Geometric Deep Learning on Graphs and Manifolds

Michael Bronstein[1,2,3] Joan Bruna[6] Arthur Szlam[5], Xavier Bresson[4] Yann LeCun[5,6]

https://qdata.github.io/deep2Read

Presenter: Arshdeep Sekhon

https://qdata.github.io/deep2Read

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

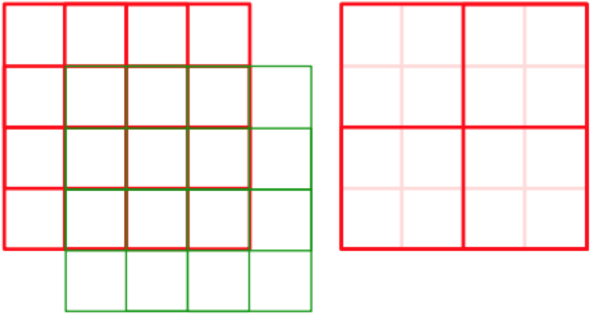
Basics of Euclidean CNNs

- 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

Graph Theory

- Weighted undirected graph G with vertices $V = \{1, ..., n\}$,
- edges $E \subset V \times V$
- edge weights $w_{ij} \geq 0$ for $(i, j) \in E$
- Functions over the vertices $L^2(V) = \{f : V \rightarrow R\}$
- Vectors in hilbert space: $f = (f1, \ldots, fn)$, encoding value of function at every node
- Hilbert space with inner product $< f, g >_{L^2(V)} = \Sigma_{i \in V} f_i g_i = f^T g$

Graph Laplacian

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

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: $\Delta f_i = \Sigma_{i,j} w_{ij}(f_i - f_j)$
- difference between f and its local average: $f_i \Sigma_{ij} w_{ij} - \Sigma_{ij} w_{ij} f_j$
- Represented as a positive semi-definite $n \times n$,
- $\Delta = D - W$ where
- $W = (w_{ij})$ and $D = diag(\Sigma_{j \neq i} wij)$

Smoothness of function

- Dirichlet Energy: a measure of how much the function f changes over $M \subset R^N$

$$||f||_G^2 = \frac{1}{2}\Sigma_{ij} w_{ij}(f_i - f_j)^2 = \boldsymbol{f^T \Delta f} \quad (1)$$

- measures the smoothness of f (how fast it changes locally)

Riemannian manifolds

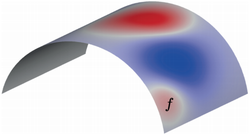
- Manifold $\mathcal{X}$ = topological space
- Tangent plane $T_x\mathcal{X}$ = local Euclidean representation of manifold $\mathcal{X}$ around x
- Riemannian metric describes the local intrinsic structure at x

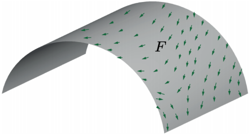
$$\langle \cdot, \cdot \rangle_{T_x\mathcal{X}} : T_x\mathcal{X} \times T_x\mathcal{X} \to \mathbb{R}$$

- Scalar fields $f : \mathcal{X} \to \mathbb{R}$ and vector fields $F : \mathcal{X} \to T\mathcal{X}$
- Hilbert spaces with inner products

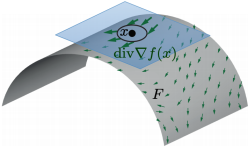
$$\langle f, g \rangle_{L^2(\mathcal{X})} = \int_{\mathcal{X}} f(x)g(x)dx$$

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





Manifold Laplacian



- Laplacian $\Delta : L^2(\mathcal{X}) \to L^2(\mathcal{X})$

  $$\Delta f(x) = -\mathrm{div}\,\nabla f(x)$$

  where gradient $\nabla : L^2(\mathcal{X}) \to L^2(T\mathcal{X})$
  and divergence $\mathrm{div} : L^2(T\mathcal{X}) \to L^2(\mathcal{X})$
  are adjoint operators

  $$\langle F, \nabla f \rangle_{L^2(T\mathcal{X})} = \langle -\mathrm{div} F, f \rangle_{L^2(\mathcal{X})}$$

- 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)

Orthogonal bases on graphs

- find class of functions smooth
- Find the smoothest orthogonal basis

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

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

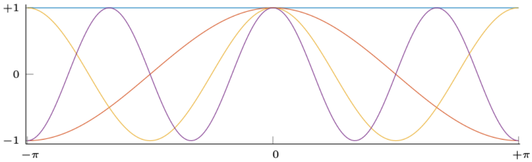
Can be reoformulated as:

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

laplacian eigen vectors are the solutions to this equation

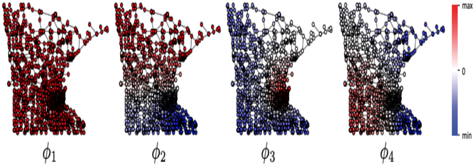
Laplacian Eigen Vectors

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

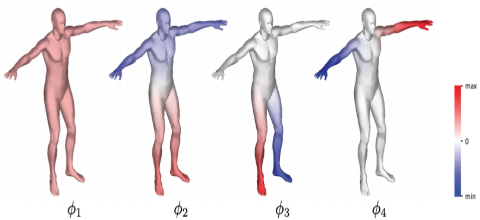


First eigenfunctions of 1D Euclidean Laplacian

Laplacian Eigen Vectors for Graphs and Manifolds



First eigenfunctions of a graph Laplacian



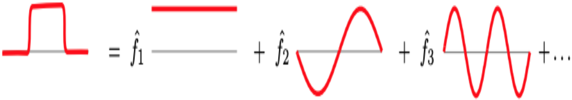
Geometric Dee

Fourier Analysis on Euclidean Spaces

- related to the solution of dirichlet

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

$$f(x) = \sum_{k \geq 0} \underbrace{\langle f, e^{ikx} \rangle_{L^2([-\pi,\pi])}}_{\hat{f}_k \text{ Fourier coefficient}} e^{ikx}$$



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

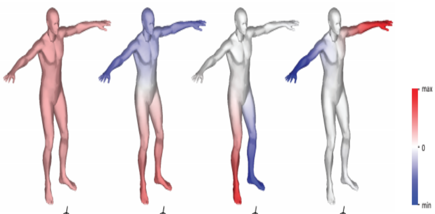
Fourier Analysis on graphs

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

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

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

λ_k = frequency



ϕ_1 ϕ_2 ϕ_3 ϕ_4

First Fourier basis elements of a manifold.

Euclidean Conv Basics

Given two functions $f, g : [-\pi, \pi] \to \mathbb{R}$ their convolution is a function

$$(f \star g)(x) = \int_{-\pi}^{\pi} f(x')g(x - x')dx'$$

- Shift-invariance: $f(x - x_0) \star g(x) = (f \star g)(x - x_0)$
- Convolution theorem: Fourier transform diagonalizes the convolution operator ⇒ convolution can be computed in the Fourier domain as

$$\widehat{(f \star g)} = \hat{f} \cdot \hat{g}$$

Convolution theorem in graphs

Convolution of two vectors $\mathbf{f} = (f_1, \ldots, f_n)^\top$ and $\mathbf{g} = (g_1, \ldots, g_n)^\top$

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

$$= \boldsymbol{\Phi} \begin{bmatrix} \hat{g}_1 & & \\ & \ddots & \\ & & \hat{g}_n \end{bmatrix} \boldsymbol{\Phi}^\top \mathbf{f}$$

Convolution theorem in graphs

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

$$= \mathbf{\Phi} \begin{bmatrix} \hat{f}_1 \cdot \hat{g}_1 \\ \vdots \\ \hat{f}_n \cdot \hat{g}_n \end{bmatrix}$$

Spectral Convolution

defined by analogy:

$$\mathbf{f} \star \mathbf{g} = \underbrace{\sum_{k \geq 1} \underbrace{\langle \mathbf{f}, \phi_k \rangle_{L^2(\mathcal{V})} \langle \mathbf{g}, \phi_k \rangle_{L^2(\mathcal{V})}}_{\text{product in the Fourier domain}} \phi_k}_{\text{inverse Fourier transform}}$$

$$\mathbf{f} \star \mathbf{g} = \Phi \operatorname{diag}(\hat{g}_1, \ldots, \hat{g}_n) \Phi^\top \mathbf{f}$$

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

Issues

- 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

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

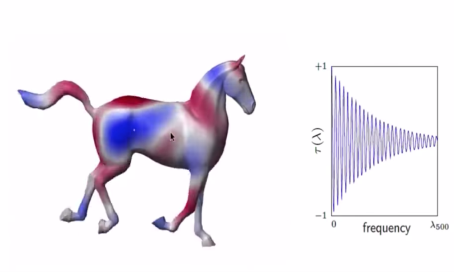
Localization in space = smoothness in frequency domain

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

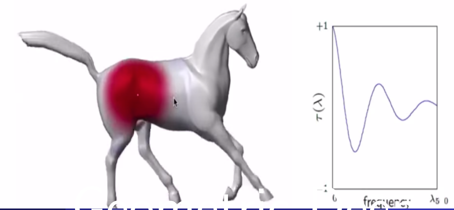
Application of the parametric filter with learnable parameters α

$$\tau_\alpha(\mathbf{\Delta})\mathbf{f} = \mathbf{\Phi} \begin{pmatrix} \tau_\alpha(\lambda_1) & & \\ & \ddots & \\ & & \tau_\alpha(\lambda_n) \end{pmatrix} \mathbf{\Phi}^\top \mathbf{f}$$

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 $\mathbf{h}^0 : \mathcal{V} \to \mathbb{R}^{d_0}$ (where $\mathcal{V}$ is the vertices of the graph), set

$$\begin{aligned} \mathbf{h}_j^{(i+1)} &= f^i(\mathbf{h}_j^{(i)}, \mathbf{c}_j^{(i)}) \\ \mathbf{c}_j^{(i+1)} &= \sum_{j' \in N(j)} \mathbf{W}_{jj'} \mathbf{h}_{j'}^{(i+1)}. \end{aligned}$$

Spatial and Spectral link

- pick a number r

$$\mathbf{h}^{(i+1)} = f^{(i)}\left(\mathbf{W}^0\mathbf{h}^{(i)}, \mathbf{W}^1\mathbf{h}^{(i)}, \ldots, \mathbf{W}^r\mathbf{h}^{(i)}\right)$$

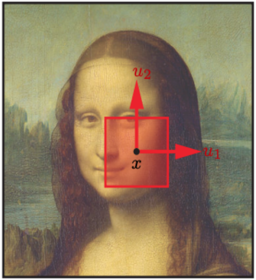
- higher the power of r, richer the filter class
- but tradeoff between test time and power of filters
- Edge decoration
- Vertex decoration
- Interaction Nets

What does GNN look like on a euclidean grid

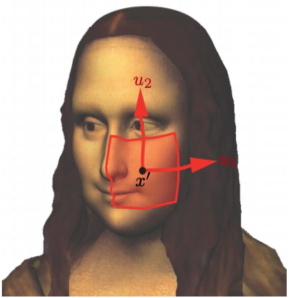
- Graph is a regular lattice
- gives isotropic filters
- less expressive than a conventional ConvNet
  - no notion of up and down
  - conv nets have implicit ordering implies edge knowledge
- For example, local correlation among pixels /translation, easy to reorder shuffled patches of iimages

Geodesic Polar Coordiantes

Patch operators



Image



Manifold

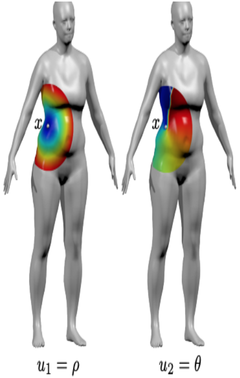
Convolution on Manifolds

- Geodesic polar coordinates

$$\mathbf{u}(x, y) = (\rho(x, y), \theta(x, y))$$

- Set of weighting functions

$$w_1(\mathbf{u}), \ldots, w_J(\mathbf{u})$$



Spatial convolution

$$(f \star g)(x) = \sum_{j=1}^{J} g_j \underbrace{\int_{\mathcal{X}} w_j(\mathbf{u}(x, x')) f(x') dx'}_{\text{patch operator } \mathcal{D}_j(x) f}$$

where $g_1, \ldots, g_J$ are the spatial filter coefficients

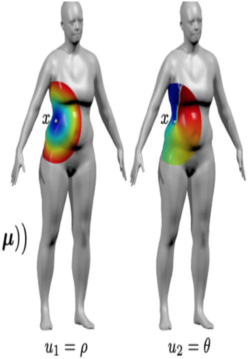
- Geodesic polar coordinates

$$\mathbf{u}(x,y) = (\rho(x,y), \theta(x,y))$$

- Gaussian weighting functions

$$w_{\boldsymbol{\mu},\boldsymbol{\Sigma}}(\mathbf{u}) = \exp\left(-\tfrac{1}{2}(\mathbf{u}-\boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1}(\mathbf{u}-\boldsymbol{\mu})\right)$$

with learnable covariance $\boldsymbol{\Sigma}$ and mean $\boldsymbol{\mu}$

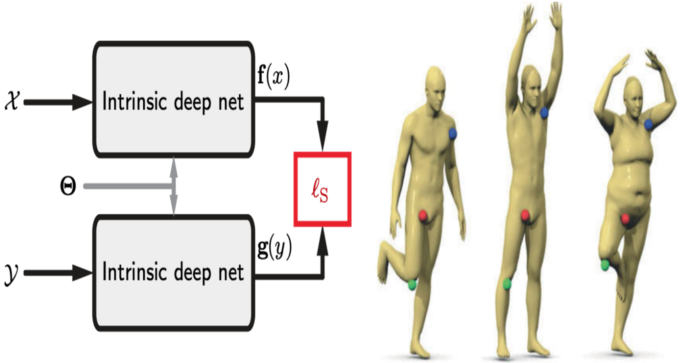


Spatial convolution

$$(f \star g)(x) = \sum_{j=1}^{J} g_j \underbrace{\int_{\mathcal{X}} w_{\boldsymbol{\mu}_j,\boldsymbol{\Sigma}_j}(\mathbf{u}(x,x')) f(x') dx'}_{\text{patch operator } \mathcal{D}_j(x) f}$$

where $g_1, \ldots, g_J$ are the spatial filter coefficients and $\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_J$ and $\boldsymbol{\Sigma}_1, \ldots, \boldsymbol{\Sigma}_J$ are patch operator parameters

Correspondence I: Local Feature Learning

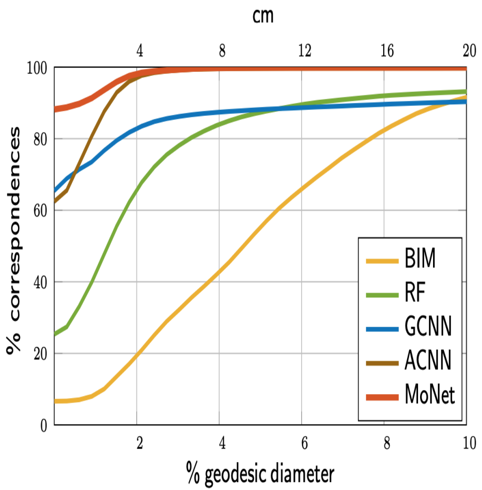


Siamese net two net instances with shared parameters Θ

Correspondence II: Labelling

- Groundtruth correspondence$\pi : X \rightarrow Y$ from query shape X to some reference shape Y (discretized with n vertices)
- Correspondence = label each query vertex x as reference vertex y
- Net output at x after softmax layer$= probabilitydistributiononY$

Correspondence Results



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

Matrix Completion

$$\min_{\mathbf{X}\in\mathbb{R}^{m\times n}} \quad \|\mathbf{X}\|_* \quad \text{s.t.} \quad x_{ij} = a_{ij} \;\; \forall ij \in \Omega$$

$$\min_{\mathbf{X}\in\mathbb{R}^{m\times n}} \quad \|\mathbf{X}\|_* + \mu\|\boldsymbol{\Omega} \circ (\mathbf{X} - \mathbf{A})\|_{\mathrm{F}}^2$$

$$\min_{\mathbf{X}\in\mathbb{R}^{m\times n}} \quad \mu\|\boldsymbol{\Omega} \circ (\mathbf{X} - \mathbf{A})\|_{\mathrm{F}}^2 + \mu_{\mathrm{c}}\,\mathrm{tr}(\mathbf{X}\boldsymbol{\Delta}_{\mathrm{c}}\mathbf{X}^\top)$$

$$\min_{\mathbf{X}\in\mathbb{R}^{m\times n}} \quad \mu\|\boldsymbol{\Omega} \circ (\mathbf{X} - \mathbf{A})\|_{\mathrm{F}}^2 + \mu_{\mathrm{c}}\,\underbrace{\mathrm{tr}(\mathbf{X}\boldsymbol{\Delta}_{\mathrm{c}}\mathbf{X}^\top)}_{\|\mathbf{X}\|_{\mathcal{G}_{\mathrm{c}}}^2} + \mu_{\mathrm{r}}\,\underbrace{\mathrm{tr}(\mathbf{X}^\top\boldsymbol{\Delta}_{\mathrm{r}}\mathbf{X})}_{\|\mathbf{X}\|_{\mathcal{G}_{\mathrm{r}}}^2}$$

Kalofolias, Bresson, Bronstein, Vandergheynst 2014

Summary

- 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