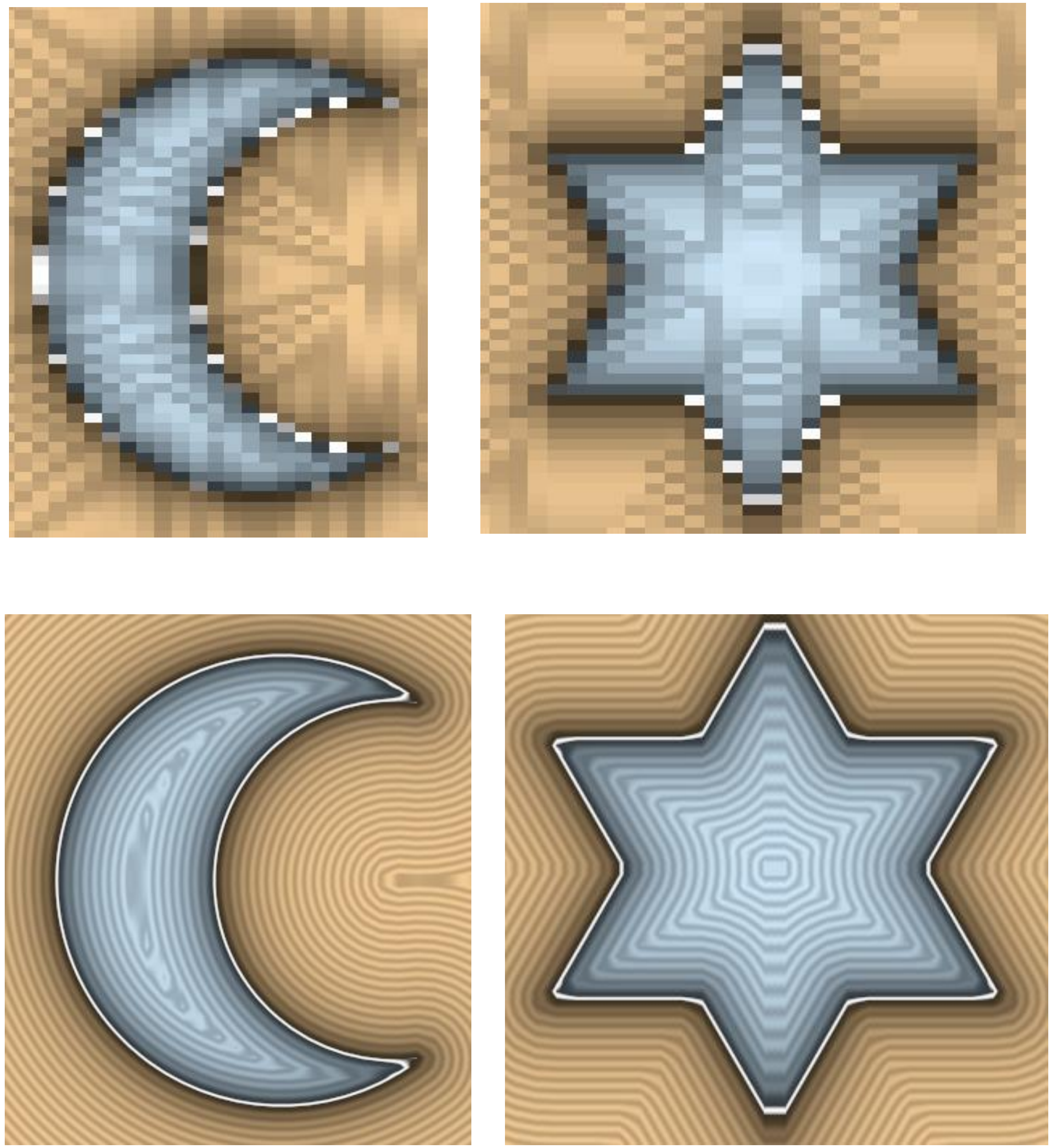
- Evaluating the SDF can be difficult
- More efficient way: **discretize the data**
- Resolution and bounding box
- Zero order field





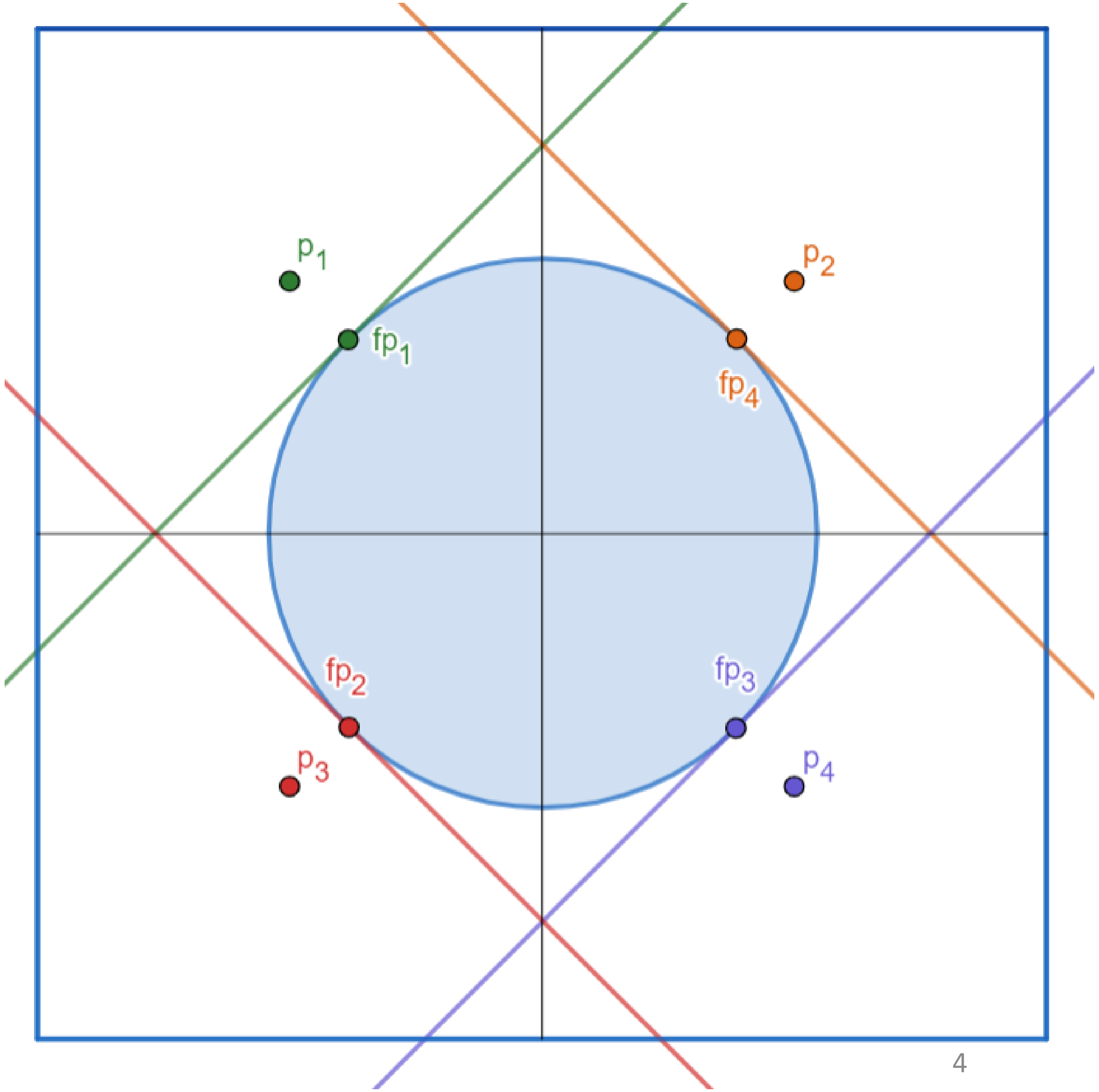
Sampled fields

- Combination of samples and filtering
- The field is **not C^1**
- We need an approriate filtering method
- The field consists of the **stored data and a filtering method**



Geometric distance field

- Store a **proxy geometry**
- For every cell:
  - Get the closest point on the surface (**footpoint**)
  - **Fit a geometry** that approximates the surface around the footpoint



Expectations

- The proxy geometry has to reconstruct the local differential geometry up to a given order
  - **Order 1: footpoint** and **normal**
  - **Order 2: footpoint, normal** and **curvatures**
- The SDF of the geometric invariant has to be easily computed

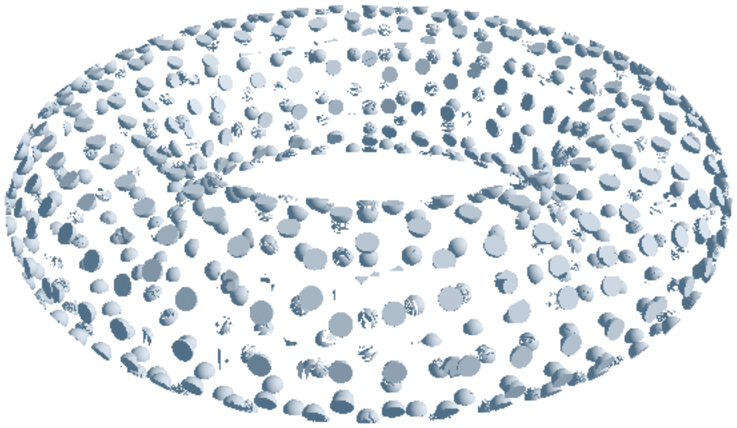
Overview

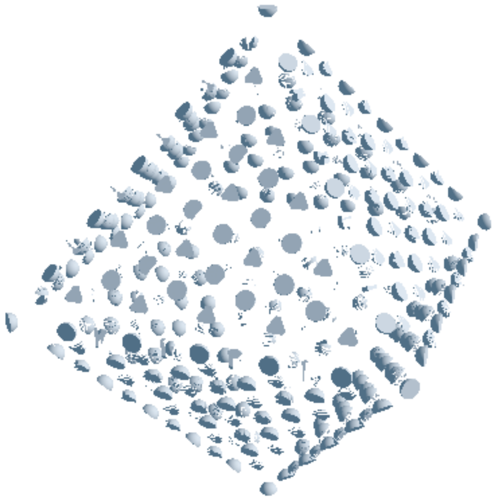
1. **Geometric fields in 2D**
    - Generating fields
    - New filtering method
2. **Geometric fields in 3D**
    - Generating and filtering the Order 1 field
    - Order 2 field
        - Proxy geometry
        - Generating the field
        - Filtering

Footpoint

Can be computed with the gradient of the SDF

$$\boldsymbol{fp}(\boldsymbol{p}) = \boldsymbol{p} - f(\boldsymbol{p}) \cdot \nabla f \qquad \text{where} \qquad \nabla f = \begin{bmatrix} \frac{\partial f}{\partial x_1} \\ \frac{\partial f}{\partial x_2} \\ \vdots \\ \frac{\partial f}{\partial x_n} \end{bmatrix}$$





Derivatives

For analytic input the footpoint and derivatives can be computed from the SDF

$$\partial_i\, d = \partial_i \|x - p\| = \frac{x_i - pi}{\|x - p\|}$$

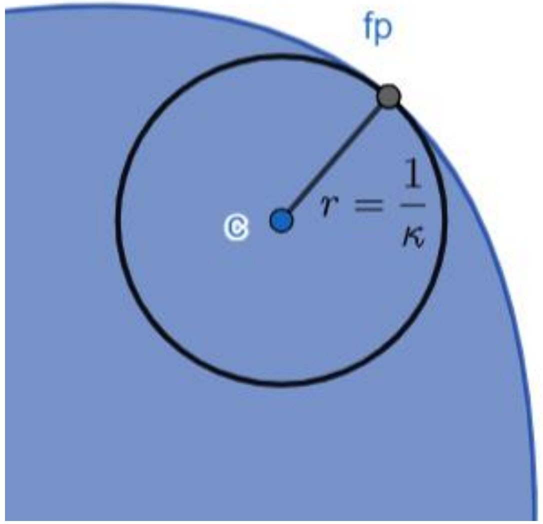
$$\partial_{ij}\, d = \partial_{ij} \|x - p\| = \frac{-1}{\|x - p\|} \partial_j\, p_i \;-\; \frac{(x_i - pi)(xj - pj)}{\|x - p\|^3}$$

Source: Xinghua Song , Bert Jüttler , Adrien Poteaux. Hierarchical Spline Approximation of the Signed Distance Function

Geometric fields on the plane

- The order 1 field stores the **tangent** at the footpoint

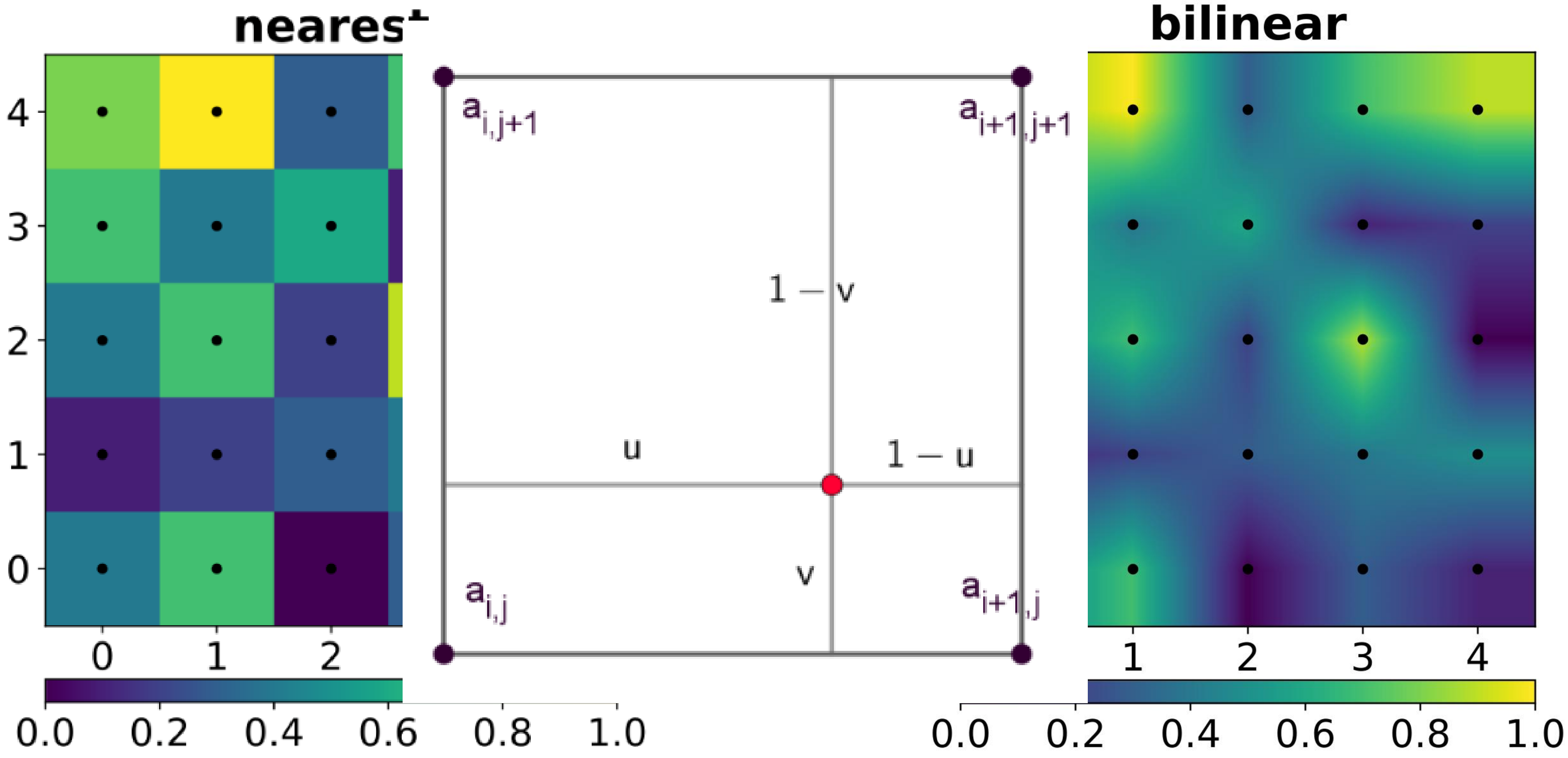
$$d_{line} = \langle x - p, n \rangle$$



- The order 2 field stores the **osculating circle**

Source: Róbert Bán és Gábor Valasek. „Geometric Distance Fields of Plane Curves"

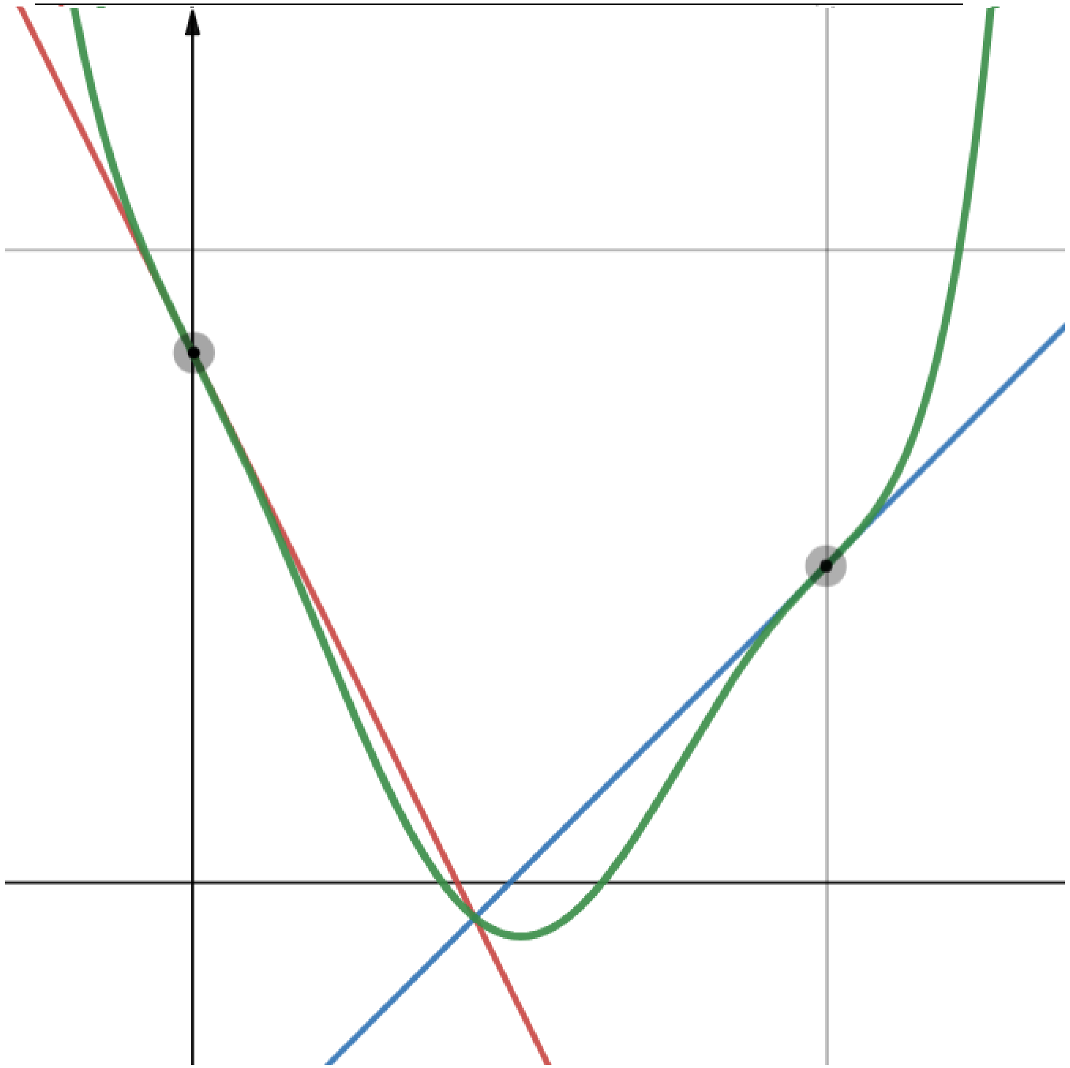
Bilinear filtering



- Calculate the distances from the geometry in the 4 nearest texels, and interpolating the result

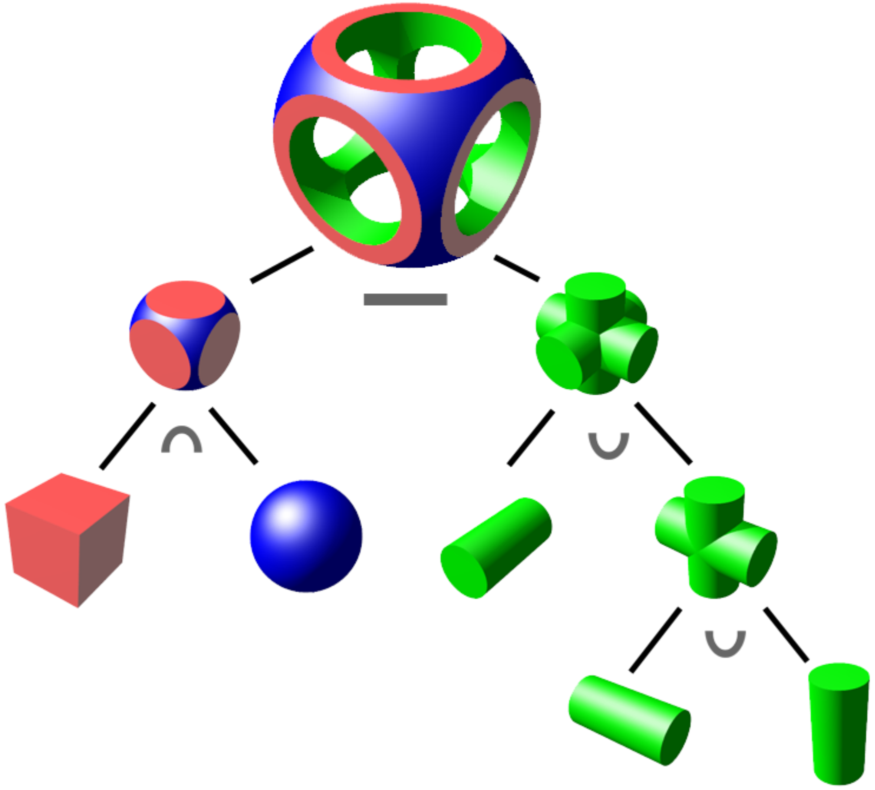
Issues with standard filtering

- Hardware accelerated bi/trilinear filtering is not accurate
- **Using higher order data** – problem with derivatives
- Algebraic fields: higher order interpolation (eg. Hermite)
- Geometric
  - Blending function
  - Local CSG

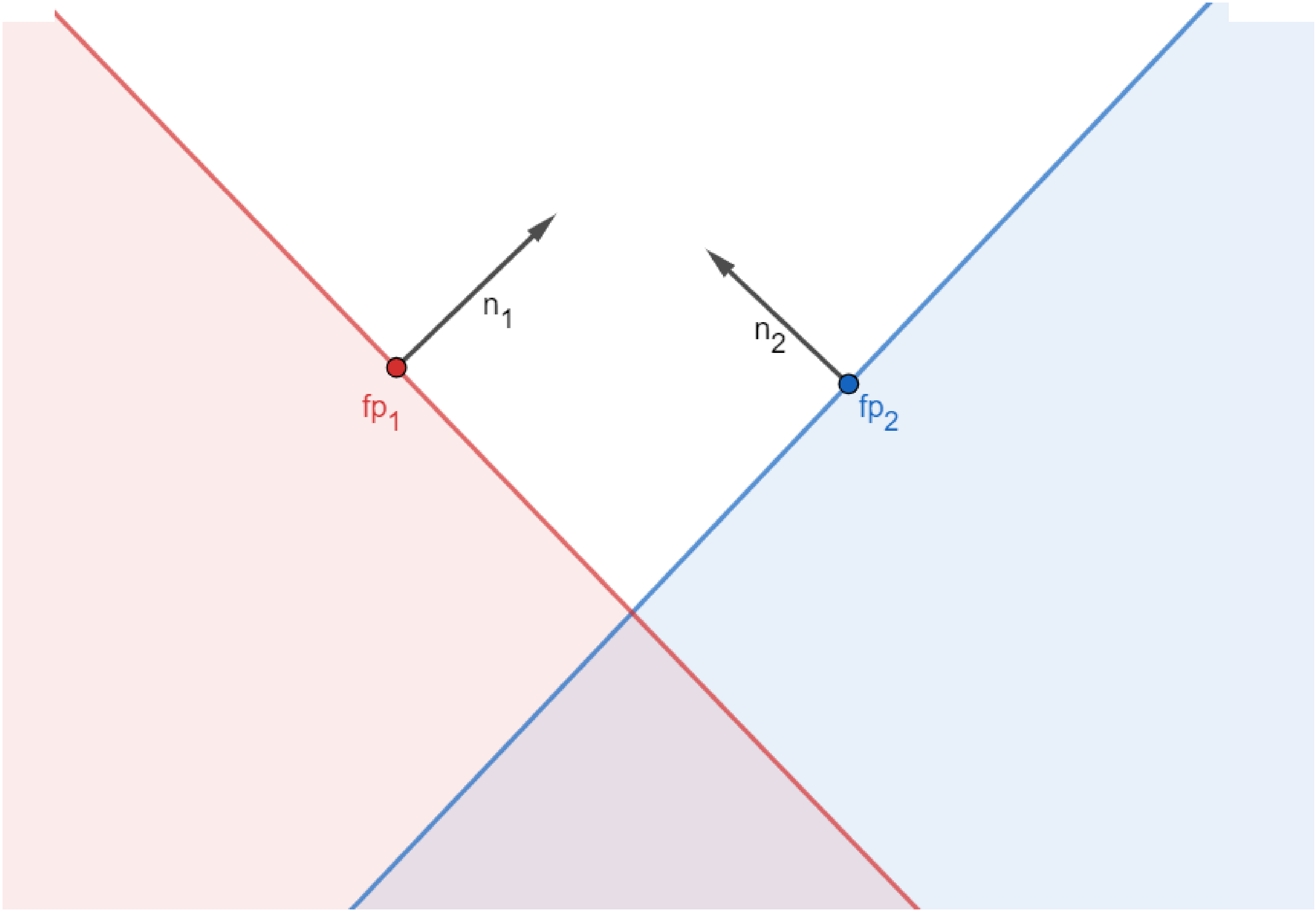


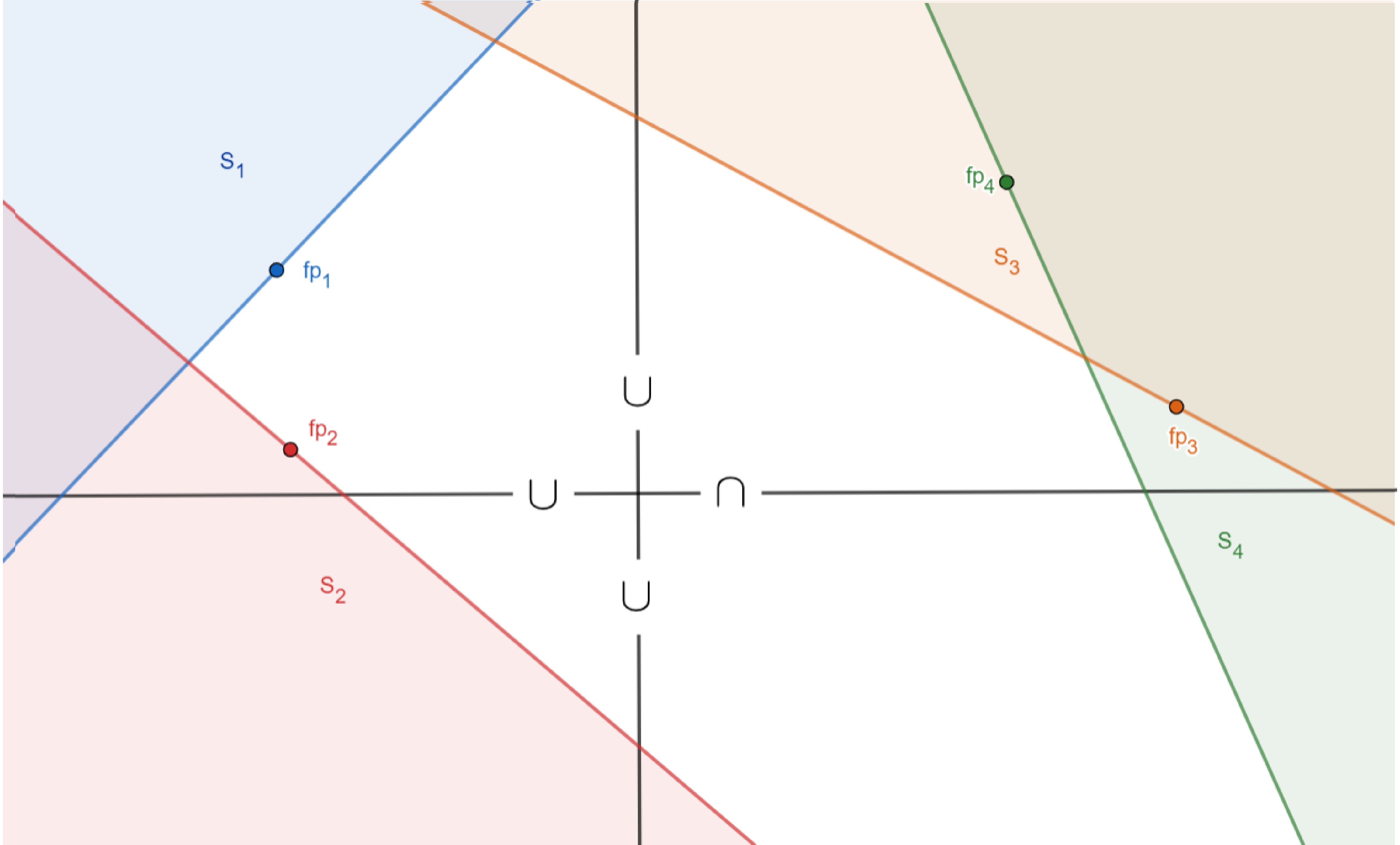
CSG filtering

- Let's take advantage of the fact that we are storing geometries
- Bilding a **CSG (constructive solid geometry) tree from the stored halfplanes**



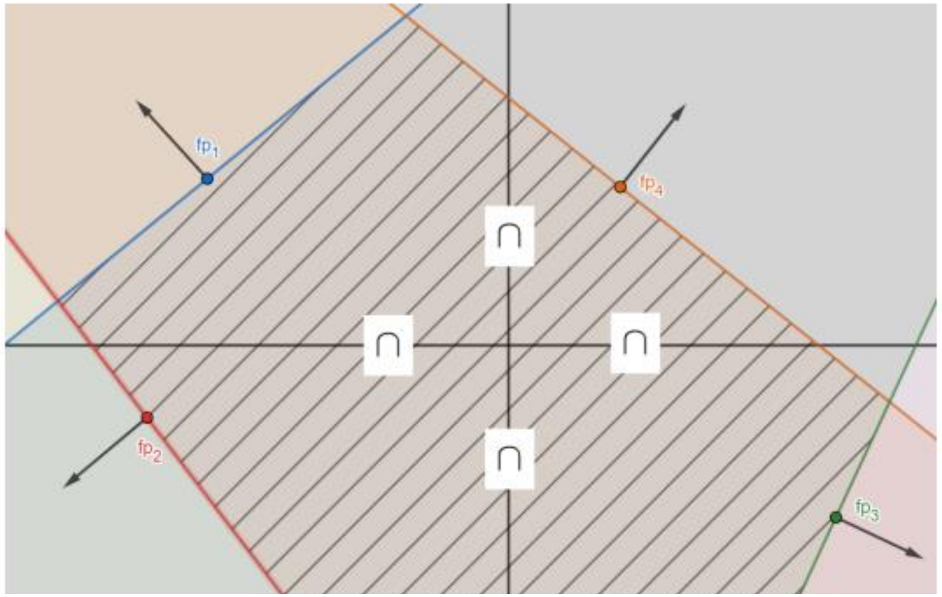
Intersection or union

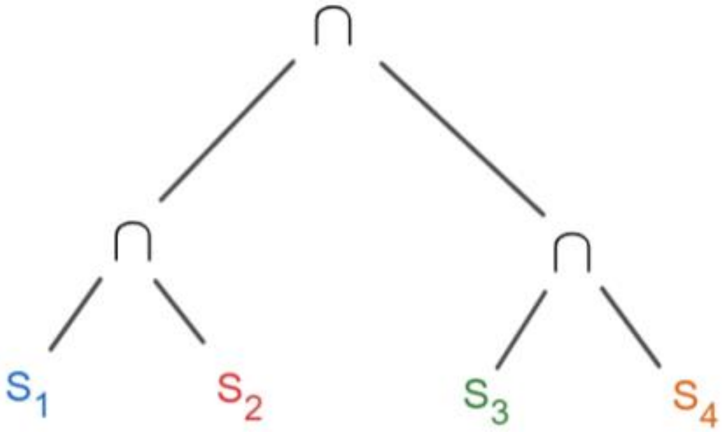


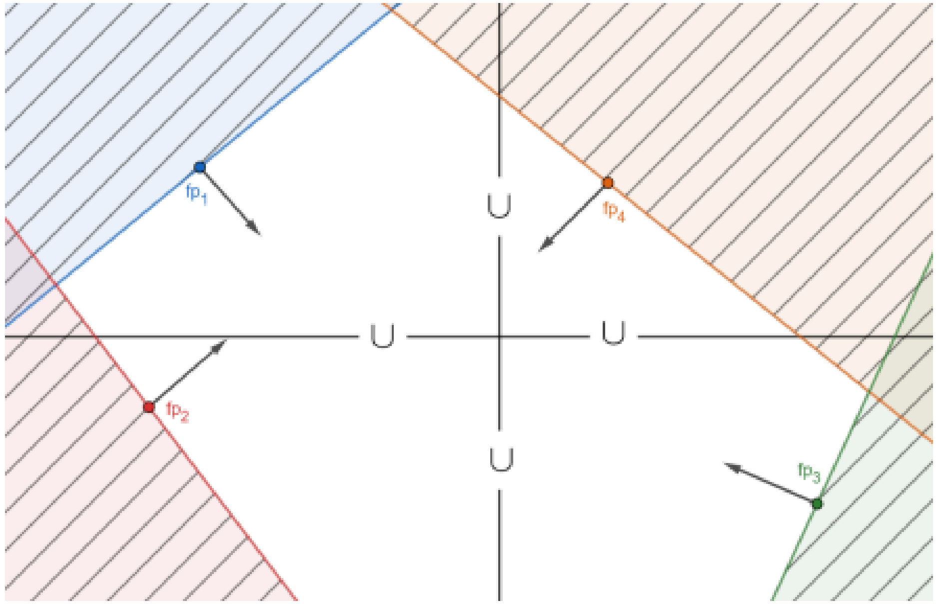


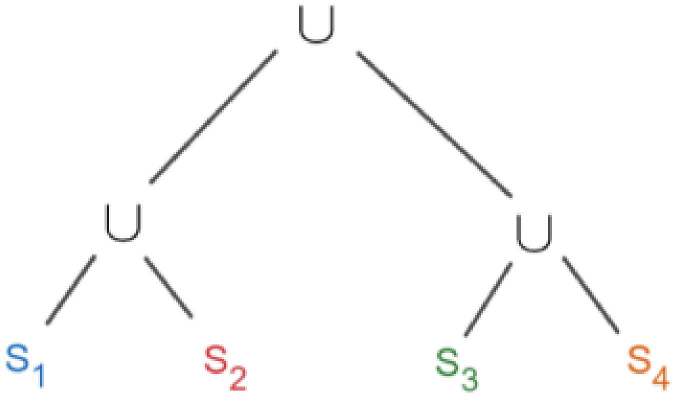
S1
fp1
fp2
S2
fp4
S3
fp3
S4
∪
∪
∩
∪

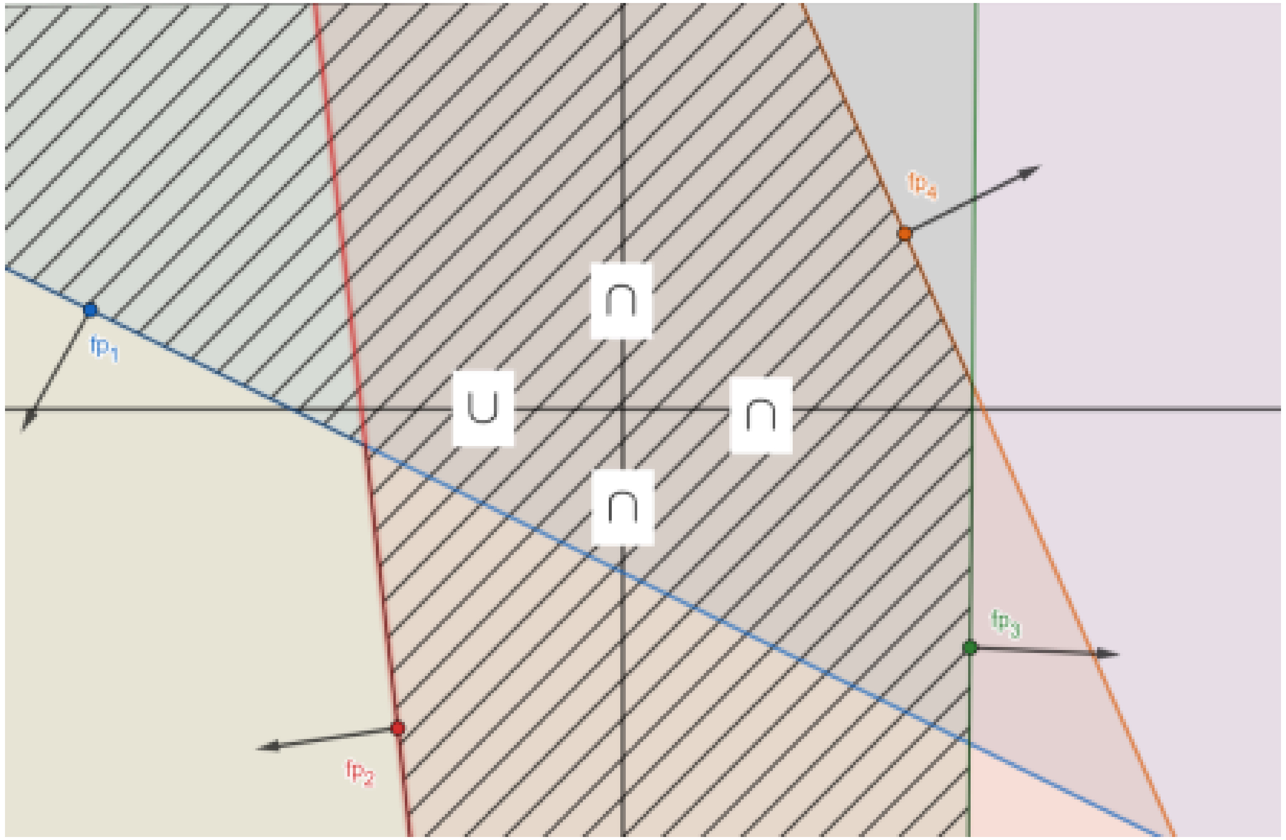
Building the CSG tree

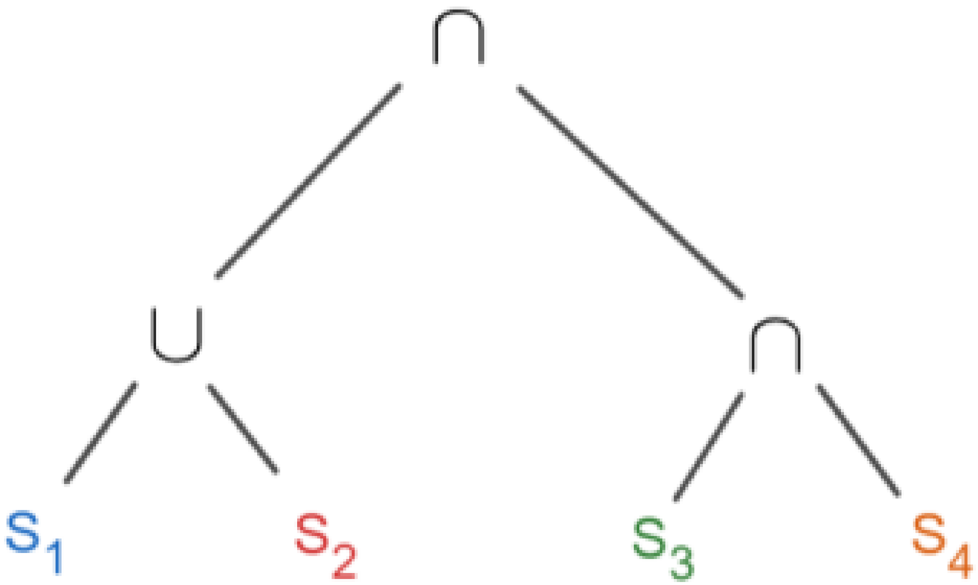


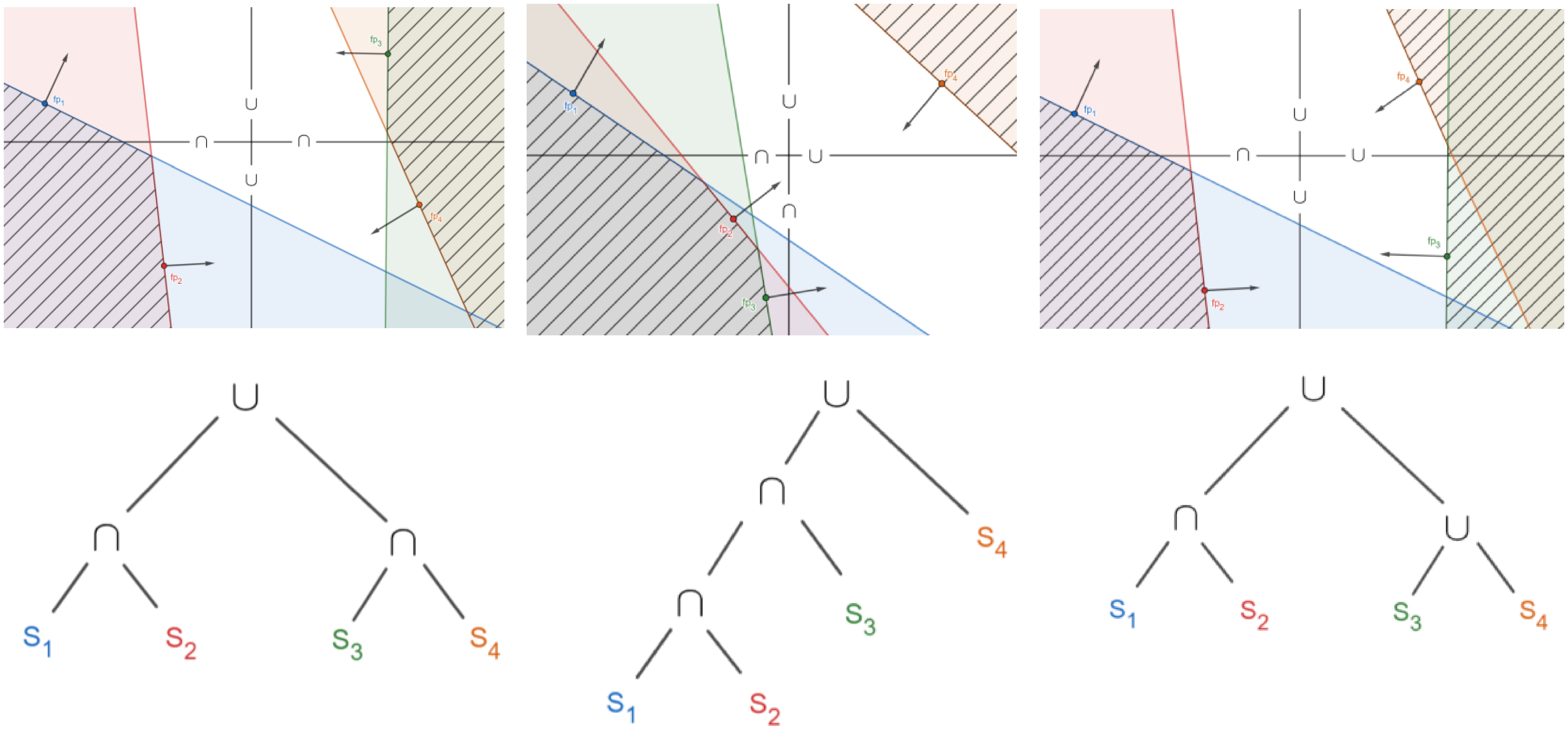


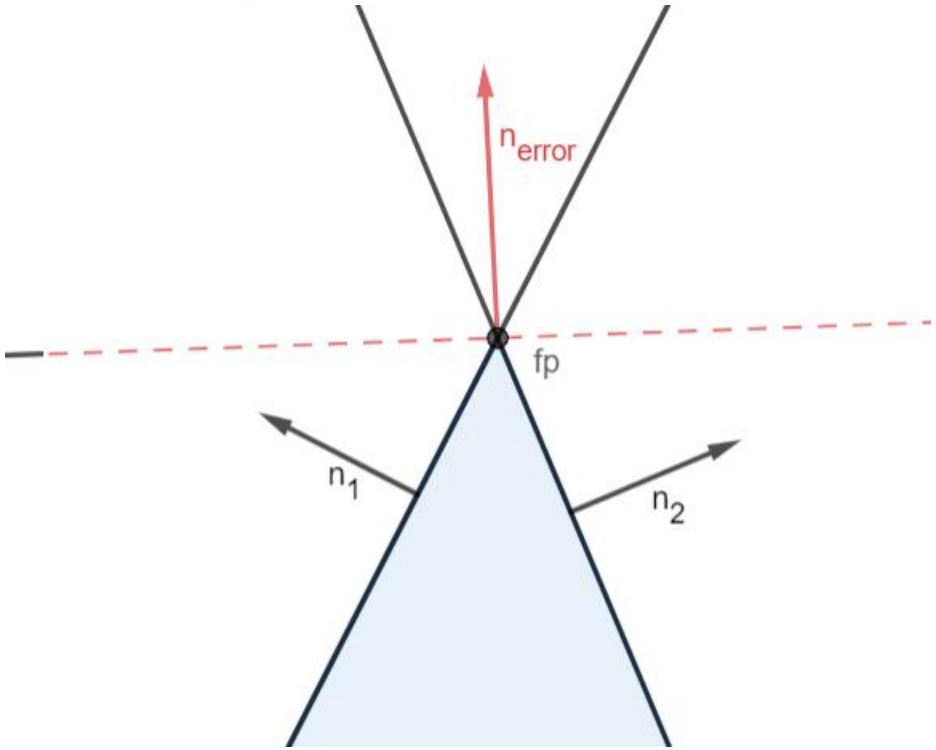
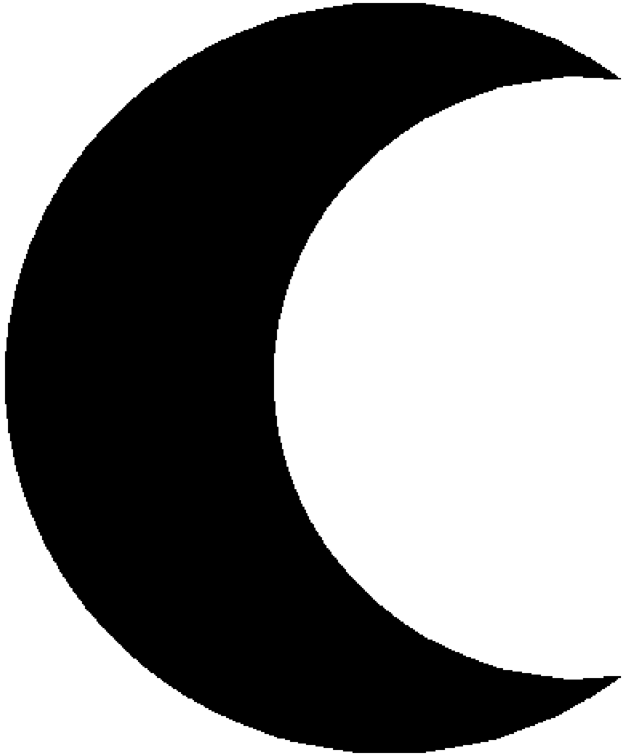




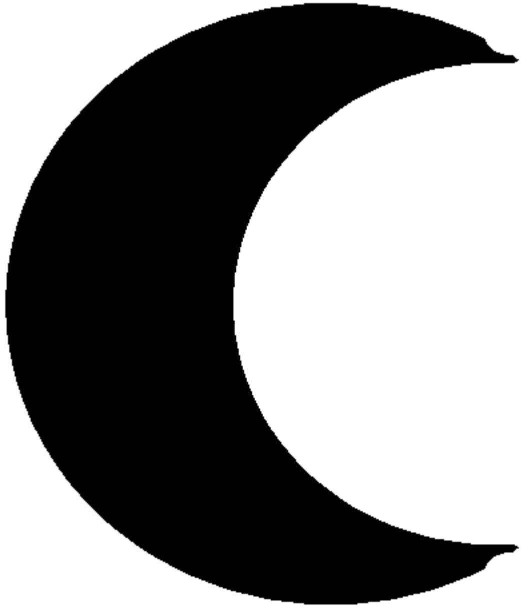

fp1
fp2
fp3
fp4
∩
∪
∩
∩

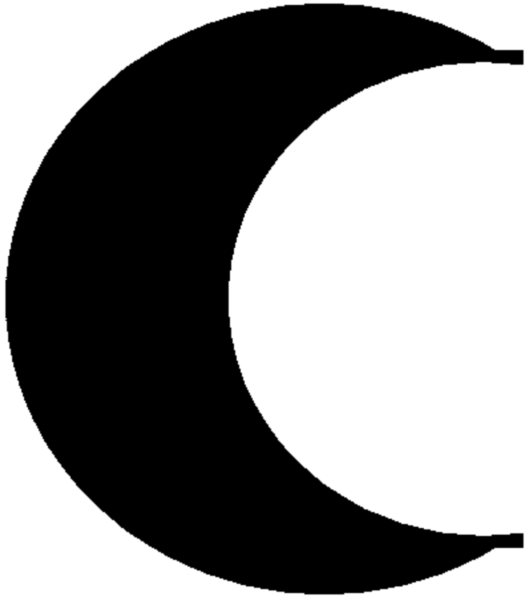

∩
∪
∩
S1
S2
S3
S4

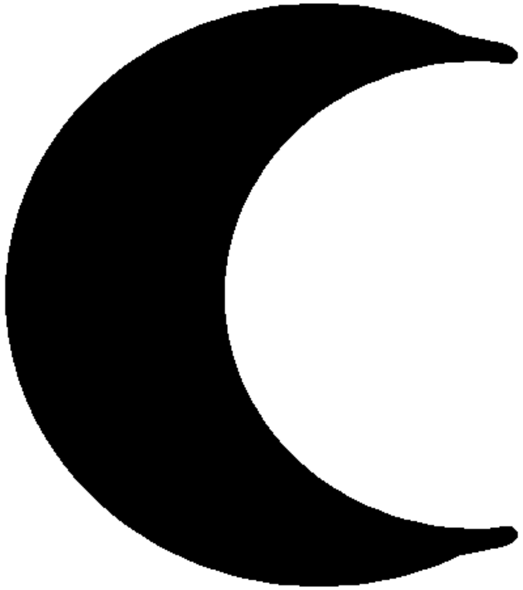

fp1
fp2
fp3
fp4
∪
∩
S1
S2
S3
S4

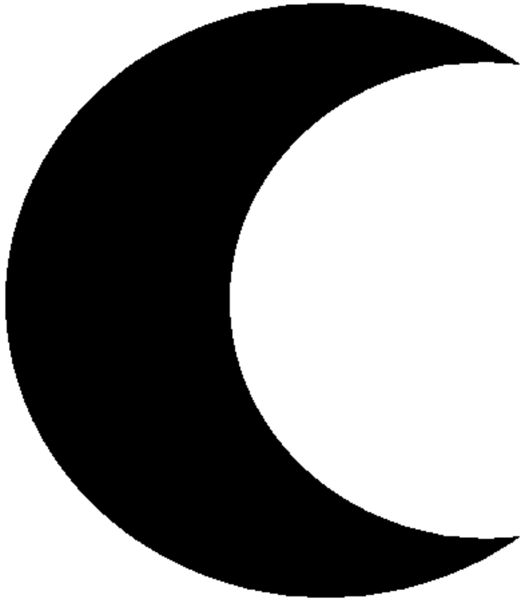

n error
fp
n 1
n 2


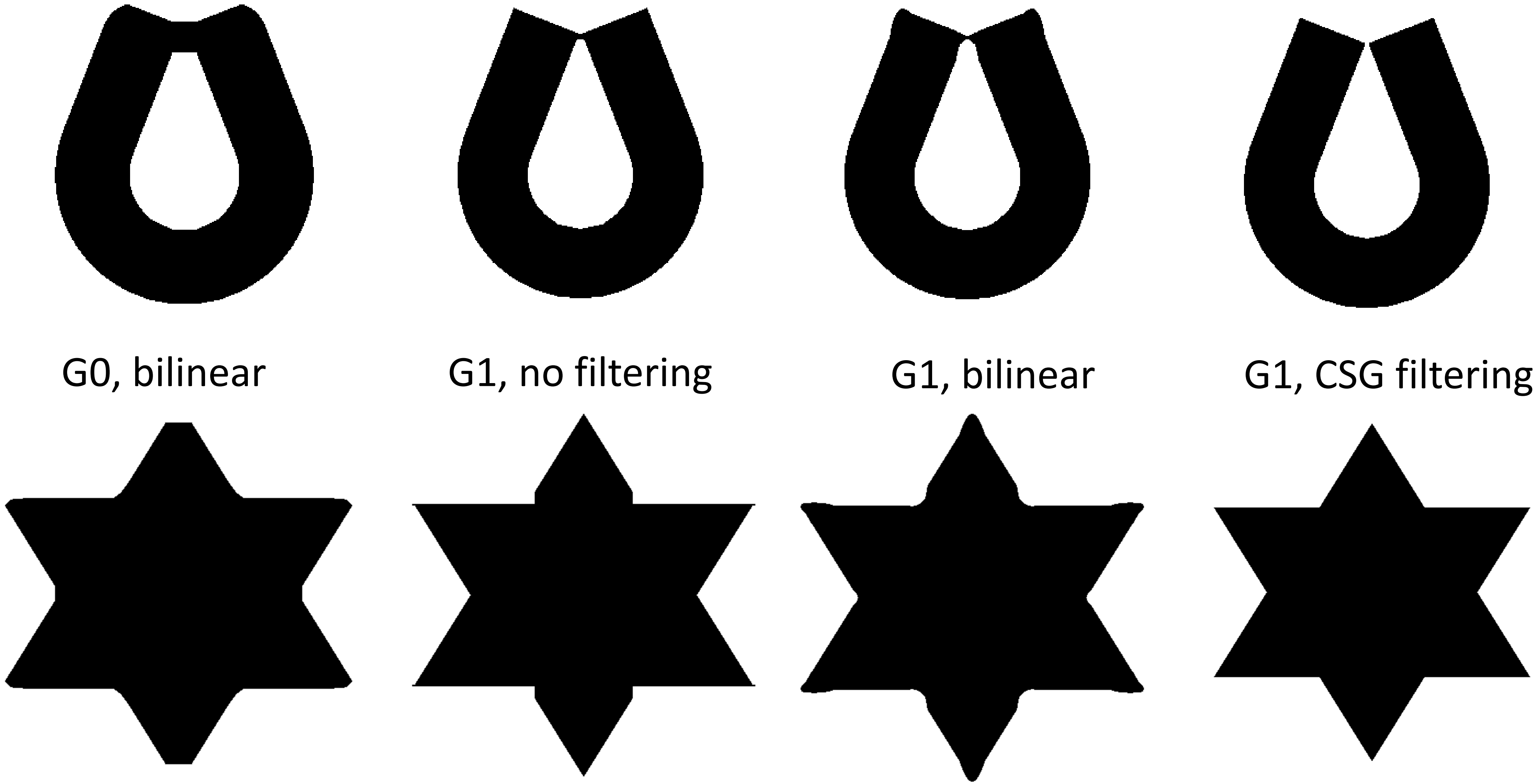
Results


G0, bilinear

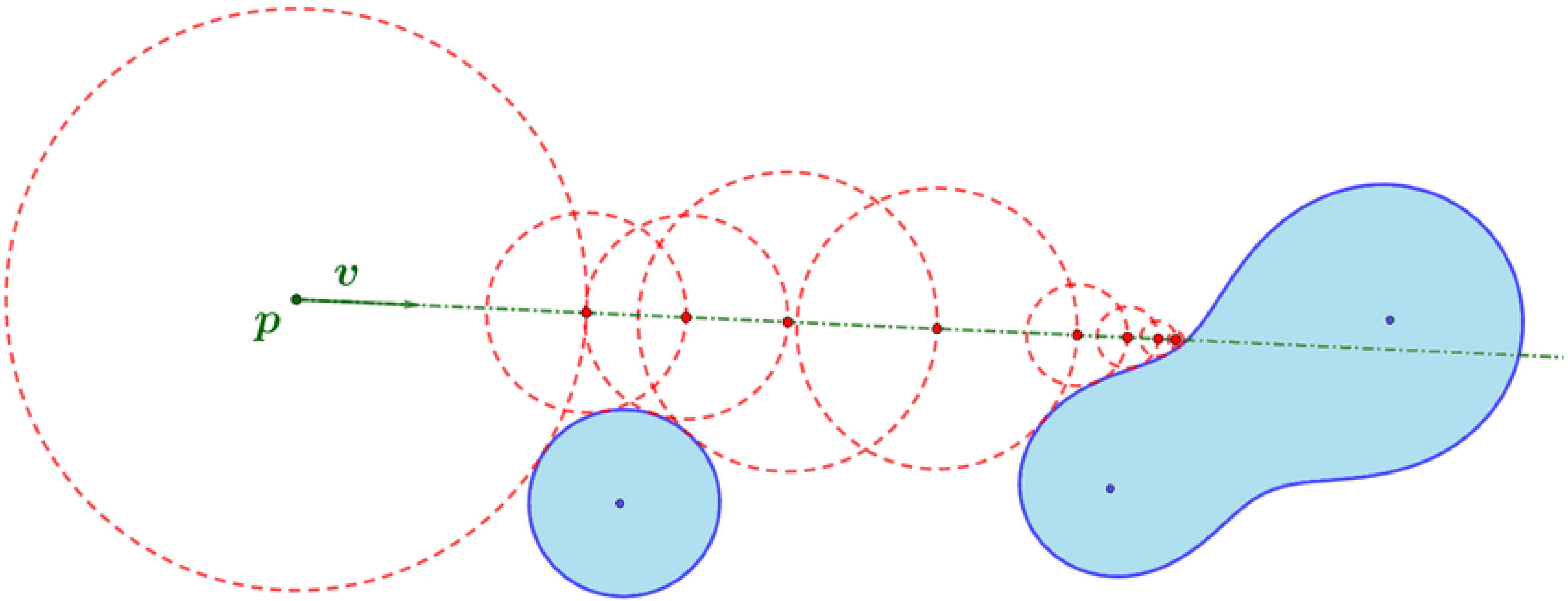

G1, no filtering


G1, bilinear

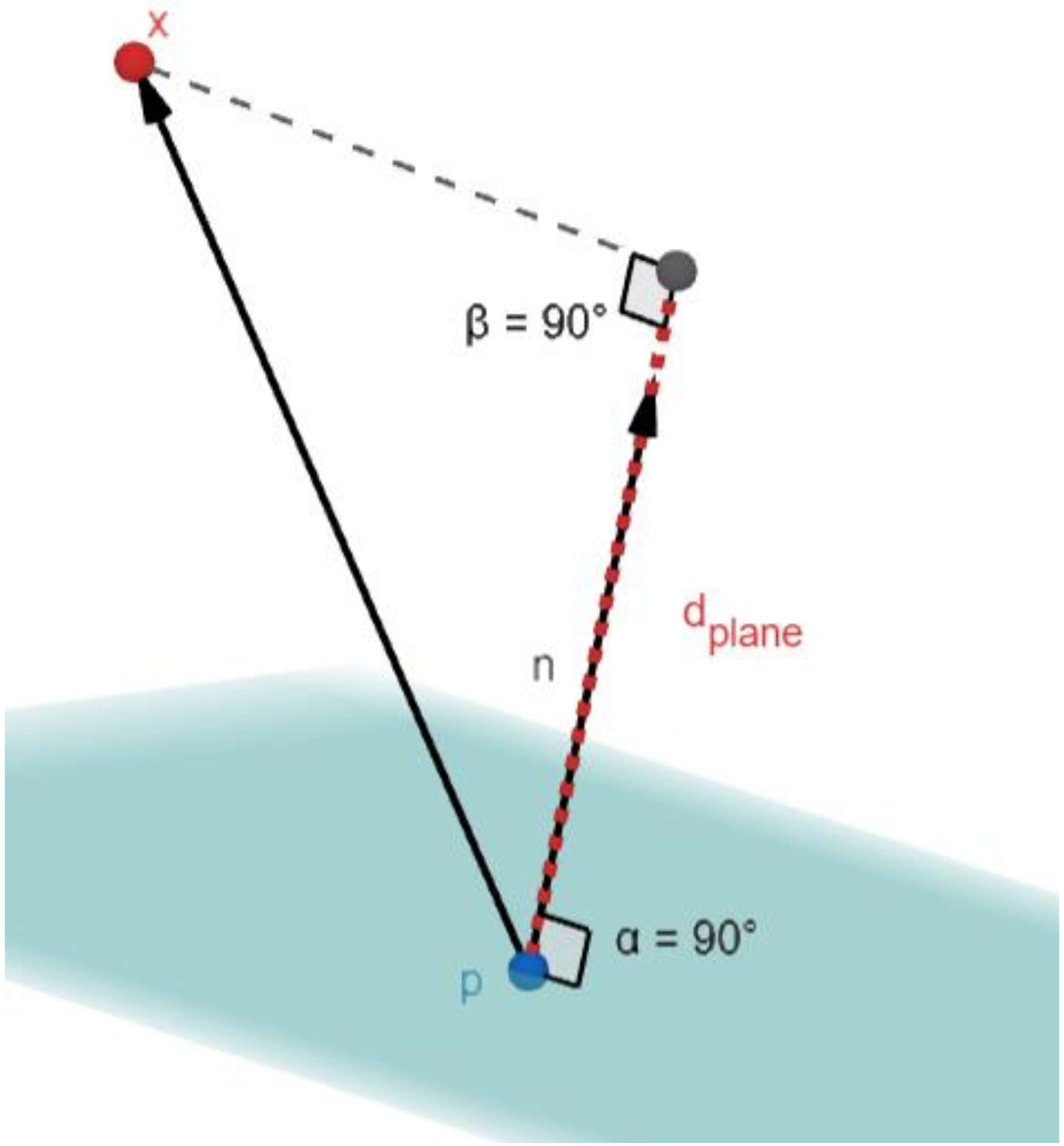

G1, CSG filtering


G0, bilinear
G1, no filtering
G1, bilinear
G1, CSG filtering

Distance fields in 3D



Order 1 field

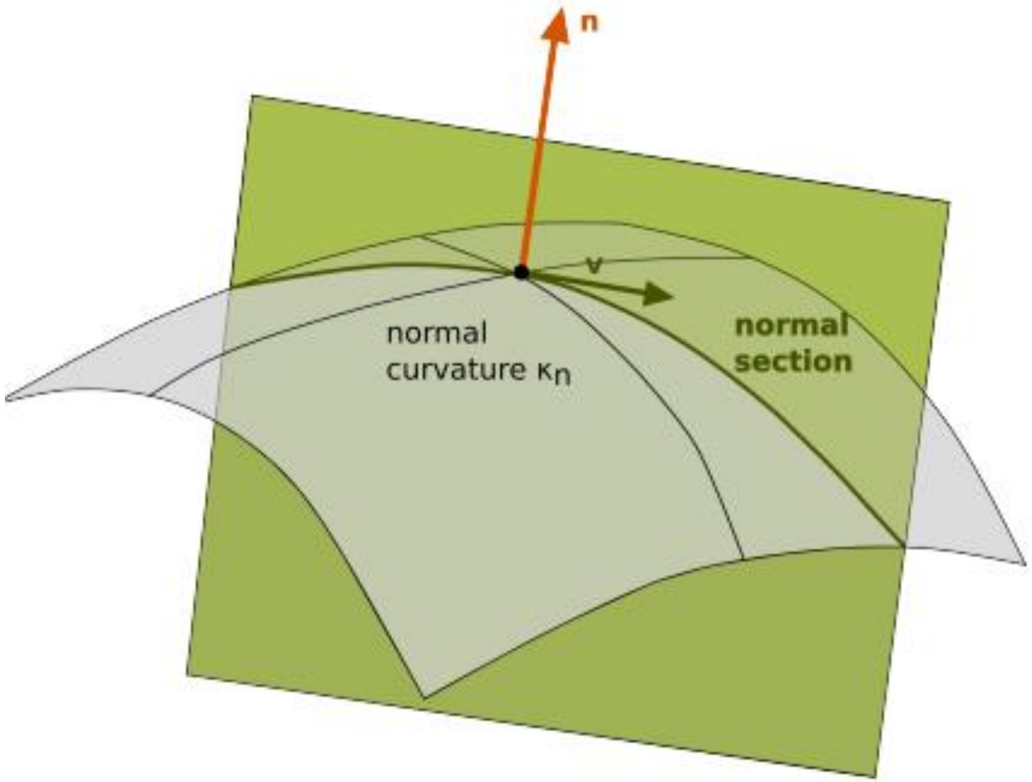


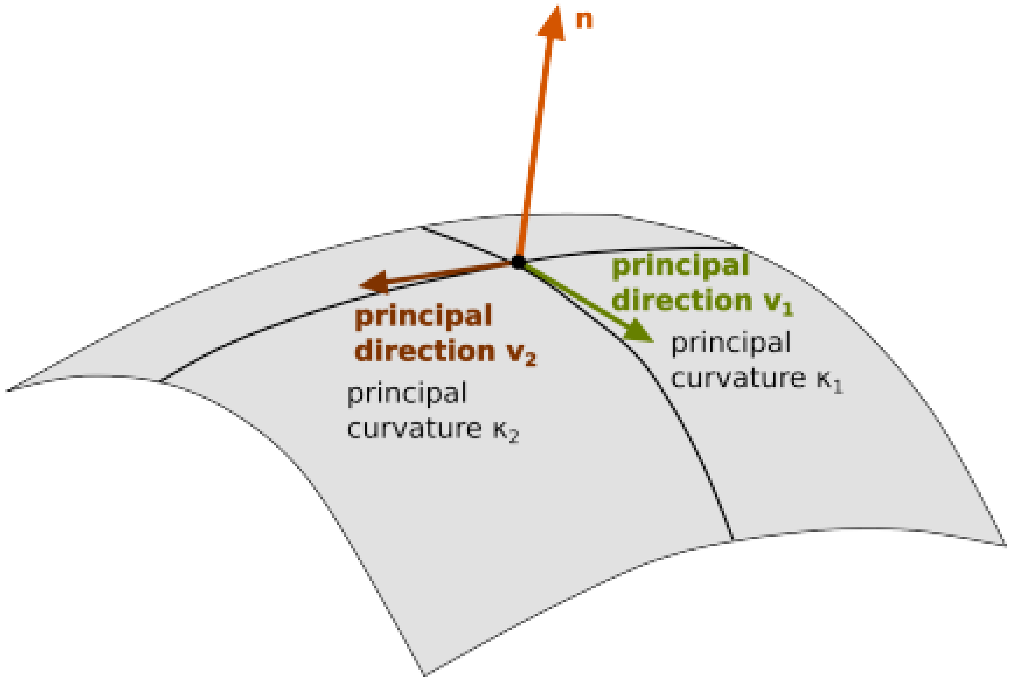
- The geometric invariant is the **tangent plane** at the footpoint
- Defined by the footpoint and normal

$$d_{plane} = \langle x - p, n \rangle$$

Order 2 field

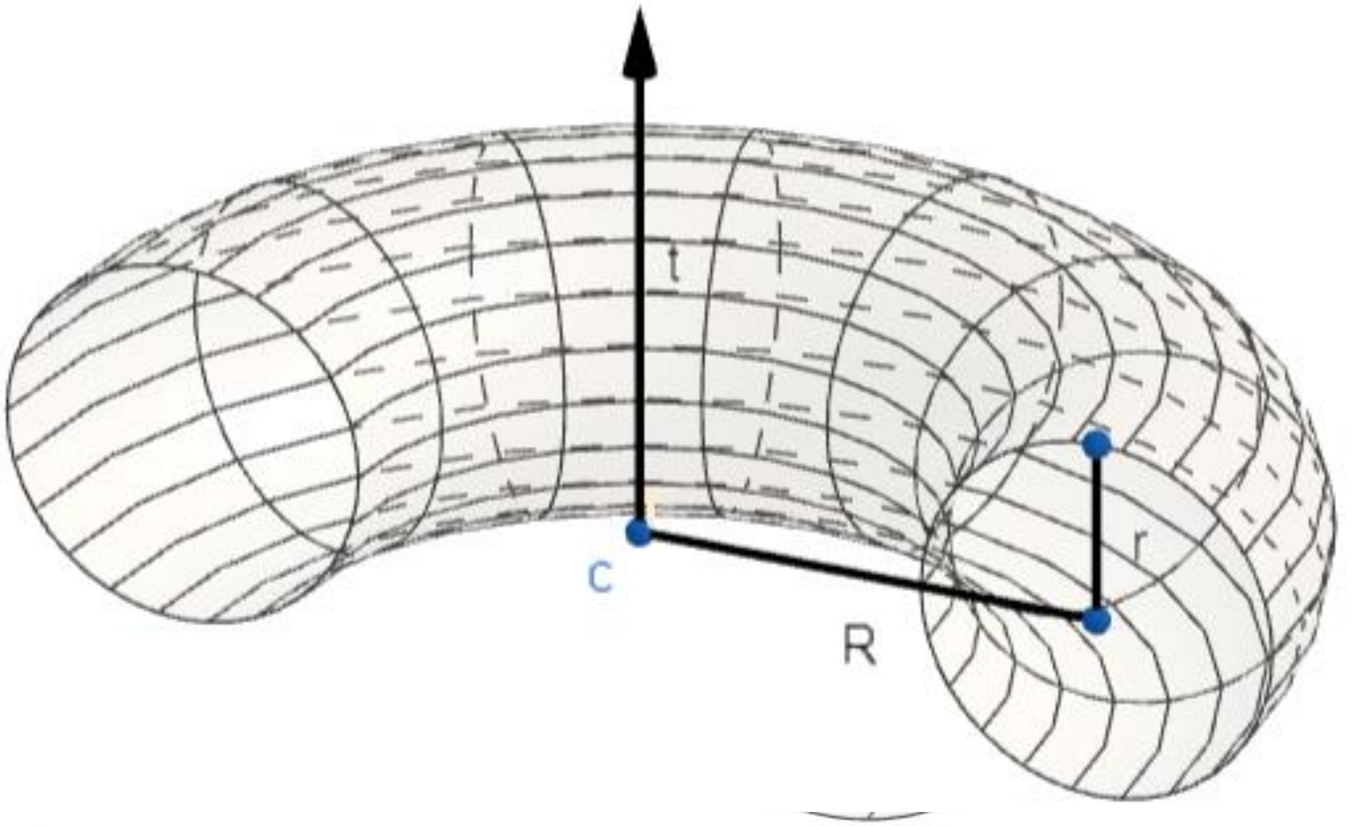
- Footpoint, normal, principal curvatures and directions





Source: Vas Lia, Surfaces

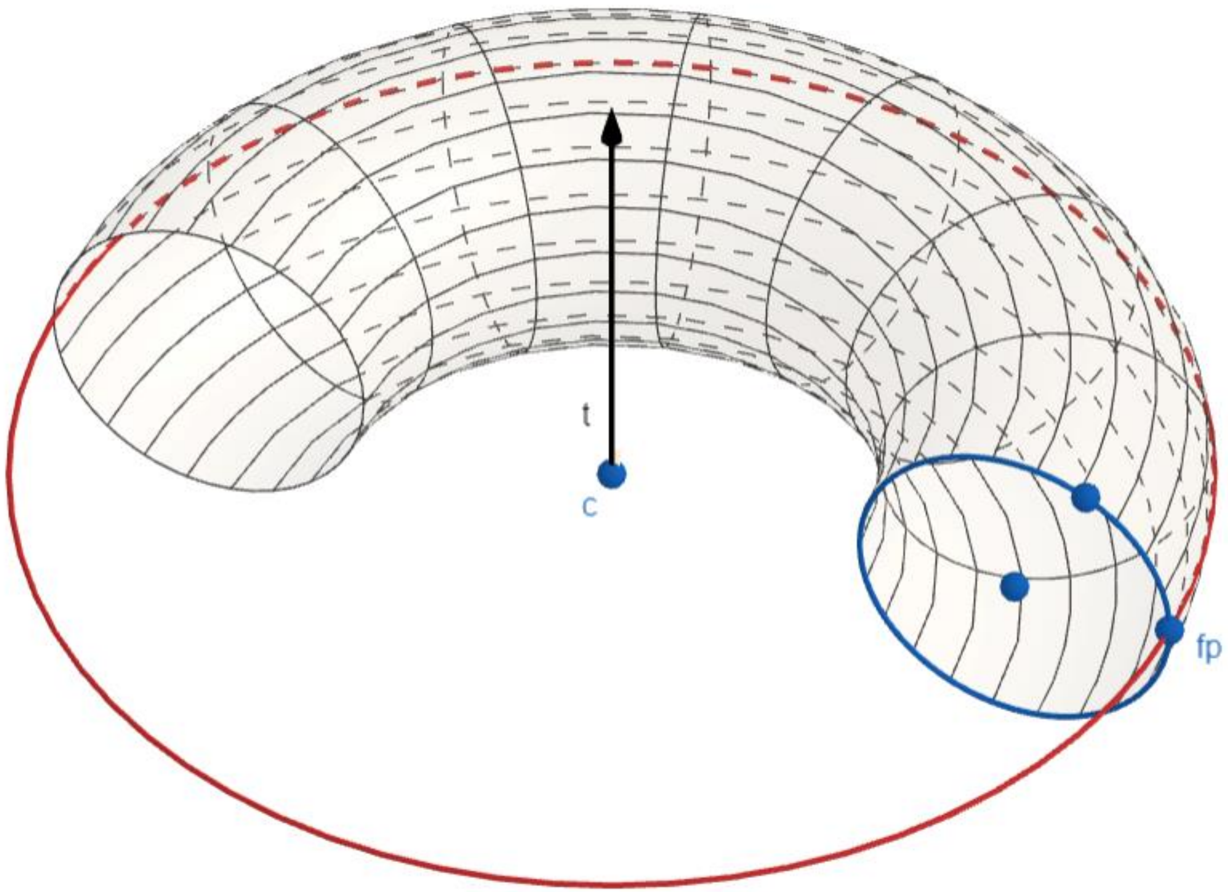
Why the torus?



- Can represent every combination of principal curvatures
- The SDF can be computed with a simple formula

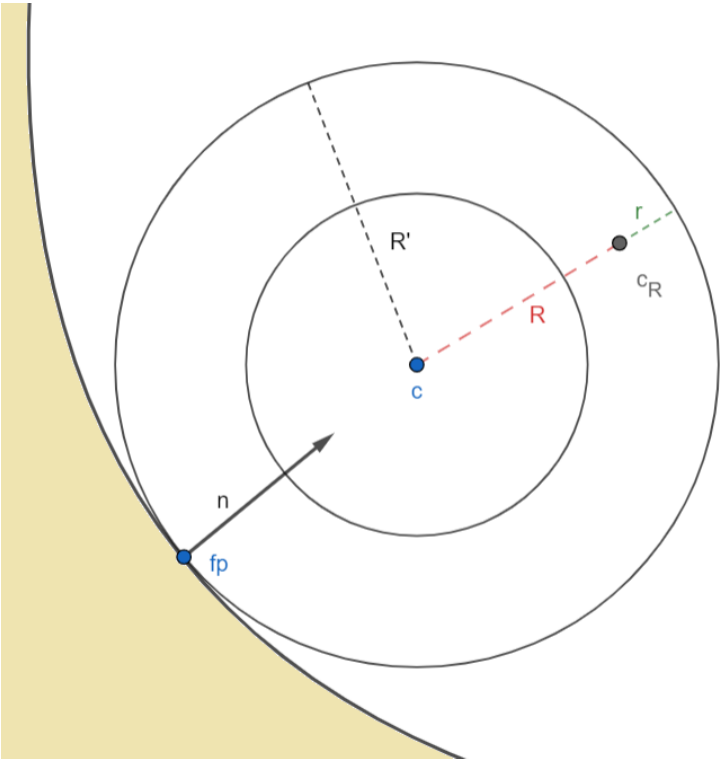
Source: David Eberly. Fitting 3D Data with a Torus

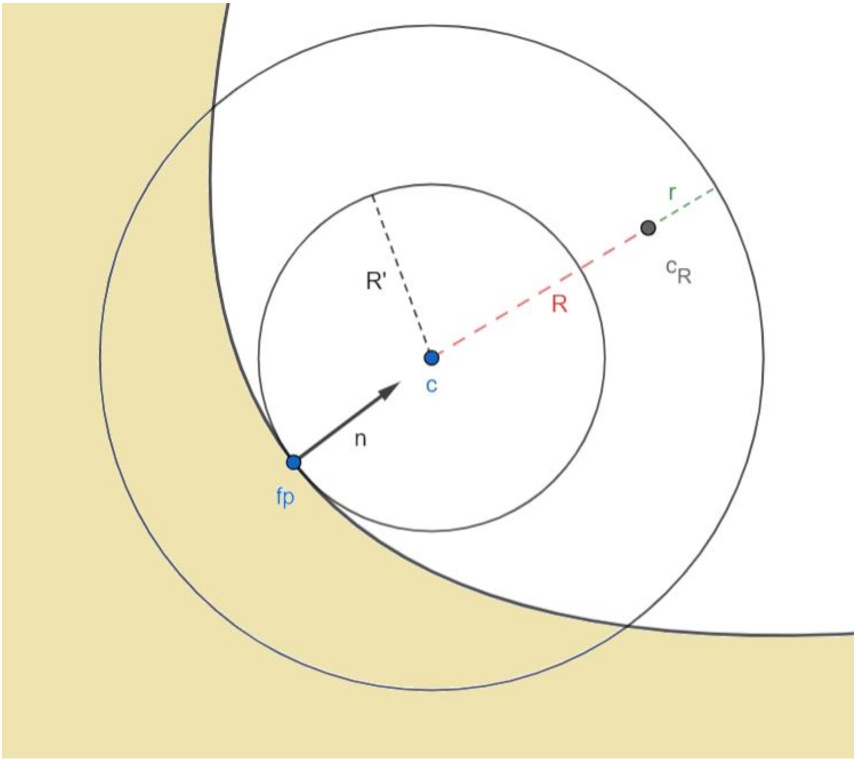
A better representation



The footpoint representation

- Works for **plane and infinite cylinder**
- The **previous formula can be used** for the SDF
- The radius and the center can be computed from the principal curvatures





Fitting the torus

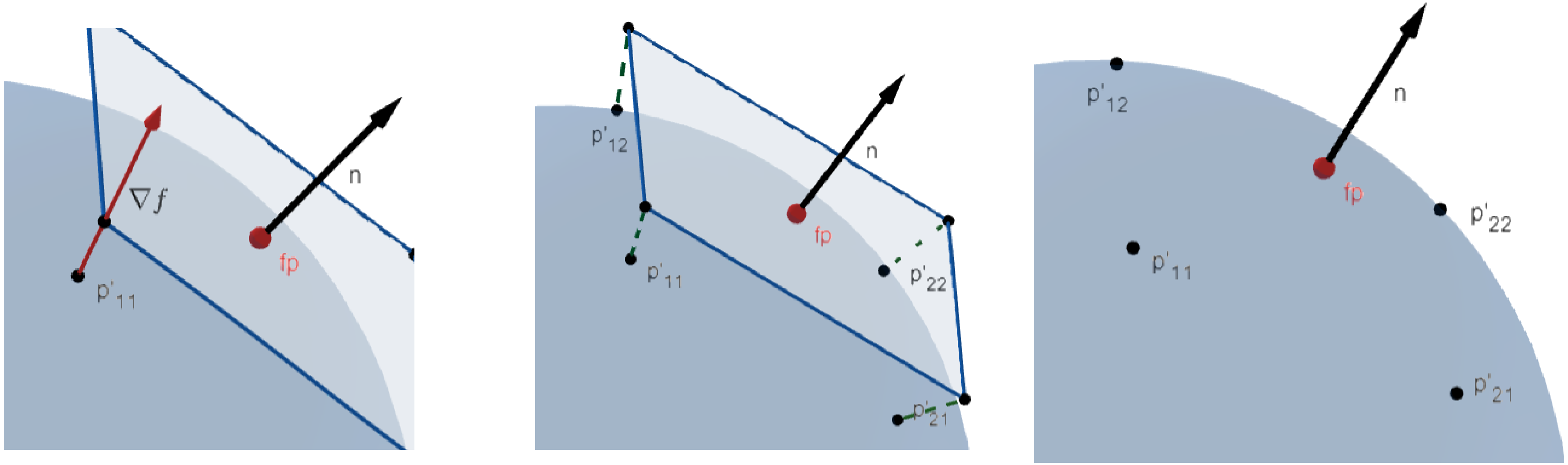
1. points from the surface

2. fitting an degree 3 surface on the points

3. computing the principal curvatures with the Weingarten matrix

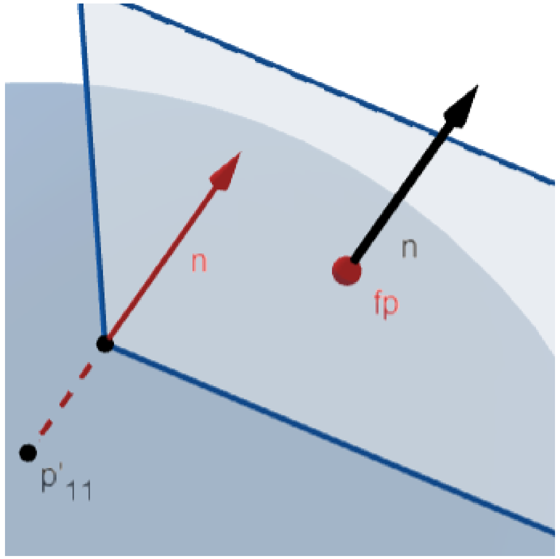
1. Surface points

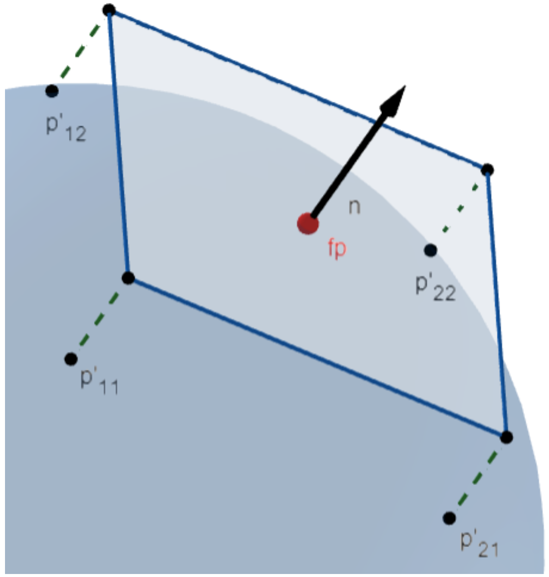
- **Finer grid** around the footpoint
- Find the **closest points** of the surface to the points of the fine grid

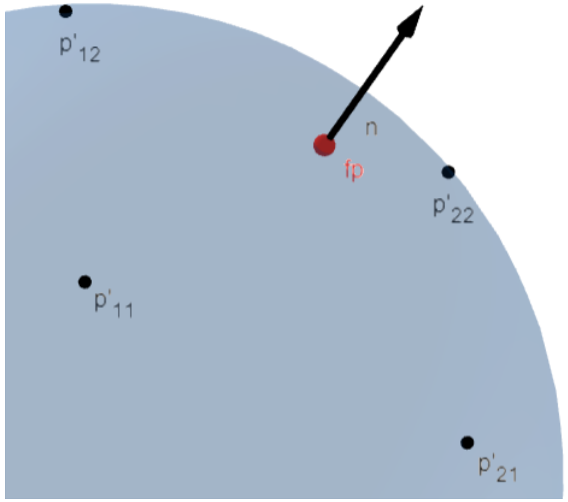


Orthogonal projection

- Instead of using footpoints **project the points** of of the fine grid orthogonally on the surface

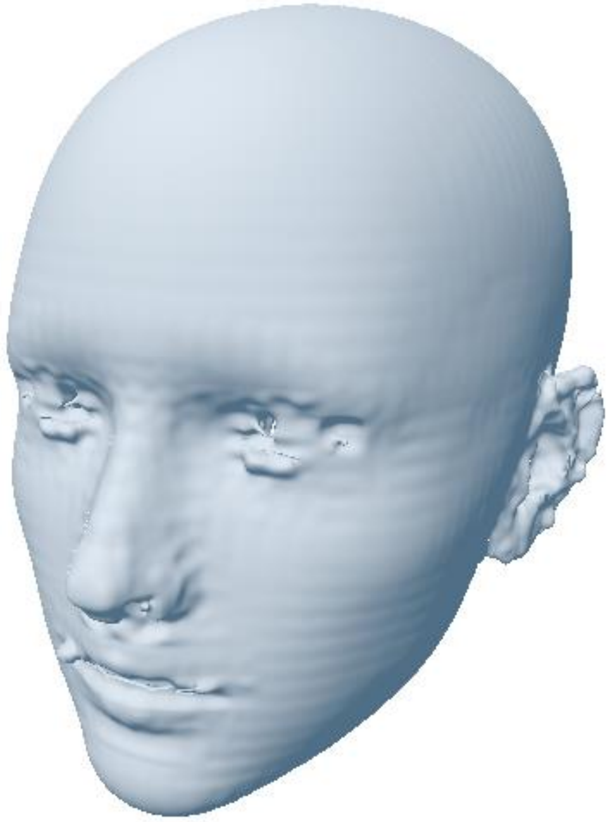






Projection vs Footpoints

- Bad approximation
- Difficult paralellization



Footpoints

Orthogonal projection

2. Degree 3 algebraic surface

$$f(x_i, y_i) = \begin{bmatrix} \frac{x_i^2}{2} & x_i y_i & \frac{y_i^2}{2} & x_i^3 & x_i^2 y_i & x_i y_i^2 & y_i^3 \end{bmatrix} \boldsymbol{b} = z_i$$

$$\boldsymbol{b} = \begin{bmatrix} A & B & C & D & E & F & G \end{bmatrix}^T$$

3. Curvatures

Weingarten matrix

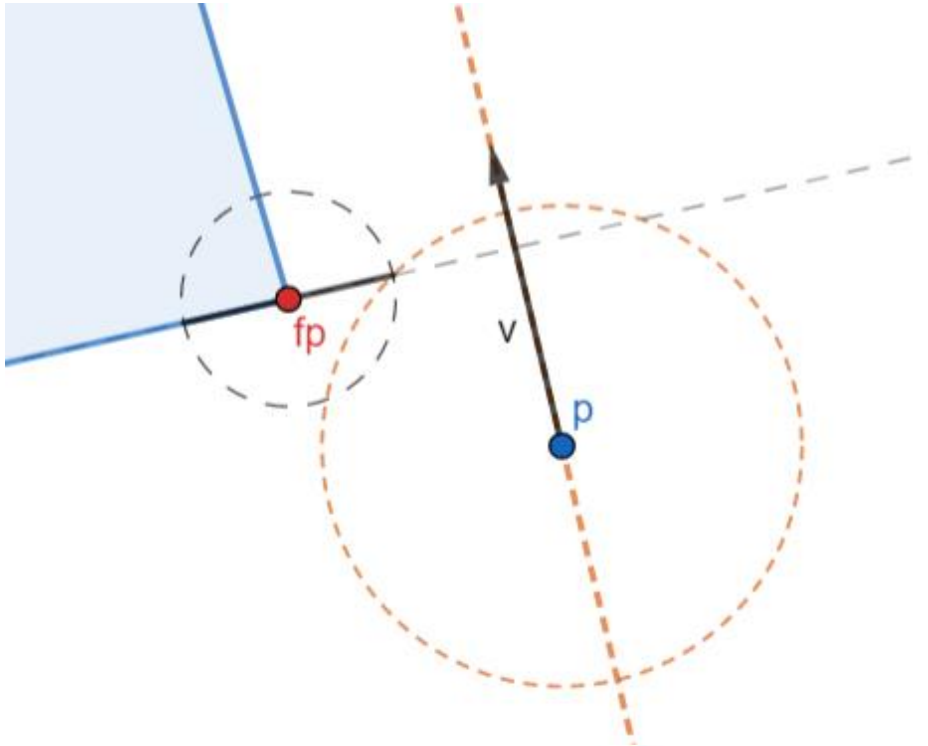
- Eigenvalues are the **principal curvatures**
- Eigenvectors are the principal directions

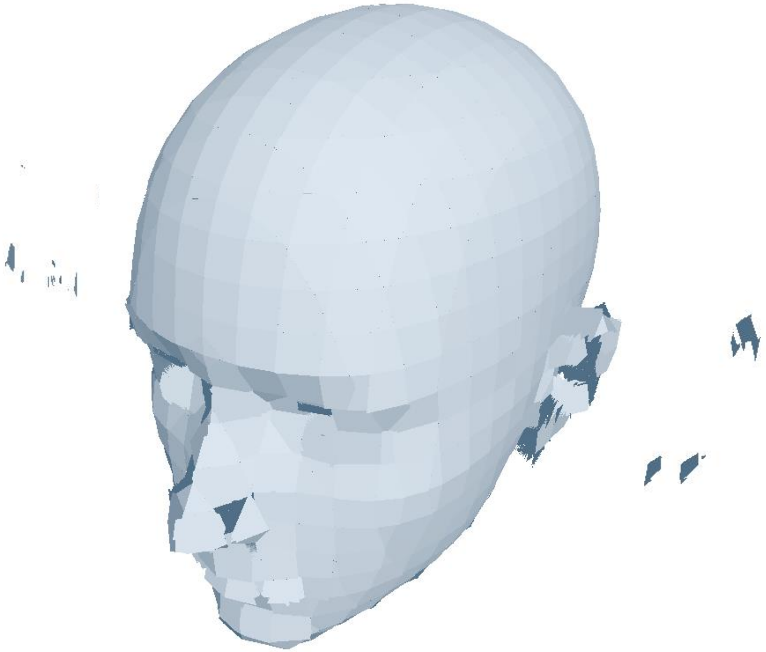
$$W = \begin{bmatrix} A & B \\ B & C \end{bmatrix}$$

The **axis** of the torus should be the **direction of the bigger principal curvature**

Source: Jack Goldfeather és Victoria Interrante. „A Novel Cubic-Order Algorithm for Approximating Principal Direction Vectors”

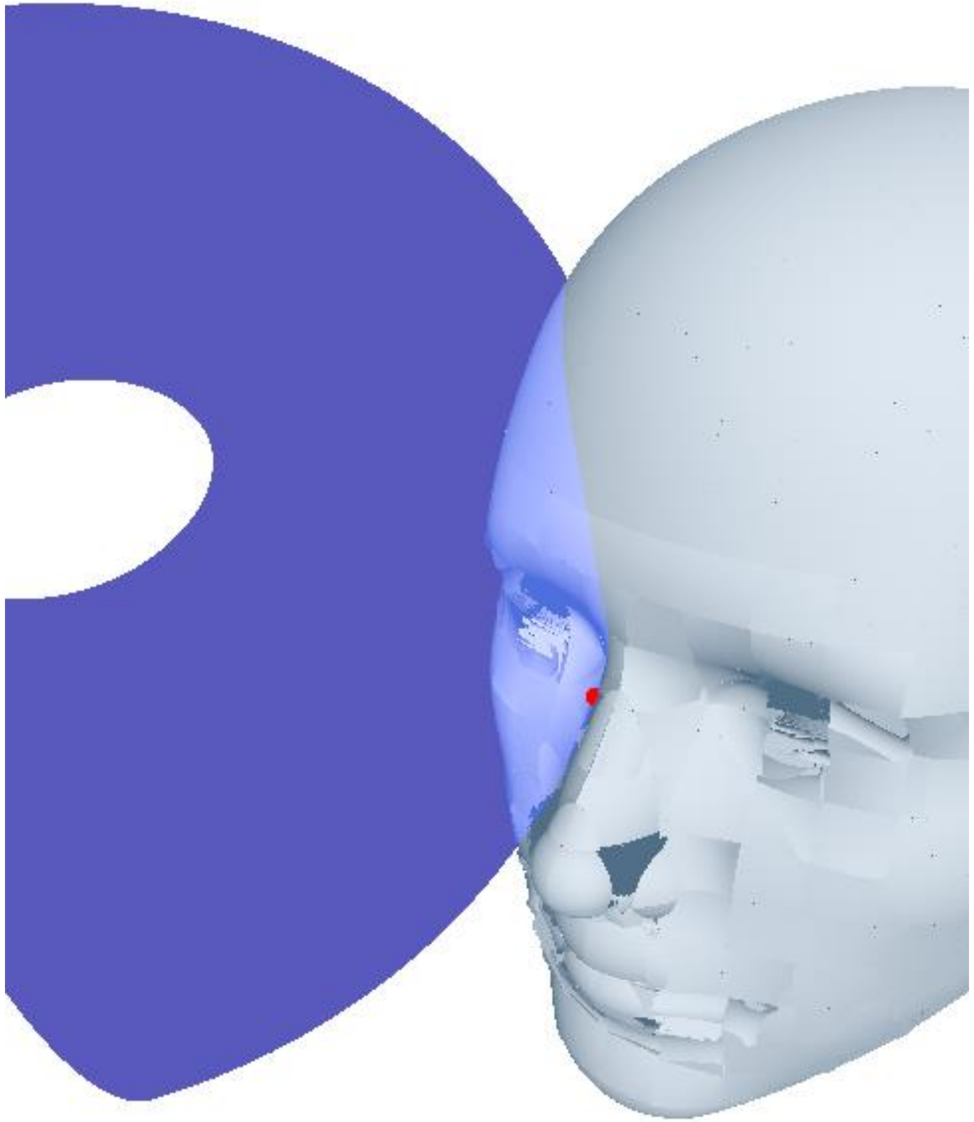
Field evaluation – Order 1



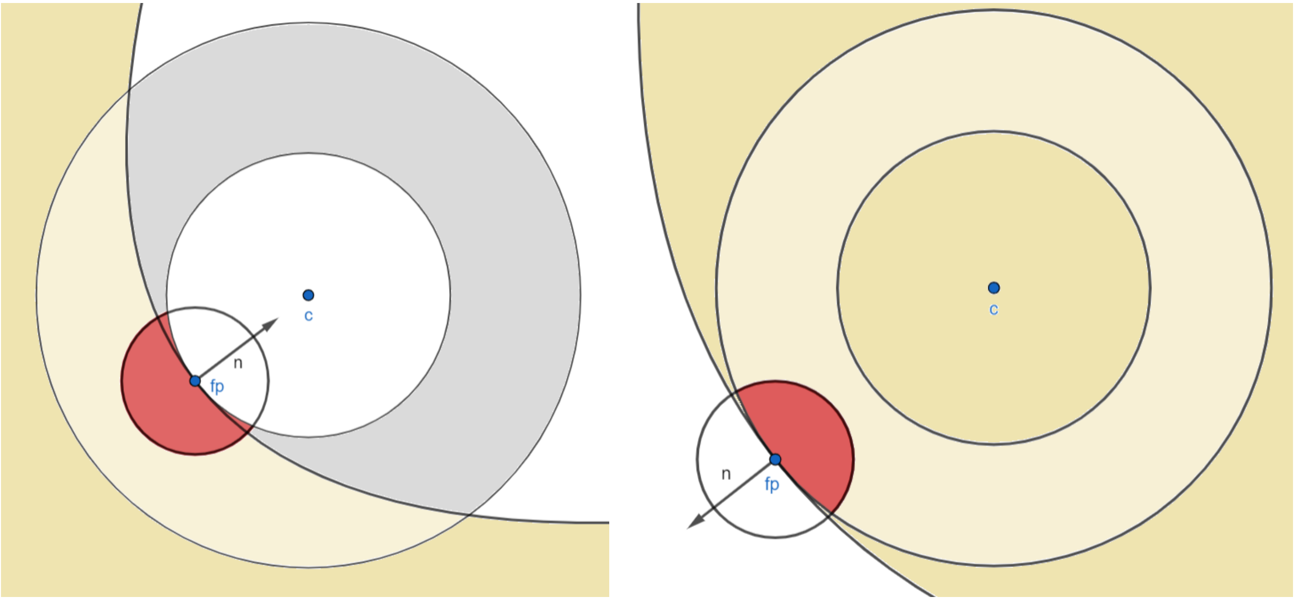


Field evaluation – Order 2

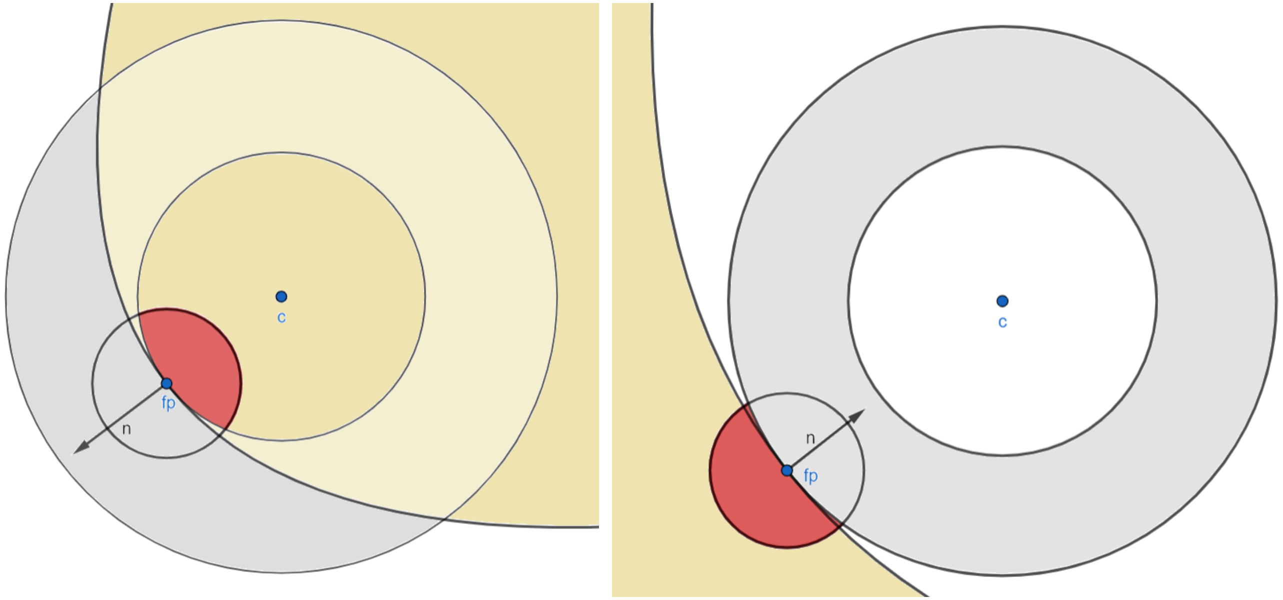
Problem: The geometry reconstructed from the data is not always the proxy of the surface



Solution

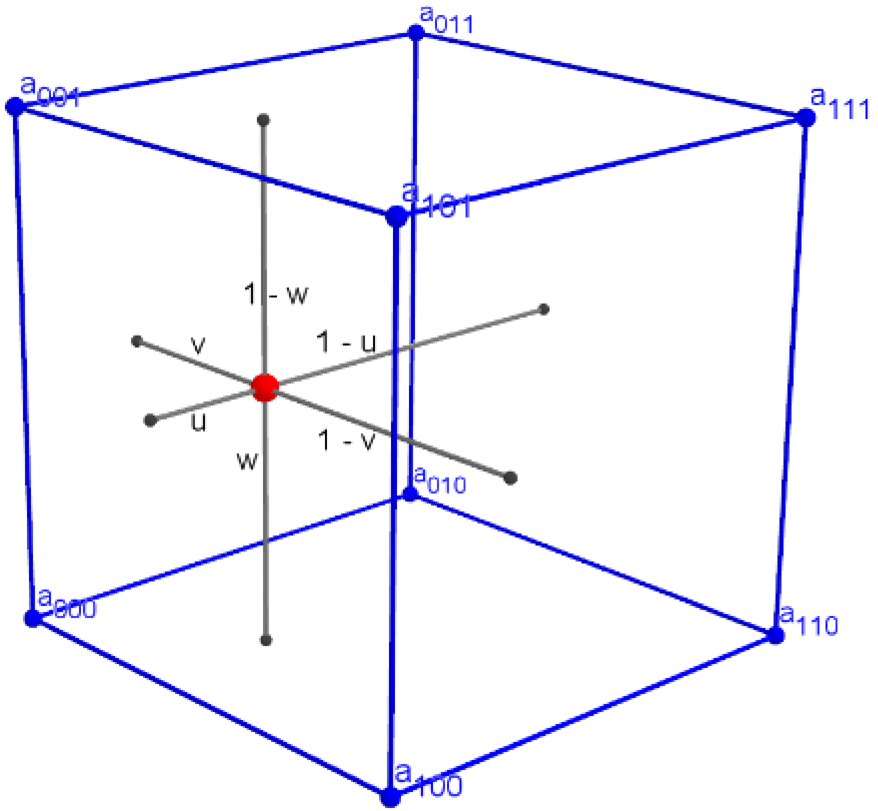


$$d = \max(d_{torus}, d_{sphere})$$

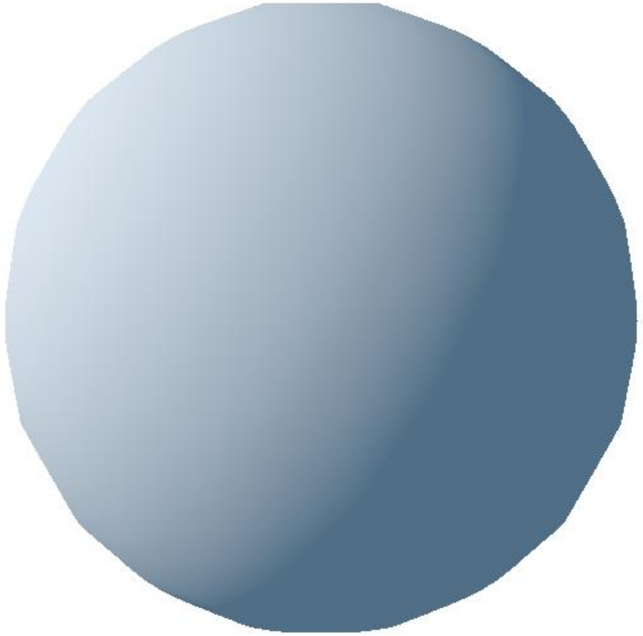


$$d = \max(-d_{torus}, d_{sphere})$$

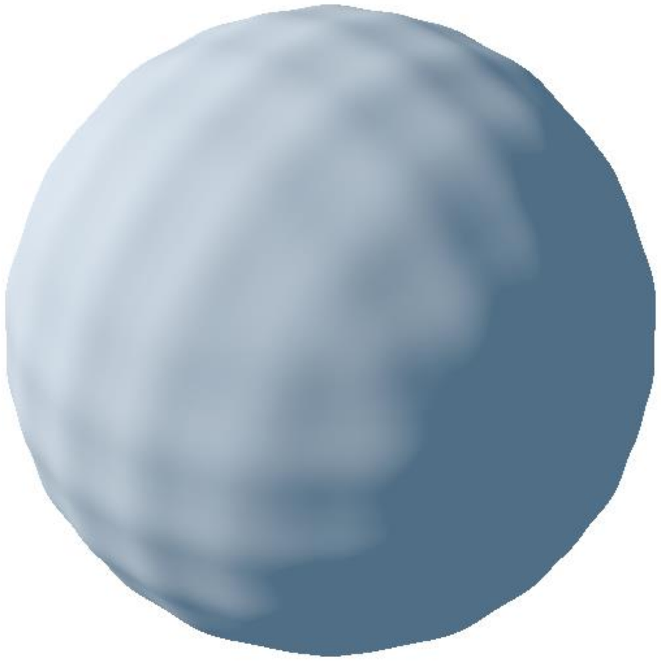
Trilinear filtering



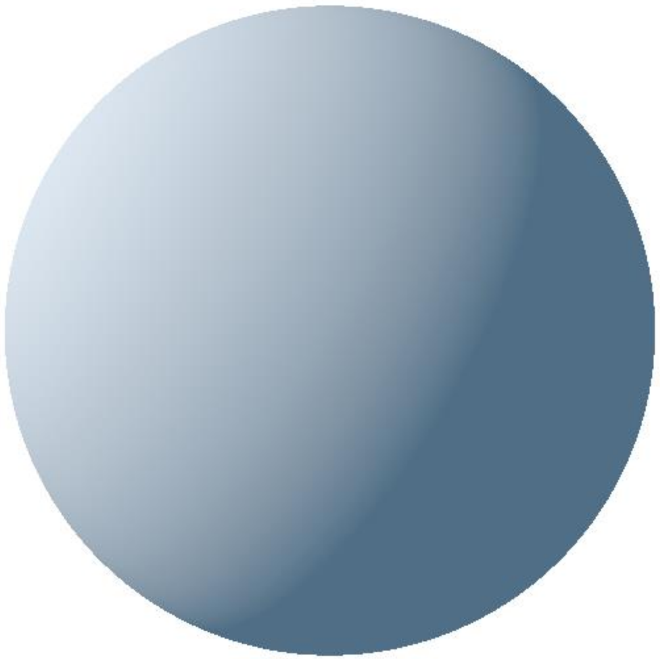
Results



G0



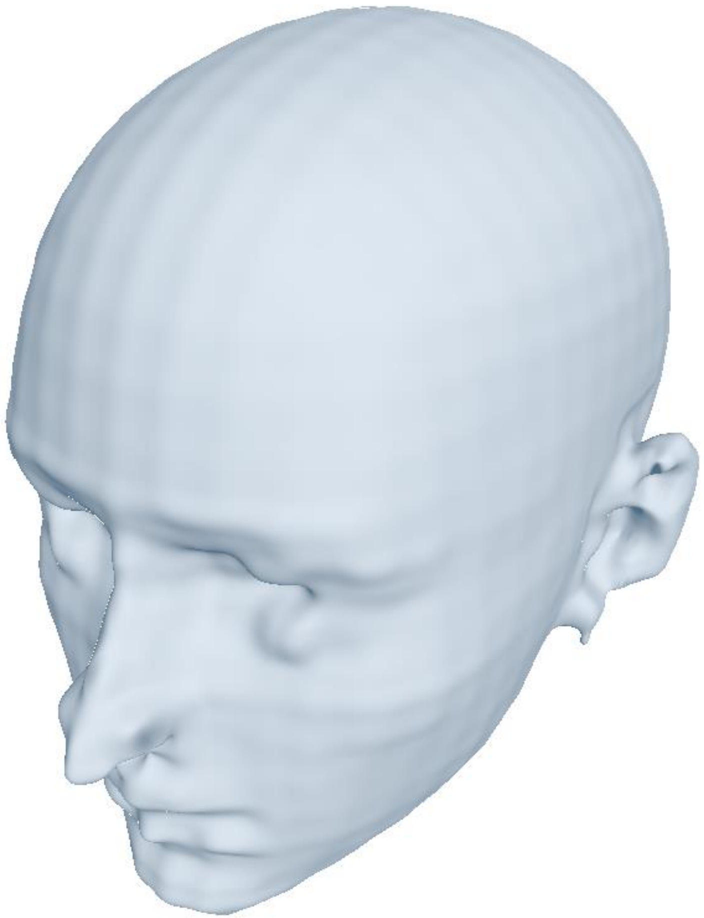
G1



G2

16 x 16 x 16

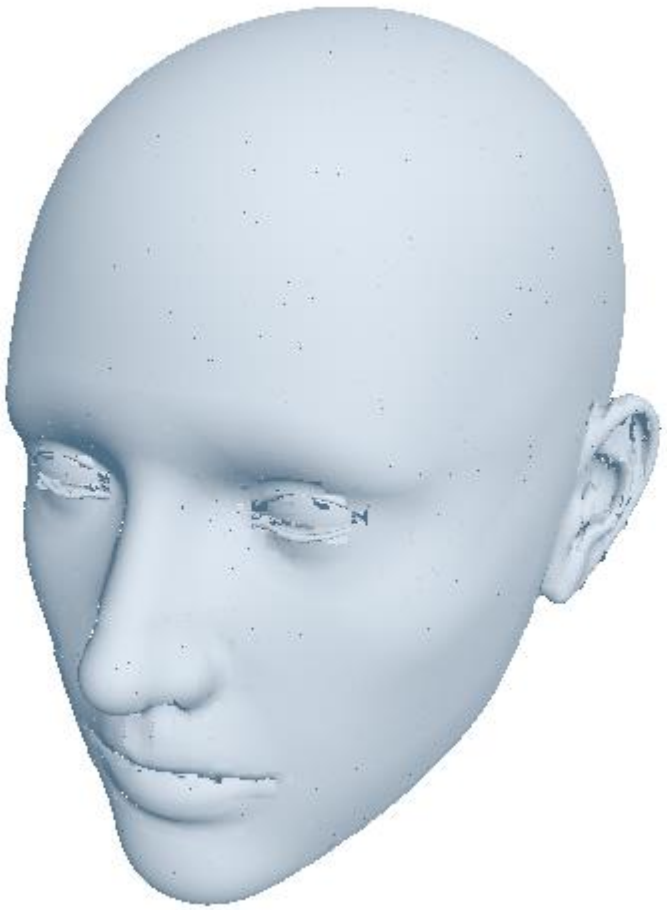
G0



G1

G2

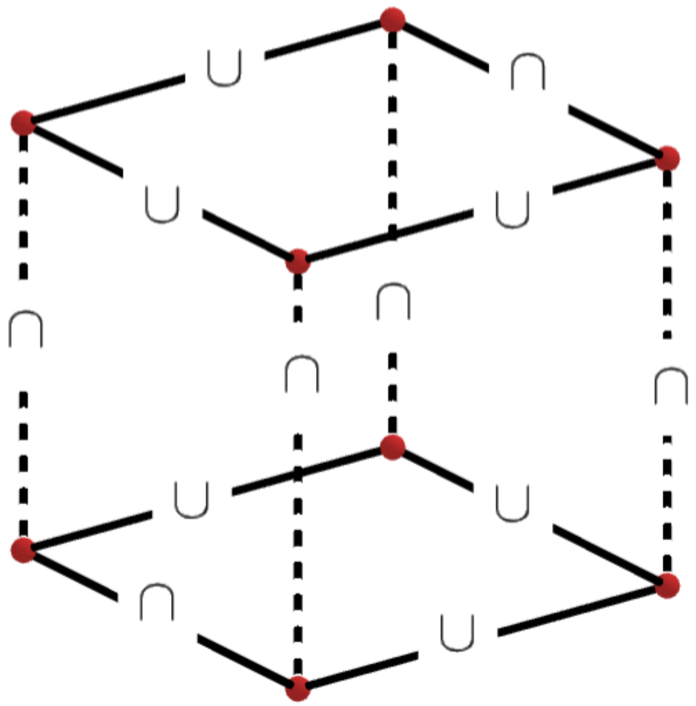
32 x 32 x 32

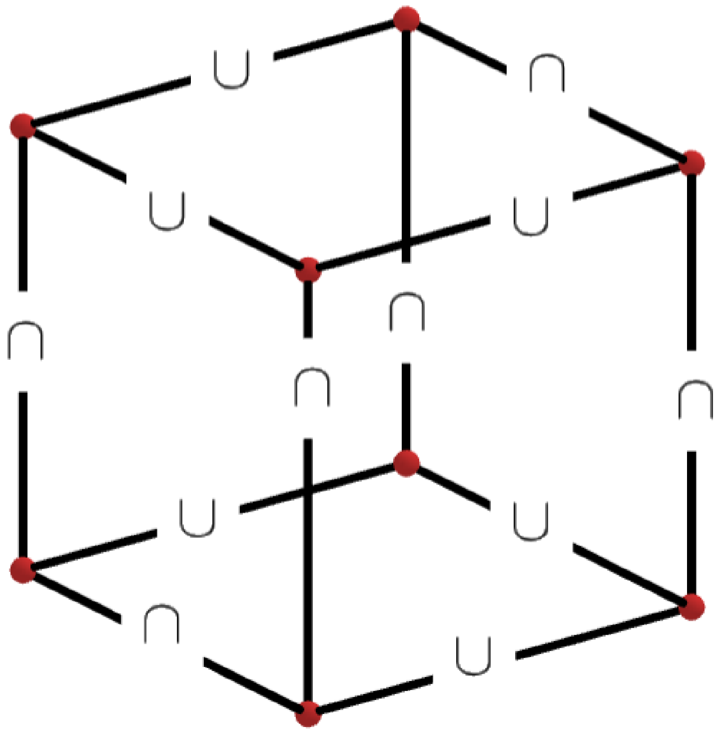


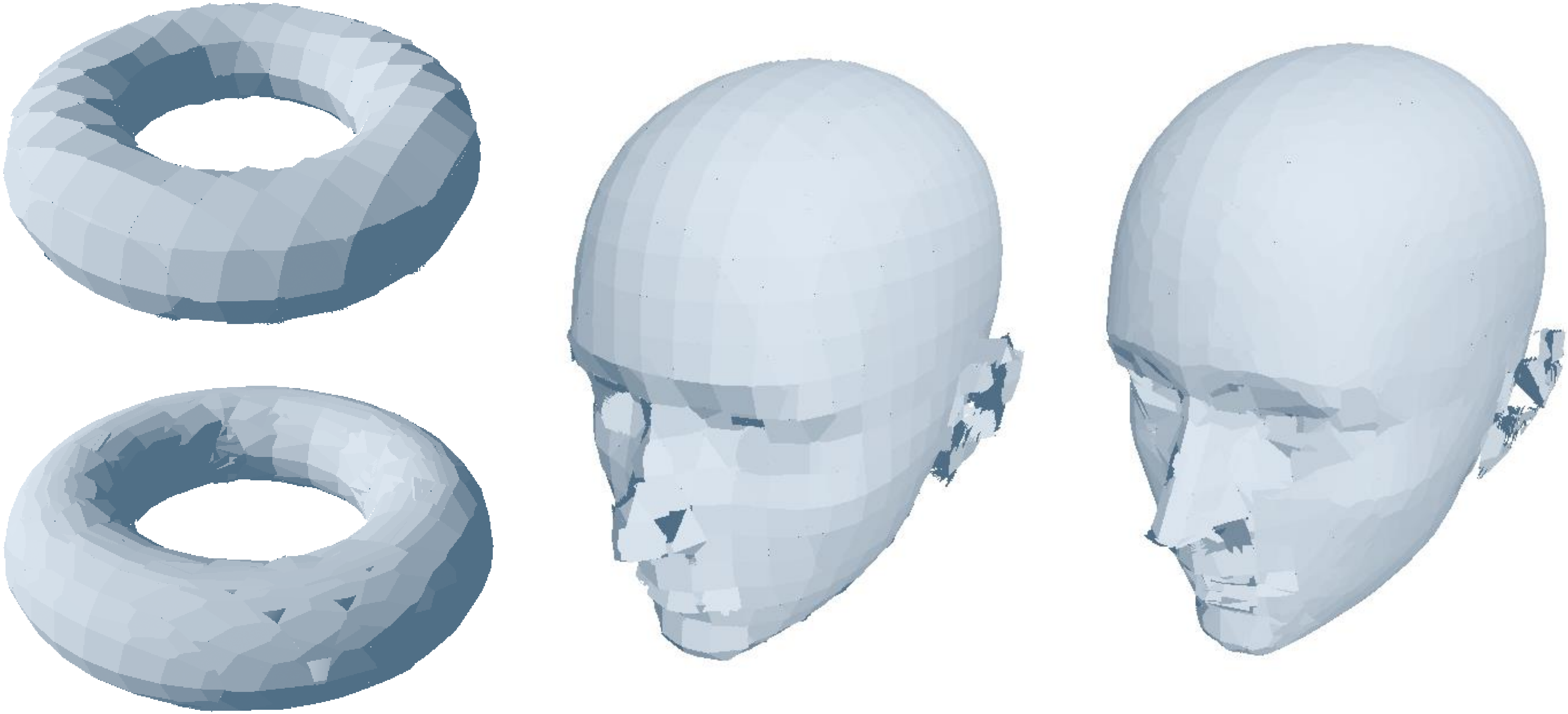
G2 100 x 100 x 100

CSG 3D

- A square, where the vertexes represent the **8 closest geometries**
- Operations on the edges
- Over 30 different cases – Work in progress
- **Optimistic method**



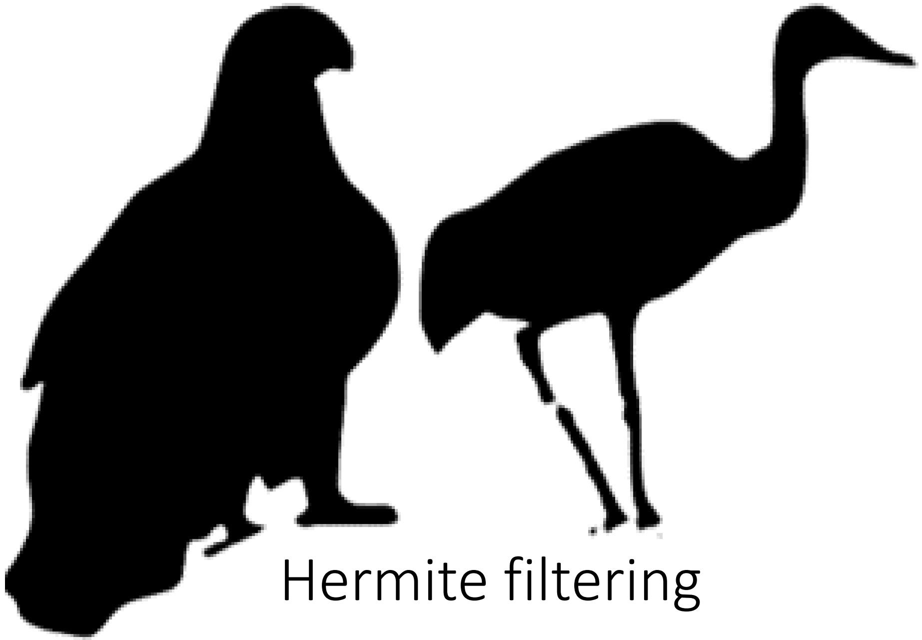




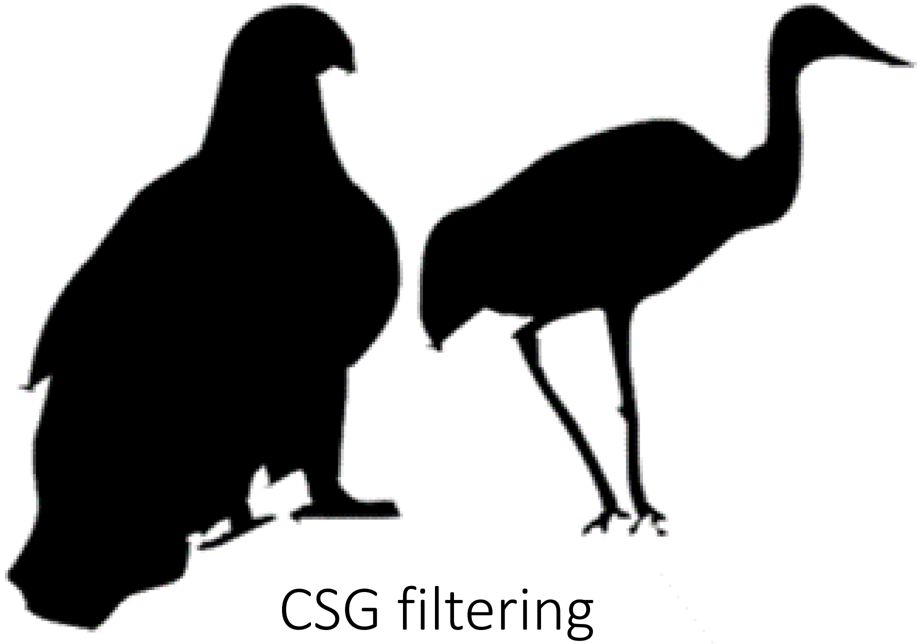
Summary

1. **Geometric fields**
   - Generating
   - Filtering with bi/trilinear method
2. **CSG filtering**
   - G1 2D field
3. **Work in progress: CSG filtering in 3D**

Thank you for the attention!



Hermite filtering



CSG filtering

Sources

- Róbert Bán és Gábor Valasek. „Geometric Distance Fields of Plane Curves”. Acta Cybernetica 25.2 (2021), 187–203. old. DOI: 10.14232/actacyb.289248. URL: https://cyber.bibl.u-szeged.hu/index.php/actcybern/article/view/4192.
- Vas Lia. Surfaces. Accessed: 2023-05-13. URL: https://liavas.net/courses/ math430/files/Surfaces_part3.pdf.
- David Eberly. Fitting 3D Data with a Torus. Accessed: 2023-05-13. URL: https://www.geometrictools.com/Documentation/TorusFitting.pdf.
- Jack Goldfeather és Victoria Interrante. „A Novel Cubic-Order Algorithm for Approximating Principal Direction Vectors”. ACM Trans. Graph. 23.1 (2004), 45–63. ISSN: 0730-0301. DOI: 10.1145/966131.966134. URL: https://doi. org/10.1145/966131.966134.
- Nelson Max. „Improved accuracy when building an orthonormal basis”. Journal of Computer Graphics Techniques (JCGT) 6.1 (2017), 9–16. old. ISSN: 2331-7418. URL: http://jcgt.org/published/0006/01/02/.
- John C. Hart. „Sphere tracing: a geometric method for the antialiased ray tracing of implicit surfaces”. The Visual Computer 12.10 (1996), 527–545. old. ISSN: 1432- 2315. URL: https://doi.org/10.1007/s003710050084.
- Róbert Bán és Gábor Valasek. „Higher Order Algebraic Signed Distance Fields”. Computer-Aided Design and Applications 20.5 (2023), 1005–1028. old. URL: https://cad-journal.net/files/vol_20/CAD__20(5)__2023__1005 - 1028.pdf.
- D. P. Hardin, T. J. Michaels és E. B. Saff. A Comparison of Popular Point Configurations on S 2 . 2016. arXiv: 1607.04590 [math.NA].
- Inigo Quilez. 2D distance functions. Accessed: 2023-05-15. URL: https://iquilezles.org/articles/distfunctions2d/.