Geometric Deep Learning on Graphs and Manifolds

Michael Bronstein\textsuperscript{1,2,3} Joan Bruna\textsuperscript{6} Arthur Szlam\textsuperscript{5}, Xavier Bresson\textsuperscript{4} Yann LeCun\textsuperscript{5,6}

https://qdata.github.io/deep2Read
Presenter: Arshdeep Sekhon

https://qdata.github.io/deep2Read
1. Motivation

2. Basics of Euclidean CNNs

3. Basics
   - Basics: Graph Theory
   - Basics: Riemannian manifolds
   - Using Dirichlet Energy

4. Spectral Domain CNNs

5. GNNs: Spatial View

6. Spatial Domain Geometric Deep Learning

7. Applications
Motivation

- What kind of geometric structure found in images/text/etc exploited by CNNs
- How to use this universal property on non euclidean domains
Examples of non euclidean domains

Manifolds

Graphs
Some Distinctions?

- Domain Structure/Data on a Domain
- Fixed Graph vs Varying Graph
- Known Graph vs Unknown Graph

https://qdata.github.io/deep2Read
Basics of Euclidean CNNs

- Translational Invariance
- Compositionality deformation stability: localization in space,\(^1\)
- constant features \(O(1)\) and \(O(n)\) computation time

---

\(^1\)“each feature extraction in our network is followed by an additional layer which performs a local averaging and a sub-sampling, reducing the resolution of the feature map. This layer introduces a certain level of invariance to distortions and translations.”
Euclidean CNNs

- defined on euclidean domains or on discrete grids
- Grids have the above mentioned properties
- inducitve bias for images
Main Idea

- Extending pooling and conv to non euclidean domains (graphs/manifolds)
- Assume stationarity and compositionality (find appropriate operators for filtering and pooling)
- How to make them fast?
Types of Non-Euclidean CNNs

Two types of non euclidean CNNs

- Spectral Domain
- Spatial Domain
Weighted undirected graph $G$ with vertices $V = \{1, \ldots, n\}$,
edges $E \subset V \times V$
edge weights $w_{ij} \geq 0$ for $(i, j) \in E$
Functions over the vertices $L^2(V) = \{f : V \to R\}$
Vectors in hilbert space: $f = (f_1, \ldots, f_n)$, encoding value of function at every node
Hilbert space with inner product $\langle f, g \rangle_{L^2(V)} = \sum_{i \in V} f_i g_i = f^T g$
Graph Laplacian

- Find geometry of a structure: measure smoothness of a function

The Laplacian measures what you could call the curvature or stress of the field.

Unnormalized Laplacian: $\Delta f_i = \sum_{i,j} w_{ij} (f_i - f_j)$

Represented as a positive semi-definite $n \times n$, $\Delta = D - W$ where $W = (w_{ij})$ and $D = \text{diag} (\sum_{j \neq i} w_{ij})$.
Graph Laplacian

- Find geometry of a structure: measure smoothness of a function
- The Laplacian measures what you could call the curvature or stress of the field.
- Unnormalized Laplacian: \( \Delta f_i = \sum_{i,j} w_{ij} (f_i - f_j) \)
- difference between \( f \) and its local average: \( f_i \sum_{j} w_{ij} - \sum_{j} w_{ij} f_j \)
- Represented as a positive semi-definite \( n \times n \),
- \( \Delta = D - W \) where
- \( W = (w_{ij}) \) and \( D = \text{diag}(\sum_{j \neq i} w_{ij}) \)
Smoothness of function

- Dirichlet Energy: a measure of how much the function $f$ changes over $M \subset \mathbb{R}^N$
  \[ \|f\|_G^2 = \frac{1}{2} \sum_{ij} w_{ij} (f_i - f_j)^2 = f^T \Delta f \] (1)
- measures the smoothness of $f$ (how fast it changes locally)
Riemannian manifolds

- **Manifold** $\mathcal{X} = \text{topological space}

- **Tangent plane** $T_x\mathcal{X} = \text{local Euclidean representation of manifold } \mathcal{X} \text{ around } x$

- **Riemannian metric** describes the local intrinsic structure at $x$

  $$\langle \cdot, \cdot \rangle_{T_x\mathcal{X}} : T_x\mathcal{X} \times T_x\mathcal{X} \to \mathbb{R}$$

- **Scalar fields** $f : \mathcal{X} \to \mathbb{R}$ and **vector fields** $F : \mathcal{X} \to T\mathcal{X}$

- **Hilbert spaces** with inner products

  $$\langle f, g \rangle_{L^2(\mathcal{X})} = \int_{\mathcal{X}} f(x)g(x)dx$$

  $$\langle F, G \rangle_{L^2(T\mathcal{X})} = \int_{\mathcal{X}} \langle F(x), G(x) \rangle_{T_x\mathcal{X}}dx$$
Manifold Laplacian

- **Laplacian** $\Delta : L^2(\mathcal{X}) \to L^2(\mathcal{X})$
  \[ \Delta f(x) = -\text{div} \nabla f(x) \]
  where gradient $\nabla : L^2(\mathcal{X}) \to L^2(T\mathcal{X})$
  and divergence $\text{div} : L^2(T\mathcal{X}) \to L^2(\mathcal{X})$
  are adjoint operators
  \[ \langle F, \nabla f \rangle_{L^2(T\mathcal{X})} = \langle \text{div} F, f \rangle_{L^2(\mathcal{X})} \]

- Laplacian is self-adjoint
  \[ \langle \Delta f, f \rangle_{L^2(\mathcal{X})} = \langle f, \Delta f \rangle_{L^2(\mathcal{X})} \]

- **Continuous limit** of graph
  Laplacian under some conditions

- **Dirichlet energy** of $f$
  \[ \langle \nabla f, \nabla f \rangle_{L^2(T\mathcal{X})} = \int_{\mathcal{X}} f(x) \Delta f(x) dx \]
  measures the smoothness of $f$ (how fast it changes locally)
find class of functions smooth

Find the smoothest orthogonal basis

$$\min_{\phi_1} E_{dir}(\psi_1) \quad s.t. \|\phi_1\| = 1$$ \quad (2)

similarly find subsequent eigen vectors orthogonal to the previos ones in order of smoothness

Can be reoformulated as:

$$\min_{\phi \in \mathbb{R}^{n \times n}} \text{trace}(\phi^T \Delta \phi) \quad s.t. \phi^T \phi = I$$ \quad (3)

laplacian eigen vectors are the solutions to this equation
Laplacian Eigen Vectors

$$\Delta = \phi \Lambda \phi^T$$ (4)

First eigenfunctions of 1D Euclidean Laplacian
Laplacian Eigen Vectors for Graphs and Manifolds

First eigenfunctions of a graph Laplacian

https://qdata.github.io/deep2Read
Fourier Analysis on Euclidean Spaces

- related to the solution of dirichlet

A function $f : [-\pi, \pi] \rightarrow \mathbb{R}$ can be written as a Fourier series

$$f(x) = \sum_{k \geq 0} \langle f, e^{ikx} \rangle_{L^2([-\pi, \pi])} e^{ikx}$$

$k \geq 0$ Fourier coefficient

$$\hat{f}_k$$

$$= \hat{f}_1 + \hat{f}_2 + \hat{f}_3 + \ldots$$

Fourier basis = Laplacian eigenfunctions: $-\frac{d^2}{dx^2}e^{ikx} = k^2 e^{ikx}$
Fourier Analysis on graphs

A function $f : \mathcal{V} \to \mathbb{R}$ can be written as Fourier series

$$f = \sum_{k=1}^{n} \langle f, \phi_k \rangle_{L^2(\mathcal{V})} \hat{f}_k \phi_k$$

Fourier basis = Laplacian eigenfunctions: $\Delta \phi_k = \lambda_k \phi_k$

$\lambda_k = \text{frequency}$

First Fourier basis elements of a manifold.
Given two functions \( f, g : [-\pi, \pi] \rightarrow \mathbb{R} \) their convolution is a function

\[
(f \ast g)(x) = \int_{-\pi}^{\pi} f(x')g(x - x')dx'
\]

- **Shift-invariance:** \( f(x - x_0) \ast g(x) = (f \ast g)(x - x_0) \)

- **Convolution theorem:** Fourier transform diagonalizes the convolution operator \( \Rightarrow \) convolution can be computed in the Fourier domain as

\[
\widehat{(f \ast g)} = \hat{f} \cdot \hat{g}
\]
Convolution of two vectors $f = (f_1, \ldots, f_n)^T$ and $g = (g_1, \ldots, g_n)^T$

$$f \ast g = \begin{bmatrix}
  g_1 & g_2 & \ldots & \ldots & g_n \\
  g_n & g_1 & g_2 & \ldots & g_{n-1} \\
  \vdots & \vdots & \ddots & \ddots & \vdots \\
  g_3 & g_4 & \ldots & g_1 & g_2 \\
  g_2 & g_3 & \ldots & \ldots & g_1 \\
\end{bmatrix}
\begin{bmatrix}
  f_1 \\
  \vdots \\
  f_n \\
\end{bmatrix}$$

$$= \Phi \begin{bmatrix}
  \hat{g}_1 \\
  \vdots \\
  \hat{g}_n \\
\end{bmatrix} = \Phi^T f$$
Convolution theorem in graphs

\[ f * g = \begin{bmatrix} g_1 & g_2 & \cdots & g_n \\
 g_n & g_1 & g_2 & \cdots & g_{n-1} \\
 \vdots & \vdots & \ddots & \vdots & \vdots \\
 g_3 & g_4 & \cdots & g_1 & g_2 \\
 g_2 & g_3 & \cdots & \cdots & g_1 \end{bmatrix} \begin{bmatrix} f_1 \\
 f_2 \\
 \vdots \\
 f_{n-1} \\
 f_n \end{bmatrix} \]

\[ = \Phi \begin{bmatrix} \hat{f}_1 \hat{g}_1 \\
 \vdots \\
 \hat{f}_n \hat{g}_n \end{bmatrix} \]
Spectral Convolution

defined by analogy:

\[ f \ast g = \sum_{k \geq 1} \langle f, \phi_k \rangle_{L^2(\mathcal{V})} \langle g, \phi_k \rangle_{L^2(\mathcal{V})} \phi_k \]

product in the Fourier domain

inverse Fourier transform

\[ f \ast g = \Phi \text{diag}(\hat{g}_1, \ldots, \hat{g}_n) \Phi^\top f \]
Issues with Spectral Graph CNN

- Not shift-invariant! (G has no circulant structure)
- Filter coefficients depend on basis $\phi_1, ..., \phi_n$
Spectral CNN

- Convolution expressed in the spectral domain $g = \phi W \phi^T f$
- $W$ is $n \times n$ diagonal matrix of learnable spectral filter coefficients
Filters are basis-dependent: does not generalize across graphs

$O(n)$ parameters per layer

$O(n^2)$ computation of forward and inverse Fourier transforms

No guarantee of spatial localization of filters: free to choose multiplier
Localization and Smoothness

**Vanishing moments:** In the Euclidean setting

\[
\int_{-\infty}^{+\infty} |x|^{2k} |f(x)|^2 dx = \int_{-\infty}^{+\infty} \left| \frac{\partial^k \hat{f}(\omega)}{\partial \omega^k} \right|^2 d\omega
\]

**Localization in space = smoothness in frequency domain**

Parametrize the filter using a smooth spectral transfer function $\tau(\lambda)$

Application of the parametric filter with learnable parameters $\alpha$

\[
\tau_\alpha(\Delta)f = \Phi \begin{pmatrix} \tau_\alpha(\lambda_1) \\ \vdots \\ \tau_\alpha(\lambda_n) \end{pmatrix} \Phi^\top f
\]
Examples

Non-smooth spectral filter (delocalized in space)
Graph Pooling

- Produce a sequence of coarsened graphs
- Max or average pooling of collapsed vertices
- Binary tree arrangement of node indices
Limitations

- Poor generalization across non-isometric domains unless kernels are localized
- Spectral kernels are isotropic due to rotation invariance of the Laplacian
- Only undirected graphs, as symmetry of the Laplacian matrix is assumed
Spatial GNNs

- Given a function $h^0: V \rightarrow \mathbb{R}^{d_0}$ (where $V$ is the vertices of the graph), set

$$h_j^{(i+1)} = f^i(h_j^{(i)}, c_j^{(i)})$$

$$c_j^{(i+1)} = \sum_{j' \in N(j)} W_{jj'} h_{j'}^{(i+1)}.$$
Spatial and Spectral link

- pick a number $r$

$$h^{(i+1)} = f^{(i)}(W^0 h^{(i)}, W^1 h^{(i)}, ..., W^r h^{(i)})$$

- higher the power of $r$, richer the filter class
- but tradeoff between test time and power of filters
- Edge decoration
- Vertex decoration
- Interaction Nets
What does GNN look like on a euclidean grid

- Graph is a regular lattice
- gives isotropic filters
- less expressive than a conventional ConvNet
  - no notion of up and down
  - conv nets have implicit ordering implies edge knowledge
- For example, local correlation among pixels /translation, easy to reorder shuffled patches of images
Geodesic Polar Coordinates

Patch operators

Image

Manifold
Convolution on Manifolds

- Geodesic polar coordinates
  \[ u(x, y) = (\rho(x, y), \theta(x, y)) \]

- Set of weighting functions
  \[ w_1(u), \ldots, w_J(u) \]

Spatial convolution

\[
(f * g)(x) = \sum_{j=1}^{J} g_j \int_{X} w_j(u(x, x')) f(x') dx'
\]

where \( g_1, \ldots, g_J \) are the spatial filter coefficients:
Convolution on Manifolds

- Geodesic polar coordinates
  
  \[ u(x, y) = (\rho(x, y), \theta(x, y)) \]

- Gaussian weighting functions
  
  \[ w_{\mu, \Sigma}(u) = \exp\left(-\frac{1}{2}(u - \mu)^T \Sigma^{-1}(u - \mu)\right) \]

  with learnable covariance \( \Sigma \) and mean \( \mu \)

Spatial convolution

\[
(f \ast g)(x) = \sum_{j=1}^{J} g_j \int_{x} w_{\mu_j, \Sigma_j}(u(x, x')) f(x') dx' \]

patch operator \( D_j(x)f \)

where \( g_1, \ldots, g_J \) are the spatial filter coefficients and \( \mu_1, \ldots, \mu_J \) and \( \Sigma_1, \ldots, \Sigma_J \) are patch operator parameters
Correspondence I: Local Feature Learning

**Siamese net**

two net instances with shared parameters Θ
Correspondence II: Labelling

- Ground truth correspondence $\pi : X \rightarrow Y$ from query shape $X$ to some reference shape $Y$ (discretized with $n$ vertices)
- Correspondence $= \text{label each query vertex } x \text{ as reference vertex } y$
- Net output at $x$ after softmax layer $= \text{probability distribution on } Y$
Correspondence Results

Correspondence evaluated using asymmetric Princeton benchmark
(training and testing: disjoint subsets of FAUST)
Matrix Completion

\[
\min_{\mathbf{X} \in \mathbb{R}^{m \times n}} \quad \|\mathbf{X}\|_* \quad \text{s.t.} \quad x_{ij} = a_{ij} \quad \forall i,j \in \Omega
\]

\[
\min_{\mathbf{X} \in \mathbb{R}^{m \times n}} \quad \|\mathbf{X}\|_* + \mu \|\Omega \circ (\mathbf{X} - \mathbf{A})\|_F^2
\]

\[
\min_{\mathbf{X} \in \mathbb{R}^{m \times n}} \quad \mu \|\Omega \circ (\mathbf{X} - \mathbf{A})\|_F^2 + \mu_c \operatorname{tr}(\mathbf{X} \Delta_c \mathbf{X}^T)
\]

\[
\min_{\mathbf{X} \in \mathbb{R}^{m \times n}} \quad \mu \|\Omega \circ (\mathbf{X} - \mathbf{A})\|_F^2 + \mu_c \operatorname{tr}(\mathbf{X} \Delta_c \mathbf{X}^T) + \mu_r \operatorname{tr}(\mathbf{X}^T \Delta_r \mathbf{X})
\]
Spectral vs Spatial Convolution on Non Euclidean Domains: Graphs and Manifolds

- Spectral Better if Graph assumed to be similar across samples
- Leveraging low dimension structure at tangent planes in manifolds for spectral convolution
- Applications