Poisson Surface Reconstruction

Siddhartha Chaudhuri

http://www.cse.iitb.ac.in/~cs749
Recap: Implicit Function Approach

- Define a function with positive values inside the model and negative values outside
Recap: Implicit Function Approach

- Define a function with positive values inside the model and negative values outside
- Extract the zero-set

Slides adapted from Kazhdan, Bolitho and Hoppe
Recap: Key Idea

- Reconstruct the surface of the model by solving for the indicator function of the shape

\[
\chi_M(p) = \begin{cases} 
1 & \text{if } p \in M \\
0 & \text{if } p \notin M
\end{cases}
\]

In practice, we define the indicator function to be -1/2 outside the shape and 1/2 inside, so that the surface is the zero level set. We also smooth the function a little, so that the zero set is well defined.

Slides adapted from Kazhdan, Bolitho and Hoppe
Recap: Challenge

- How to construct the indicator function?

Oriented points \[ \rightarrow \] Indicator function \[ \chi_M \]

Slides adapted from Kazhdan, Bolitho and Hoppe
Recap: Gradient Relationship

- There is a relationship between the normal field at the shape boundary, and the gradient of the (smoothed) indicator function.

Oriented points \( \nabla \chi_M \)
Operators

• Let's look at a 1D function $f : \mathbb{R} \to \mathbb{R}$

  • It has a first derivative given by

    $$\frac{df}{dx} = \lim_{h \to 0} \frac{f(x+h) - f(x)}{h}$$

  • ... a second derivative, and a third...

    $$\frac{d^2 f}{dx^2} = \frac{d}{dx} \frac{d}{dx} f$$
    $$\frac{d^3 f}{dx^3} = \frac{d}{dx} \frac{d}{dx} \frac{d}{dx} f$$

  • $\frac{d}{dx}$ is a general operation mapping functions to functions: it's called an **operator**

    • In fact, it's a **linear operator**: $\frac{d}{dx}(f+g) = \frac{d}{dx} f + \frac{d}{dx} g$
Variational Calculus

- Imagine we didn't know $f$, but we did know its derivative $g = \frac{df}{dx}$

- Solving for $f$ is, obviously, integration
  \[ f = \int \frac{df}{dx} \, dx = \int g \, dx \]

- But what if $g$ is not analytically integrable?
  - Then we can look for approximate solutions, drawn from some parametrized family of candidate functions
Variational Calculus

• Assume we have a family of functions $F$

• Let's minimize the mean squared approximation error over some interval $\Omega$ and functions $f \in F$

$$\text{minimize} \int_{\Omega} \left| \frac{df}{dx} - g \right|^2 dx$$
Euler-Lagrange Formulation

- **Euler-Lagrange equation**: Stationary points (minima, maxima etc) of a functional of the form

\[ \int_{\Omega} L(x, f(x), f'(x)) \, dx \]

are obtained as solutions \( f \) to the PDE

\[ \frac{\partial L}{\partial f} - \frac{d}{dx} \frac{\partial L}{\partial f'} = 0 \]
Euler-Lagrange Formulation

- **Euler-Lagrange equation**: \( \frac{\partial L}{\partial f} - \frac{d}{dx} \frac{\partial L}{\partial f'} = 0 \)

- In our case, \( L = (f'(x) - g(x))^2 \), so
  \[
  \frac{\partial L}{\partial f} = 0 \quad \frac{\partial L}{\partial f'} = 2(f'(x) - g(x))
  \]
  \[
  \frac{d}{dx} \frac{\partial L}{\partial f'} = 2(f''(x) - g'(x))
  \]

- Substituting, we get (a case of) the 1D Poisson equation:
  \[
  f''' = g' \quad \text{or} \quad \frac{d^2 f}{dx^2} = \frac{dg}{dx}
  \]
Link to Linear Least Squares

• Here, we want to minimize \( \int_{\Omega} (f'(x) - g(x))^2 \, dx \) and end up having to solve

\[
\frac{d}{dx} \frac{d}{dx} f = \frac{d}{dx} g
\]

i.e. the two sides are equal at all points \( x \)

• Let's try to discretize this!
  
  • Sample \( n \) consecutive points \( \{x_i\} \) from \( \Omega \)
    
    – Assume (for simplicity) they're evenly spaced, so \( x_{i+1} - x_i = h \)
  
  • We want to minimize \( \Sigma_i (f'(x_i) - g(x_i))^2 \)
The derivative at $x_i$ can be approximated as

$$f'(x_i) \approx \frac{f_{i+1} - f_i}{h} = \frac{1}{h} \begin{bmatrix} -1 & 1 \end{bmatrix} \begin{bmatrix} f_i \\ f_{i+1} \end{bmatrix}$$

where $f_i$ is shorthand for $f(x_i)$. 
Link to Linear Least Squares

• ... and all the derivatives can be listed in one big matrix multiplication: \( A \mathbf{f} = \mathbf{g} \), where

\[
A = \frac{1}{h} \begin{bmatrix}
-1 & 1 & 0 & \cdots & 0 & 0 \\
0 & -1 & 1 & \cdots & 0 & 0 \\
0 & 0 & -1 & \cdots & 0 & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
0 & 0 & \cdots & 0 & -1 & 1 \\
0 & 0 & \cdots & 0 & 0 & -1 \\
\end{bmatrix} \quad \mathbf{f} = \begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ \vdots \\ f_n \end{bmatrix} \quad \mathbf{g} = \begin{bmatrix} g_1 \\ g_2 \\ g_3 \\ \vdots \\ g_n \end{bmatrix}
\]

• \( \mathbf{f} \) and \( \mathbf{g} \) are discrete approximations of continuous functions \( f \) and \( g \), and \( A \) is a discrete approximation for the continuous derivative operator \( \frac{d}{dx} \)!
Flashback

• Need to solve set of equations $Af = g$ in a least squares sense

$$\text{minimize} \; ||r||^2 = ||g - Af||^2$$

• The directional derivative in direction $\delta f$ is

$$\nabla ||r||^2 \cdot \delta f = 2\delta f^T(A^Tg - A^TAf)$$

• The minimum is achieved when all directional derivatives are zero, giving the normal equations

$$A^TAf = A^Tg$$

• Thought for the (Previous) Day: Compare this equation to the Poisson equation
Link to Linear Least Squares

- **Linear Least Squares:** The $f$ that minimizes $\|Af - g\|^2$ is the solution of $A^TAf = A^Tg$

- **Euler-Lagrange:** The $f$ that minimizes
  \[ \int_\Omega \left( \frac{df}{dx}(x) - g(x) \right)^2 dx \]
  is a solution of
  \[ \frac{d}{dx} \frac{d}{dx} f = \frac{d}{dx} g \]

- Knowing that $A$ is the discrete version of $\frac{d}{dx}$, everything lines up except for the transpose bit.
  - How do we reconcile this?
The derivative at $x_i$ can also be approximated as

$$f'(x_i) \approx \frac{f_i - f_{i-1}}{h} = \frac{1}{h} [-1, 1] \begin{bmatrix} f_{i-1} \\ f_i \end{bmatrix}$$

... and derivatives at all $x_i$ as $B \mathbf{f}$, where

$$B = \frac{1}{h} \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 & 0 \\ -1 & 1 & 0 & \cdots & 0 & 0 \\ -1 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & 0 \\ 0 & 0 & \cdots & -1 & 1 \end{bmatrix}$$

... which is just $-A^T$!

Can rewrite normal equations as $(-A^T)A\mathbf{f} = (-A^T)\mathbf{g}$.
Uniqueness of Solutions

- The discrete operator $A$ we constructed is full-rank (invertible), and gives a unique solution $A^{-1}g$ for $f$.
- But the corresponding continuous problem has multiple solutions (e.g. if $f$ is a solution, $(f + \text{constant})$ is also a solution).

**Explanation:** $Af = g$ implicitly imposes the boundary condition $f_n = -g_n$ (see the last row of the matrix).
  - In higher dimensions, the operator matrix $A$ is non-square (maps scalar field to vector field) and not invertible. The system is overdetermined and we seek least-squares solutions.
Discrete Second Derivative

- Multiplying the matrices, we get the discrete second derivative operator (the 1D Laplacian)

\[ \frac{d^2}{dx^2} = \frac{d}{dx} \frac{d}{dx} \text{ discretized to } (-A^T)A = \frac{1}{h^2} \]

\[ \begin{bmatrix} -2 & 1 & 0 & \cdots & 0 & 0 \\ 1 & -2 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & -2 & 1 \\ 0 & 0 & \cdots & 1 & -2 \end{bmatrix} \]

If you actually do the multiplication, this term is -1 and not -2. This is because our discretization does not correctly model the derivative at the end of the range. If you swap the matrices, the discrepancy occurs in the last element of the product instead.

... which is the same as the Taylor series approximation for the second derivative
In higher dimensions

- We have a function \( f : \mathbb{R}^p \to \mathbb{R}^q \)

- Differential operators (in 3D):
  - **Gradient** (of scalar-valued function): \( \nabla f = \left( \frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}, \frac{\partial f}{\partial z} \right) \)
  - **Divergence** (of vector-valued function): \( \nabla \cdot V = \frac{\partial V_x}{\partial x} + \frac{\partial V_y}{\partial y} + \frac{\partial V_z}{\partial z} \)
  - **Laplacian** (of scalar-valued function): \( \Delta f = \nabla \cdot \nabla f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} + \frac{\partial^2 f}{\partial z^2} \)
In higher dimensions

- We have a function $f: \mathbb{R}^p \to \mathbb{R}^q$

- We can discretize the domain as before, and obtain discrete analogues of the gradient $\nabla (A)$, divergence $\nabla \cdot (-A^T)$ and Laplacian $\Delta = (\nabla \cdot) \nabla (-A^T A)$

- Note that the gradient and divergence matrices are no longer square (more on this next class)
Takeaway

• A continuous variational problem can be approximated by a discrete one
  • Continuous function $\rightarrow$ Discrete vector of values
  • Continuous operator $\rightarrow$ Discrete matrix
  • Function composition $\rightarrow$ Matrix multiplication
  • Euler-Lagrange solution $\rightarrow$ Linear Least Squares

• Rest of this class: Overview of the pipeline of Poisson surface reconstruction

• Next class: The Galerkin approximation for recovering a continuous function from the discrete setup
Implementation

Given the Points:
- Set octree
- Compute vector field
- Compute indicator function
- Extract iso-surface
Implementation: Adaptive Octree

Given the Points:
- Set octree
- Compute vector field
- Compute indicator function
- Extract iso-surface

Slides adapted from Kazhdan, Bolitho and Hoppe
Implementation: Vector Field

Given the Points:

- Set octree
- Compute vector field
  - Define a function space
  - Splat the samples
- Compute indicator function
- Extract iso-surface
Implementation: Vector Field

Given the Points:

- Set octree
- Compute vector field
  - Define a function space
  - Splat the samples
- Compute indicator function
- Extract iso-surface

Slides adapted from Kazhdan, Bolitho and Hoppe
Implementation: Vector Field

Given the Points:

- Set octree
- Compute vector field
  - Define a function space
  - Splat the samples
- Compute indicator function
- Extract iso-surface
Implementation: Vector Field

Given the Points:

- Set octree
- Compute vector field
  - Define a function space
  - Splat the samples
- Compute indicator function
- Extract iso-surface

Slides adapted from Kazhdan, Bolitho and Hoppe
Implementation: Vector Field

Given the Points:

- Set octree
- Compute vector field
  - Define a function basis
  - Splat the samples
- Compute indicator function
- Extract iso-surface

Slides adapted from Kazhdan, Bolitho and Hoppe
Implementation: Vector Field

Given the Points:

- Set octree
- Compute vector field
  - Define a function basis
  - Splat the samples
- Compute indicator function
- Extract iso-surface

Slides adapted from Kazhdan, Bolitho and Hoppe
Implementation: Vector Field

Given the Points:

- Set octree
- Compute vector field
  - Define a function basis
  - Splat the samples
- Compute indicator function
- Extract iso-surface

Slides adapted from Kazhdan, Bolitho and Hoppe
Implementation: Vector Field

Given the Points:

- Set octree
- Compute vector field
  - Define a function space
  - Splat the samples
- Compute indicator function
- Extract iso-surface

Slides adapted from Kazhdan, Bolitho and Hoppe
Implementation: Indicator Function

Given the Points:

- Set octree
- Compute vector field
- Compute indicator function
  - Compute divergence
  - Solve Poisson equation
- Extract iso-surface

Slides adapted from Kazhdan, Bolitho and Hoppe
Implementation: Indicator Function

Given the Points:

- Set octree
- Compute vector field
- Compute indicator function
  - Compute divergence
  - Solve Poisson equation
- Extract iso-surface
Implementation: Indicator Function

Given the Points:

- Set octree
- Compute vector field
- Compute indicator function
  - Compute divergence
  - Solve Poisson equation
- Extract iso-surface
Implementation: Surface Extraction

Given the Points:

- Set octree
- Compute vector field
- Compute indicator function
- Extract iso-surface