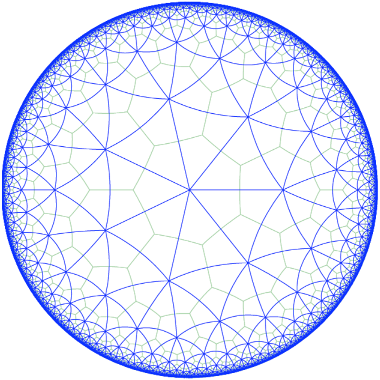
Emerging Directions in Deep Learning

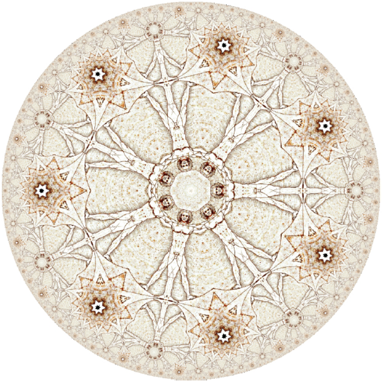
Curvatures

Mihai-Sorin Stupariu, 2024-2025

Motivation

Papers: graph embedding in the hyperbolic space. Why?

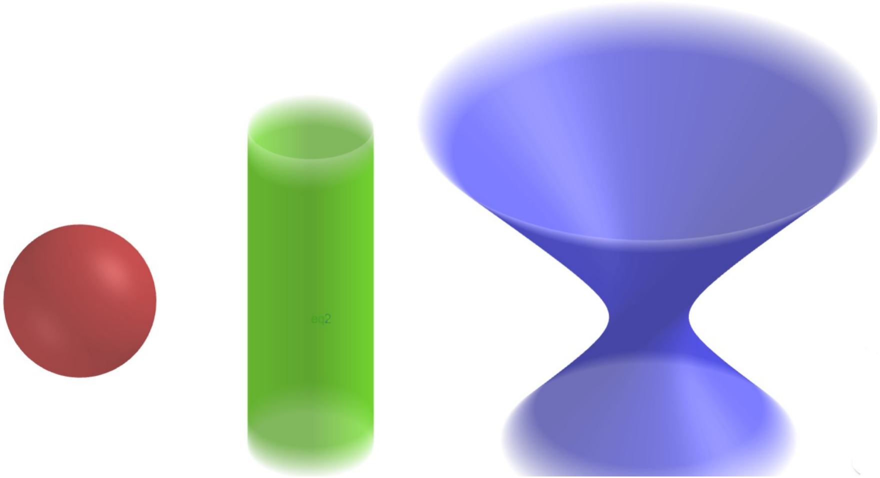




Source: Krioukov et al., 2010

Motivation

Is the geometry of these surfaces (2-dimensional manifolds) different?



Curvature of a parametrized plane curve

Consider $c : I \to \mathbb{R}^2$, $c(t) = (x(t), y(t))$ a plane smooth parametrized curve. The curvature of c at $c(t)$ is

$$\kappa(t) = \frac{|x'(t)y''(t) - x''(t)y''(t)}{(x'(t)^2 + (y'(t)^2)^{\frac{3}{2}}}.$$

Compute the curvature for $c(t) = (\cos(t), \sin(t))$ and then for $c_1(t) = (\cos(2t), \sin(2t))$.

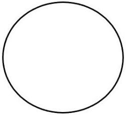
Curvature of a parametrized plane curve

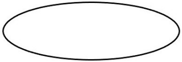
Consider $c : I \to \mathbb{R}^2$, $c(t) = (x(t), y(t))$ a plane smooth parametrized curve. The curvature of c at $c(t)$ is

$$\kappa(t) = \frac{|x'(t)y''(t) - x''(t)y''(t)}{(x'(t)^2 + (y'(t)^2)^{\frac{3}{2}}}.$$

Compute the curvature for $c(t) = (\cos(t), \sin(t))$ and then for $c_1(t) = (\cos(2t), \sin(2t))$.

What about the curvature of these curves?





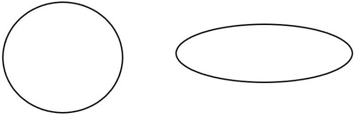
Curvature of a parametrized plane curve

Consider $c : I \to \mathbb{R}^2$, $c(t) = (x(t), y(t))$ a plane smooth parametrized curve. The curvature of c at $c(t)$ is

$$\kappa(t) = \frac{|x'(t)y''(t) - x''(t)y''(t)}{(x'(t)^2 + (y'(t)^2)^{\frac{3}{2}}}.$$

Compute the curvature for $c(t) = (\cos(t), \sin(t))$ and then for $c_1(t) = (\cos(2t), \sin(2t))$.

What about the curvature of these curves?



Q: Why should we care about curvature? **A:** Because, under certain assumptions and up to some transformations, it determines the curve.

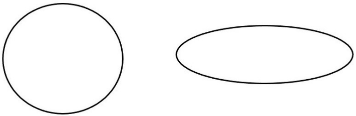
Curvature of a parametrized plane curve

Consider $c : I \to \mathbb{R}^2$, $c(t) = (x(t), y(t))$ a plane smooth parametrized curve. The curvature of c at $c(t)$ is

$$\kappa(t) = \frac{|x'(t)y''(t) - x''(t)y''(t)|}{(x'(t)^2 + (y'(t)^2)^{\frac{3}{2}}}.$$

Compute the curvature for $c(t) = (\cos(t), \sin(t))$ and then for $c_1(t) = (\cos(2t), \sin(2t))$.

What about the curvature of these curves?



Q: Why should we care about curvature? **A:** Because, under certain assumptions and up to some transformations, it determines the curve. More in textbooks such as M. do Carmo's book, T. Shifrin's course notes

Gaussian and mean curvature of a parametrized surface

- Parameterization $(u, v) \mapsto \varphi(u, v) \in \mathbb{R}^3$ of a parametrized surface (surface embedded in $\mathbb{R}^3$). The **Gaussian curvature** at $P = \varphi(u, v)$ can be computed as follows:

$$\kappa_G(P) = \frac{eg - f^2}{EG - F^2}.$$

Gaussian and mean curvature of a parametrized surface

- Parameterization $(u, v) \mapsto \varphi(u, v) \in \mathbb{R}^3$ of a parametrized surface (surface embedded in $\mathbb{R}^3$). The **Gaussian curvature** at $P = \varphi(u, v)$ can be computed as follows:

$$\kappa_G(P) = \frac{eg - f^2}{EG - F^2}.$$

- The coefficients of the first (E, F, G) and of the second (e, f, g) fundamental form involve partial derivatives

$$E = \left\| \frac{\partial \varphi}{\partial u} \right\|^2, \quad F = \left\langle \frac{\partial \varphi}{\partial u}, \frac{\partial \varphi}{\partial v} \right\rangle, \quad G = \left\| \frac{\partial \varphi}{\partial v} \right\|^2,$$

$$e = \left\langle N, \frac{\partial^2 \varphi}{\partial u^2} \right\rangle, \quad f = \left\langle N, \frac{\partial^2 \varphi}{\partial u \partial v} \right\rangle, \quad g = \left\langle N, \frac{\partial^2 \varphi}{\partial v^2} \right\rangle.$$

Gaussian and mean curvature of a parametrized surface

- Parameterization $(u, v) \mapsto \varphi(u, v) \in \mathbb{R}^3$ of a parametrized surface (surface embedded in $\mathbb{R}^3$). The **Gaussian curvature** at $P = \varphi(u, v)$ can be computed as follows:

$$\kappa_G(P) = \frac{eg - f^2}{EG - F^2}.$$

- The coefficients of the first (E, F, G) and of the second (e, f, g) fundamental form involve partial derivatives

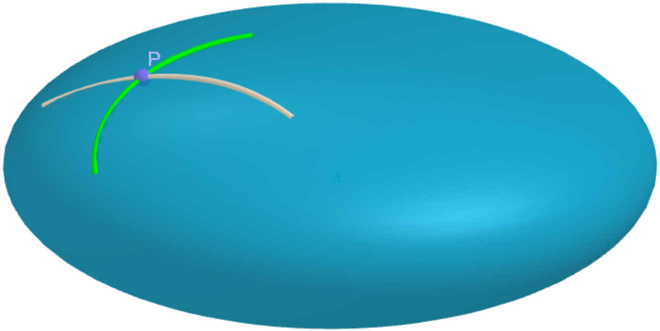
$$E = \left\| \frac{\partial \varphi}{\partial u} \right\|^2, \quad F = \left\langle \frac{\partial \varphi}{\partial u}, \frac{\partial \varphi}{\partial v} \right\rangle, \quad G = \left\| \frac{\partial \varphi}{\partial v} \right\|^2,$$

$$e = \left\langle N, \frac{\partial^2 \varphi}{\partial u^2} \right\rangle, \quad f = \left\langle N, \frac{\partial^2 \varphi}{\partial u \partial v} \right\rangle, \quad g = \left\langle N, \frac{\partial^2 \varphi}{\partial v^2} \right\rangle.$$

- Try to compute it for the sphere $(u, v) \mapsto (\cos(u)\cos(v), \cos(u)\sin(v), \sin(u))$ and for the cylinder $(u, v) \mapsto (\cos(u), \sin(u), v)$.

Relationship to the local geometry of the surface

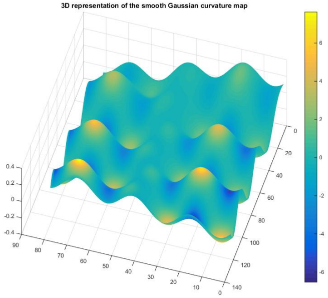
$\frac{\partial \varphi}{\partial u}$, $\frac{\partial \varphi}{\partial v}$ are spanning the tangent space at P
N is the normal vector at P

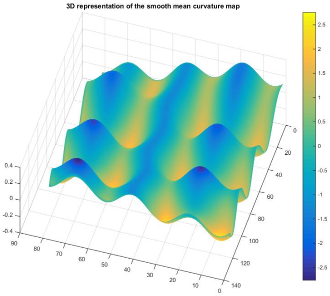


Curvature maps for the Gaussian and for the mean curvature

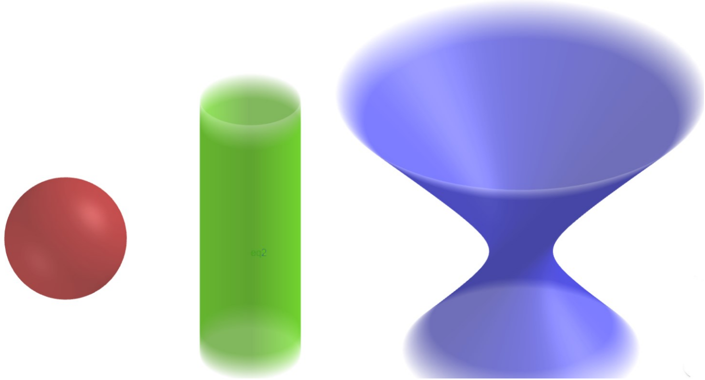
The surface is given by

$$\varphi(u,v) = 0.2\sin(u)\sin(v^{1/5}) - 0.1\cos(5u),\ \ u \in [1,5],\ v \in [1,7].$$





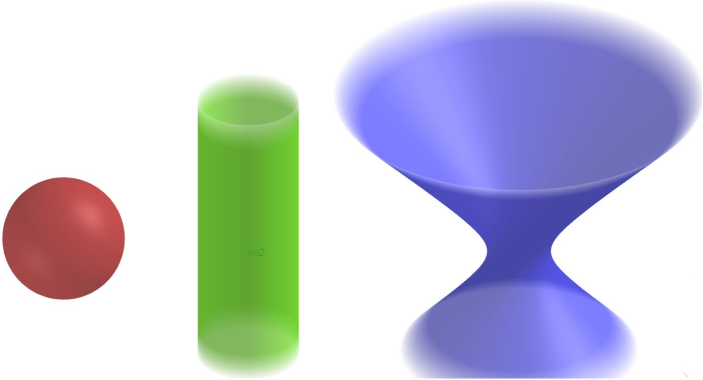
Gaussian curvature for sphere, cylinder and hyperbolid



Positive κ_G Zero κ_G Negative κ_G

(at any point, for each surface
speak about **elliptic** / **planar** / **hyperbolic** points)

Gaussian curvature for sphere, cylinder and hyperbolid



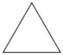
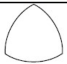
Positive κ_G Zero κ_G Negative κ_G

(at any point, for each surface
speak about **elliptic** / **planar** / **hyperbolic** points)

Could you imagine a surface that has all types of points?

Instead of conclusion

TABLE I: Characteristic properties of Euclidean, spherical, and hyperbolic geometries. *Parallel lines* is the number of lines that are parallel to a line and that go through a point not belonging to this line, and $\zeta = \sqrt{|K|}$.

Property	Euclidean	Spherical	Hyperbolic
Curvature K	0	> 0	< 0
Parallel lines	1	0	∞
Triangles are	normal	thick	thin
Shape of triangles			
Sum of $\triangle$ angles	π	$> \pi$	$< \pi$
Circle length	$2\pi r$	$2\pi \sin \zeta r$	$2\pi \sinh \zeta r$
Disk area	$2\pi r^2/2$	$2\pi(1 - \cos \zeta r)$	$2\pi(\cosh \zeta r - 1)$

Source: Krioukov et al., 2010

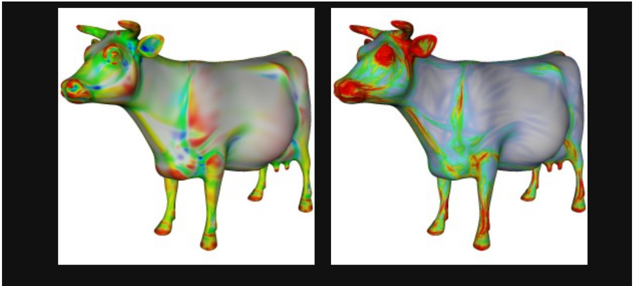
In the hyperbolic model, the length of circles increases exponentially (similar to the number of nodes at a given distance with respect to the root in a *b*-ary tree).

Use of curvatures in relationship to ML approaches

Examples:
Medical imaging
Facial expression

Discrete representations - triangle meshes

Instead of smooth surfaces, discrete representations are used: triangle / polygon meshes.
Curvatures maps for such meshes are widely used:

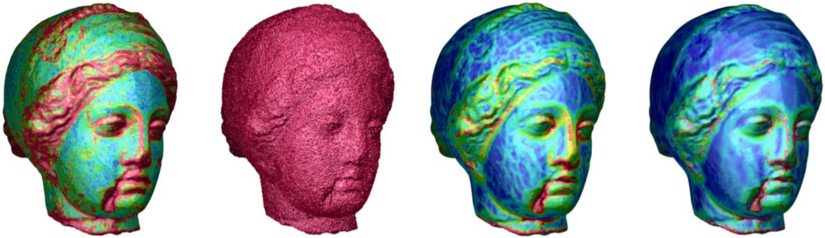


Visualization of the curvatures (left) and derivative of curvature (right) estimated on a 3D mesh

Source: Rusinkiewicz, 2004

Discrete representations - triangle meshes

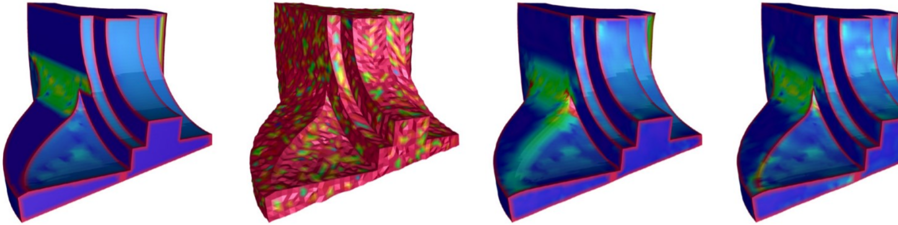
Instead of smooth surfaces, discrete representations are used: triangle / polygon meshes.
Curvatures maps for such meshes are widely used:



Source: Hildebrandt and Polthier, 2004

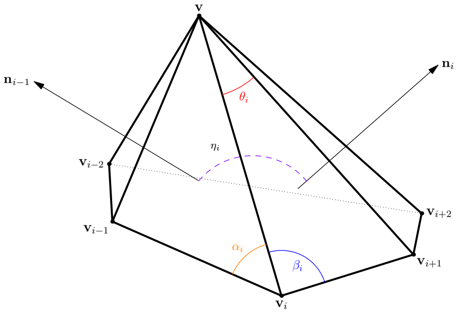
Discrete representations - triangle meshes

Instead of smooth surfaces, discrete representations are used: triangle / polygon meshes.
Curvatures maps for such meshes are widely used:



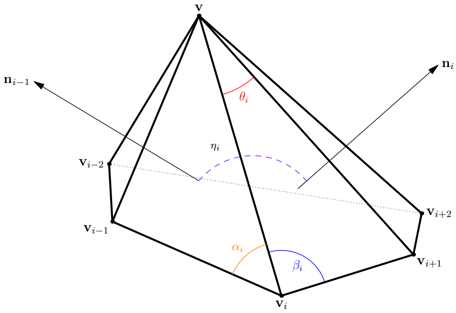
Source: Hildebrandt and Polthier, 2004

Local structure. Notation: 1-ring $\mathcal{N}_{\mathbf{v}}$



Geometric elements around a vertex $\mathbf{v}$:

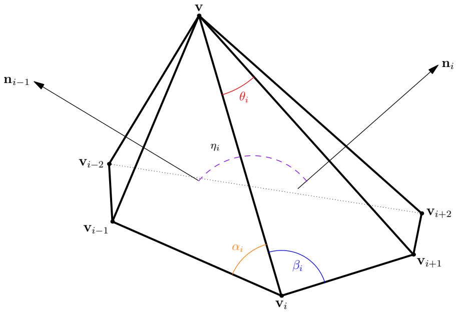
Local structure. Notation: 1-ring $\mathcal{N}_{\mathbf{v}}$



Geometric elements around a vertex **v**:

- Edges / faces incident to **v** (or associated measures – lengths, areas).

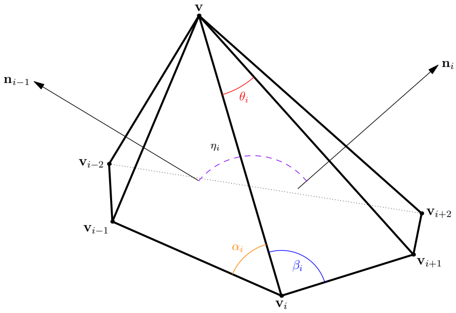
Local structure. Notation: 1-ring $\mathcal{N}_{\mathbf{v}}$



Geometric elements around a vertex $\mathbf{v}$:

- Edges / faces incident to $\mathbf{v}$ (or associated measures – lengths, areas).
- Angles $(\theta_i)_i$ between edges incident to $\mathbf{v}$.

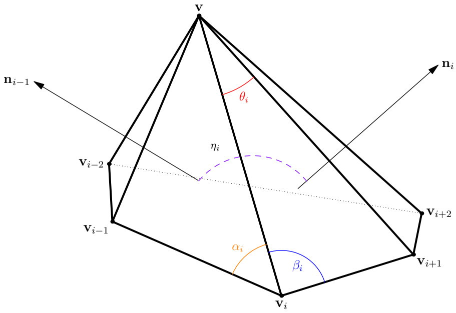
Local structure. Notation: 1-ring $\mathcal{N}_{\mathbf{v}}$



Geometric elements around a vertex $\mathbf{v}$:

- Edges / faces incident to $\mathbf{v}$ (or associated measures – lengths, areas).
- Angles $(\theta_i)_i$ between edges incident to $\mathbf{v}$.
- Angles $(\eta_i)_i$ between normals of faces incident to $\mathbf{v}$.

Local structure. Notation: 1-ring $\mathcal{N}_{\mathbf{v}}$



Geometric elements around a vertex **v**:

- Edges / faces incident to **v** (or associated measures – lengths, areas).
- Angles $(\theta_i)_i$ between edges incident to **v**.
- Angles $(\eta_i)_i$ between normals of faces incident to **v**.
- Angles $(\alpha_i)_i, (\beta_i)_i$ between edges of the 1-ring that are not incident to **v**.

Method 1: Gauss-Bonnet scheme (1) GB1

- **Gaussian curvature at v**

$$K_{\mathbf{v}} = \frac{2\pi - \sum_{\mathbf{v}_i \in \mathcal{N}_{\mathbf{v}}} \theta_i}{\frac{1}{3}A}, \tag{1}$$

where $2\pi - \sum_{\mathbf{v}_i \in \mathcal{N}_{\mathbf{v}}} \theta_i$ is the angular defect at **v**, and A is the total area of the triangles in the 1-ring neighborhood of v

Method 1: Gauss-Bonnet scheme (1) GB1

- **Gaussian curvature at v**

$$K_{\mathbf{v}} = \frac{2\pi - \sum_{\mathbf{v}_i \in \mathcal{N}_{\mathbf{v}}} \theta_i}{\frac{1}{3} A}, \tag{1}$$

  where $2\pi - \sum_{\mathbf{v}_i \in \mathcal{N}_{\mathbf{v}}} \theta_i$ is the angular defect at **v**, and A is the total area of the triangles in the 1-ring neighborhood of v

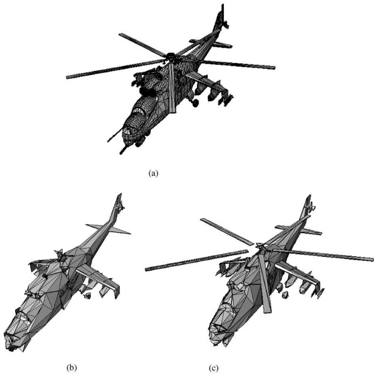
- **Mean curvature** at **v**

$$H_{\mathbf{v}} = \frac{\frac{1}{4} \sum_{\mathbf{v}_i \in \mathcal{N}_{\mathbf{v}}} \|\overrightarrow{\mathbf{v}\mathbf{v}_i}\| \eta_i}{\frac{1}{3} A} \tag{2}$$

  (measures the variation of the normals along the edges incident to **v**)

Method 1: Gauss-Bonnet scheme (1) GB1

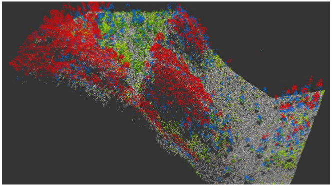
Used by [Dyn et al., 2001]; [Kim et al., 2002] for simplifying triangle meshes

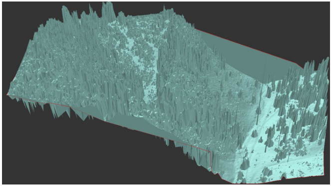


Helicopter model. (a) Original. (b), (c) Simplified versions. In (c) the discrete curvatures were used.
Source: [S.J. Kim, C.H. Kim, D. Levin, *Computers & Graphics*, 2002]

Problem statement

Vegetation structures (e.g. trees) are visible in a high density point cloud and in the associated triangulation.





Methodology

- The high resolution LiDAR point cloud was used; a 2.5 Delaunay triangulation was generated direclty from the original point cloud ((x, y)-duplicates, due to vegetation, were eliminated).

Methodology

- The high resolution LiDAR point cloud was used; a 2.5 Delaunay triangulation was generated direclty from the original point cloud ((x, y)-duplicates, due to vegetation, were eliminated).
- Compute the values of discrete curvatures for the triangle mesh associated to the original point cloud. For each discrete curvature get a grid of averaged curvatures (cell size 1m).

Methodology

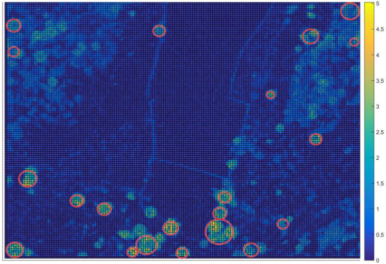
- The high resolution LiDAR point cloud was used; a 2.5 Delaunay triangulation was generated direclty from the original point cloud ((x, y)-duplicates, due to vegetation, were eliminated).
- Compute the values of discrete curvatures for the triangle mesh associated to the original point cloud. For each discrete curvature get a grid of averaged curvatures (cell size 1m).
- Construct regular grids (cell size 1m) of curvatures. For each method and cell one considered the vertices lying in that cell and then one computed the average value of the curvatures corresponding to these vertices.

Methodology

- The high resolution LiDAR point cloud was used; a 2.5 Delaunay triangulation was generated direclty from the original point cloud ((x, y)-duplicates, due to vegetation, were eliminated).
- Compute the values of discrete curvatures for the triangle mesh associated to the original point cloud. For each discrete curvature get a grid of averaged curvatures (cell size 1m).
- Construct regular grids (cell size 1m) of curvatures. For each method and cell one considered the vertices lying in that cell and then one computed the average value of the curvatures corresponding to these vertices.
- Use pattern recognition techniques (the Hough transform, implemented in Matlab, sensitivity factor 0.85) for detecting circles: horizontal projections of tree crowns usually yield circular shapes.

Tree detection — results

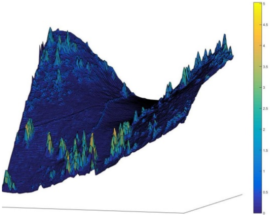
LiDAR point cloud (colours represent height above ground, in particular trees are coloured in red).



Grid generated by using the mean curvature, as provided by the shape operator method. The red circles represent trees detected

Tree detection – results

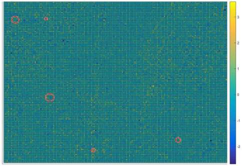
The point cloud (3D representation).



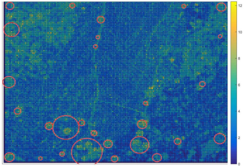
Grid of mean curvatures for SO (3D representation).

Comparisons – mean curvature grids

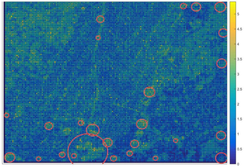
Point cloud.



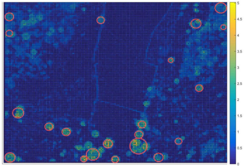
ET



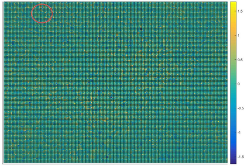
GB1



GB2

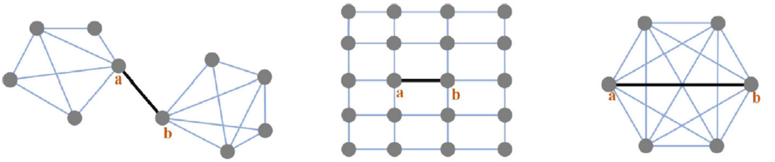


SO



TA

Curvatures for graphs



(a) Negative Curvature (b) Zero Curvature (c) Positive Curvature

Source: Li et al., 2022