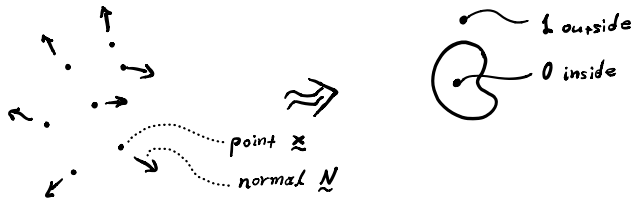
Poisson Surface Reconstruction

is an *implicit* approach, meaning that it does not use the points directly in the surface reconstruction process, but rather finds a function that has its zero level-set coincide with the surface we aim to reconstruct.

Kahzdan's version uses an indicator (characteristic) function.

There is an internal assumption that this function is smooth.



Goal: Find a function $\chi(\underset{\sim}{x})$ whose gradient best matches the input normals expressed as a Normal field $\underset{\sim}{N}(\underset{\sim}{x})$:

$$\chi = \arg\min_{\chi} \int \| \underset{\sim}{\nabla}\chi(\underset{\sim}{x}) - \underset{\sim}{N}(\underset{\sim}{x}) \|_2^2 \, d\chi \qquad \xrightarrow{\text{How?..}} \qquad \Delta\chi = \nabla \cdot \underset{\sim}{N}$$

To obtain χ we need to solve a Poisson equation

(laplacian of the function = divergence of the normals)

The "marching cubes" part is related to the process of using the 0-level-set to obtain the vertices of the mesh.

Recall:

Gradient : $\nabla f = \underset{\sim}{F} = \begin{pmatrix} \partial f/\partial x \\ \partial f/\partial y \\ \partial f/\partial z \end{pmatrix}$

Divergence : $\underset{\sim}{\nabla} \cdot \underset{\sim}{N} = \frac{\partial}{\partial x} N_x + \frac{\partial}{\partial y} N_y + \frac{\partial}{\partial z} N_z$

Laplacian : $\Delta\chi = \nabla^2\chi = \nabla \cdot \underset{\sim}{\nabla}\chi$

$= (\text{div } (\text{grad } \chi)) = \frac{\partial^2}{\partial x^2}\chi + \frac{\partial^2}{\partial y^2}\chi + \frac{\partial^2}{\partial z^2}\chi$

Poisson's Equation (in $\mathbb{R}^3$)

$$\nabla^2 g(x,y,z) = f(x,y,z)$$ ← given

$$= \frac{\partial^2 g(x,y,z)}{(\partial x)^2} + \frac{\partial^2 g(x,y,z)}{(\partial y)^2} + \frac{\partial^2 g(x,y,z)}{(\partial z)^2}$$

+ Boundary conditions for $g(x,y,z)$

} goal: Find the function $g(x,y,z)$

In our case: $\nabla^2 \tilde{\chi}(x,y,z) = \nabla \cdot \underset{\sim}{V}(x,y,z)$

($\tilde{\chi}$: (convolved?) Indicator function; $\underset{\sim}{V}$: Normal vector field; known "g")

What we have:

Point samples ⟶ set S

Octree $\mathcal{O}$ of depth D and leaf nodes o

We will approximate $\tilde{\chi}(x,y,z)$ using a linear combination of basis functions

space: $\mathcal{F}_{o,F} \equiv \text{span}(F_o)$

each octree node o is associated with a base function

$$F_o(\underset{\sim}{q}) = F\left(\frac{1}{o.w}(\underset{\sim}{q} - o.\underset{\sim}{c})\right) \cdot \frac{1}{(o.w)^3}$$

(any x,y,z position) ; (global) base function

where $o.\underset{\sim}{c}$ is the node's center, $o.w$ its width,

$$F(\underset{\sim}{q}) = F(x,y,z) = \big(B(x)\,B(y)\,B(z)\big)^{*n=3} \qquad B(t) = \begin{cases} 1 & |t|<0.5 \\ 0 & \text{otherwise} \end{cases}$$

$*n=3$: 3rd convolution..

Continuous approximation of the normal field $\underset{\sim}{V}$:

$$\underset{\sim}{V}(\underset{\sim}{q}) = \sum_{s \in S} \left\{ \sum_{o \in Nghbr_D(s)} \left\{ \alpha_{o,s}\, F_o(\underset{\sim}{q})\, \underset{\sim}{s.N} \right\} \right\}$$

} This way we can obtain $\underset{\sim}{V}$ at any position $\underset{\sim}{q}$

$Nghbr_D(s)$: 8 depth-D nodes closest to $\underset{\sim}{s.p}$

$\alpha_{o,s}$: Trilinear interpolation weights

§4.5 improves this for non-uniform samples

Solving $\nabla^2 \tilde{\chi} = \nabla \cdot \underset{\sim}{V}$ to obtain $\tilde{\chi}(\underset{\sim}{q})$

Using a linear combination of the function bases $F_o(\underset{\sim}{q})$,

our solution will belong in the space $\mathcal{F}_{O,F}$.

Issue: Even though $\tilde{\chi}(\underset{\sim}{q}) \in \mathcal{F}_{O,F}$ & $\underset{\sim}{V}(\underset{\sim}{q}) \in \mathcal{F}_{O,F}$ by their definitions,

$\nabla^2 \tilde{\chi}(\underset{\sim}{q})$ and $\nabla \cdot \underset{\sim}{V}(\underset{\sim}{q})$ might $\notin \mathcal{F}_{O,F}$

Workaround: minimize the difference of their projections (which $\in \mathcal{F}_{O,F}$ ...)
(hoping that this minimizes their actual difference?)

$$\tilde{\chi}(\underset{\sim}{q}) = \underset{\tilde{\chi}}{\operatorname{argmin}} \sum_{o \in \mathcal{O}} \left\{ \left\| \left\langle \nabla^2 \tilde{\chi} - (\nabla \cdot \underset{\sim}{V}) \,,\, F_o \right\rangle \right\|^2 \right\}$$

wait... these aren't vectors!
· function projection?
· function norm?

$$= \underset{\tilde{\chi}}{\operatorname{argmin}} \sum_{o \in \mathcal{O}} \left\{ \left\| \left\langle \nabla^2 \tilde{\chi}, F_o \right\rangle - \left\langle (\nabla \cdot \underset{\sim}{V}) \,,\, F_o \right\rangle \right\|^2 \right\}$$

at which $\underset{\sim}{q}$?

Discretization to solve: We define $\underset{\sim}{v} \in \mathbb{R}^{|\mathcal{O}|}$ with $v_i = \left\langle \nabla \cdot \underset{\sim}{V}, F_i \right\rangle$

Let $\tilde{\chi} = \sum_{o \in \mathcal{O}} \{ x_o F_o \}$ — o-th component of unknown vector $\underset{\sim}{x} \in \mathbb{R}^{|\mathcal{O}|}$

Define matrix $\underline{L} \in \mathbb{R}^{|\mathcal{O}| \times |\mathcal{O}|}$ with $L_{ij} = \left\langle \frac{\partial^2 F_i}{\partial x^2}, F_j \right\rangle + \left\langle \frac{\partial^2 F_i}{\partial y^2}, F_j \right\rangle + \left\langle \frac{\partial^2 F_i}{\partial z^2}, F_j \right\rangle$

$$\underset{\sim}{x} = \underset{\underset{\sim}{x}}{\operatorname{argmin}} \left\| \underline{L} \underset{\sim}{x} - \underset{\sim}{v} \right\|_2^2$$

→ we can then compute $\tilde{\chi}(\underset{\sim}{q})$

! what if we apply weights to remove outliers ??

$$\partial \tilde{\mathcal{M}} \triangleq \left\{ \underset{\sim}{q} \in \mathbb{R}^3 \,\middle|\, \tilde{\chi}(\underset{\sim}{q}) = \gamma \right\} \text{ with } \gamma = \frac{1}{|S|} \sum_{s \in S} \left\{ \tilde{\chi}(\underset{\sim}{s}.p) \right\}$$

now we can find values for $\underset{\sim}{q}$ that satisfy this...
use "marching cubes" to get the final mesh...