

# Geodesic Convolutional Neural Networks

# Outline

- ▶ Introduction & Motivation
- ▶ Background
  - i. Point Clouds & Meshes
  - ii. Laplace Beltrami Operator & Heat Diffusion
  - iii. Spectral Shape Descriptors (Related Work)
- ▶ Methods
- ▶ Results
- ▶ Demonstration

# Introduction & Motivation

Aim to generalize the idea of a convolutional “filter” to process patches of mesh objects instead of patches of images.

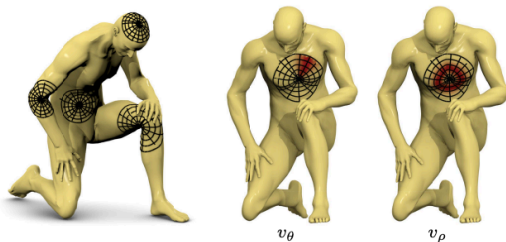


Figure 1: Geodesic patches on a shape

# Background

## Formal Definition of a 3D Shape

1. Connected, smooth compact two-dimensional manifold  $X$
2. Locally, each point  $x$  is homeomorphic to a 2-D Euclidean space (tangent plane,  $T_x X$ )

# Discretization of a 3D Shape from Point Clouds

Given a realized point cloud  $\{x_1, x_2, \dots, x_N\} \in X$ , we can define a **triangular mesh**  $(V, E, F)$

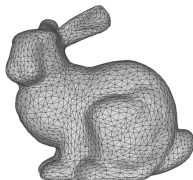
---

Point cloud



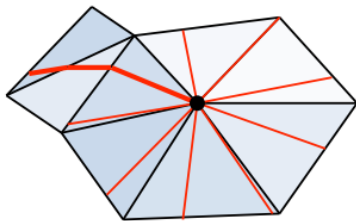
---

Polygon mesh



## Discretization of a 3D Shape from Point Clouds (contd.)

1. Each *interior* edge  $ij \in E$  is only shared by 2 triangular faces  $ikj, jhi \in F$  while *boundary* edges only have 1 associated triangular face
2. Vertices are located at  $\{x_1, x_2, \dots, x_N\}$
3. A function  $f : X \rightarrow \mathbb{R}$  is sampled on  $V$  and can be defined by  $\mathbf{f} = (f(x_1), f(x_2), \dots, f(x_N))^T$ , a  $N$  dimensional vector
4. The set of vertices directly connected to  $i$  is called the *1-ring* of  $i$



# Laplace Beltrami Operator

Generalization of the Laplacian to non-Euclidean space

1. Intrinsic (dependent only on the Riemannian metric)
2. Isometric (invariant to distance preserving deformations of a manifold)
3. Yields an eigen-decomposition with real non-negative eigenvalues  $\lambda_i$ , and an orthonormal basis of eigenfunctions  $\phi_i(x)$ .

## Laplace Beltrami Operator (on a mesh!)

Since we can't work in the function space (we only have points on the manifold), we work on the discretized version defined by

$$L = A^{-1}W$$

where  $L$  is a  $N \times N$  matrix.

- ▶ We can define its eigenvalues and orthonormal basis with the traditional matrix eigen-decomposition of  $L$ .

**The main takeaway is that you can construct a mesh, and define an operator on it.**



# Spectral Shape Descriptors

Most take the form of

$$f(x) = \sum_{k \geq 1} \tau(\lambda_k) \phi_k^2(x) \approx \sum_k^K \tau(\lambda_k) \phi_k^2(x) \quad (1)$$

where  $\tau(\cdot)$  is some transfer function, and  $\lambda_k$  and  $\phi_k(\cdot)$  are the respective eigenvalues and eigenvectors of the LBO.

## 1. Heat Kernel Signature

- a)  $\tau_t(\lambda) = e^{-\lambda t}$
- b) Poor localization

## 2. Wave Kernel Signature

- a)  $\tau_\nu(\lambda) = e^{\frac{\log \nu - \log \lambda}{2\sigma^2}}$
- b) Poor globalization

## 3. Optimal Spectral Descriptors

- a)  $\tau_q(\lambda) = \sum_{m=1}^M a_{qm} \beta_m(\lambda)$
- b) Have to learn the spline parameters

## Distance on a Riemannian Manifold

The length of a differentiable curve  $L(\gamma)$  on a Riemannian manifold with metric  $g$  can be given by

$$L(\gamma) = \int_a^b \sqrt{g_{\gamma(t)}(\dot{\gamma}(t), \dot{\gamma}(t))} dt \quad (2)$$

Consequently, the distance  $d(p, q)$  on a Riemannian manifold is  $L(\gamma^*)$ , where  $\gamma^*$  is the infimum of all differentiable curves which satisfy  $\gamma(a) = p$ ,  $\gamma(b) = q$

Finding the geodesic distance on a mesh can be done numerically using any Boundary Value Problem solver (fast marching algorithm, etc)

# Methods

# Geodesic Convolution

1. Defining a patch operator
2. Defining a convolution

## Defining a Patch Operator

Let  $B_{\rho_0}(x)$  be a geodesic ball of size  $\rho_0$ .

$$\Omega(x) : B_{\rho_0}(x) \rightarrow [0, \rho_0] \times [0, 2\pi]$$

Patch operator interpolates a function  $f$  in local coordinates

$$(Df(x))(\rho, \theta) = (f \circ \Omega^{-1}(x))(\rho, \theta) \quad (3)$$

$$(Df(x))(\rho, \theta) = \int_X v_{\rho, \theta}(x, y) f(y) dy \quad (4)$$

$$v_{\rho, \theta}(x, y) = v_{\rho}(x, y) v_{\theta}(x, y) \quad (5)$$

1.  $v_{\rho}(x, y) \approx$  geodesic distance between  $x, y$
2.  $v_{\theta}(x, y) \approx$  geodesic distance between the point  $y$  and the geodesic generated at  $x$ , in the direction  $\theta$

# Defining a Patch Operator (Discrete's Version)

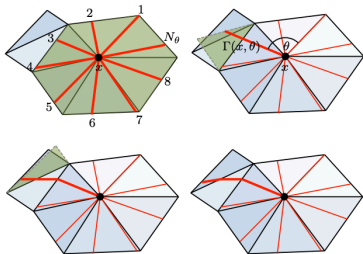


Figure 2: Construction of local geodesic polar coordinates on a triangular mesh. Shown clock-wise: division of 1-ring of vertex  $x_i$  into  $N_\theta$  equi-angular bins; propagation of a ray (bold line) by unfolding the respective triangles (marked in green).

## Defining A Convolution

$$(f \star a)(x) = \sum_{\rho, \theta} a(\theta + \Delta\theta, \rho)(Df(x))(\rho, \theta) \quad (6)$$

where  $a(\cdot, \cdot)$  is a filter.

Effectively, we are projecting  $x$  onto local angular coordinates, and performing a convolution on those coordinates.

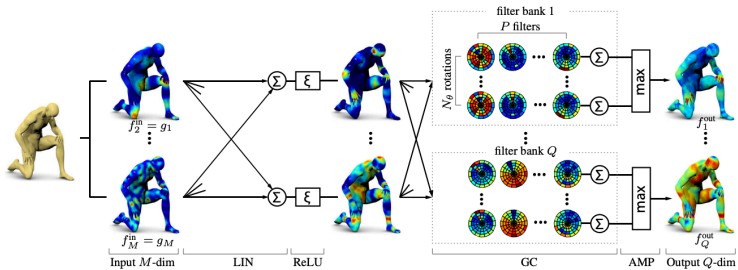
# Convolutional Layers

1. Linear Layer (standard)
2. Geodesic Convolution (GC)
  - ▶  $\sum_{p=1}^P (f_p \star a_{\Delta\theta, qp})(x)$
  - ▶ is computed for all  $N_\theta$  (similar to other GCNN paper)
3. Angular Max Pooling (AMP)
  - ▶  $\max_{\Delta\theta} f_{\Delta\theta, p}^{in}(x)$
  - ▶ follows GC layer
4. Fourier Transform Magnitude
  - ▶  $f_p^{out}(\rho, w) = |\sum_{\theta} e^{-iw\theta} (Df(x))(\rho, \theta)|$
  - ▶ removes rotational ambiguity
5. Covariance (COV)
  - ▶  $f^{out} = \int_X (f^{in}(x) - \mu)(f^{in}(x) - \mu)^T dx$
  - ▶ produces a global descriptor

The spectral shape descriptors can be recovered from some specific parametrization of the above.



# Example Architecture



# Results

Three tasks:

1. Shape Retrieval
  - ▶ discriminate between classes of shapes
2. Shape Correspondence
  - ▶ vertex labeling problem
3. Invariant descriptors
  - ▶ produces a local descriptor of  $x$

# Invariant Descriptors

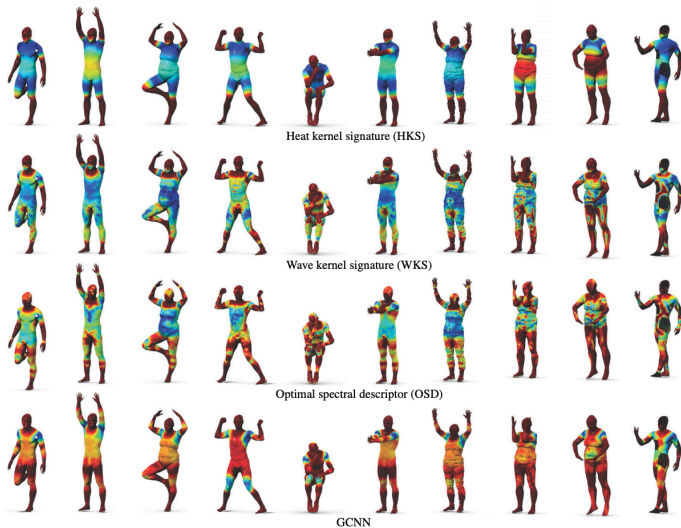


Figure 4: Normalized Euclidean distance between the descriptor at a reference point on the shoulder (white sphere) and the descriptors computed at the rest of the points for different transformations (shown left-to-right: near isometric deformations, non-isometric deformations, topological noise, geometric noise, uniform/non-uniform subsampling, missing parts). Cold and hot colors represent small and large distances, respectively; distances are saturated at the median value. Ideal descriptors would produce a distance map with a sharp minimum at the corresponding point and no spurious local minima at other locations.

# Shape Correspondence

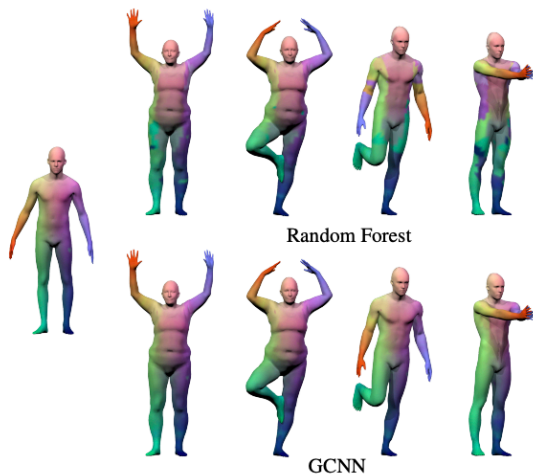


Figure 7: Example of correspondence obtained with GCNN (bottom) and random forest (top). Similar colors encode corresponding points.

# Shape Retrieval

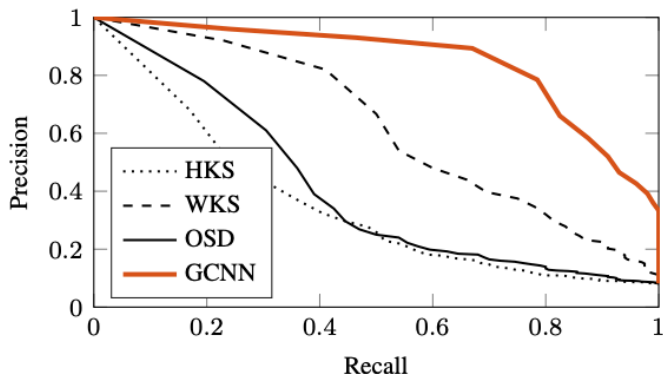


Figure 8: Performance (in terms of Precision-Recall) of shape retrieval on the FAUST dataset using different descriptors. Higher curve corresponds to better performance.

## Code Demo

[https://github.com/jonathanmasci/EG16\\_tutorial/blob/master/  
deep\\_learning\\_for\\_3D\\_shape\\_analysis.ipynb](https://github.com/jonathanmasci/EG16_tutorial/blob/master/deep_learning_for_3D_shape_analysis.ipynb)