Learning class-specific descriptors for deformable shapes using localized spectral convolutional networks

Davide Boscaini*

University of Lugano
Switzerland

SGP 2015
Graz, Austria
July 6, 2015

* In collaboration with:
Jonathan Masci, Simone Melzi, Michael Bronstein, Umberto Castellani, Pierre Vandergheynst
Taking inspiration from (successful) research in Image Analysis

1. SIFT \(^1\)
2. MSER \(^2\)
3. Shape Context \(^3\)

\(^1\) Lowe 2004; \(^2\) Matas et al. 2002; \(^3\) Belongie et al. 2000;
Taking inspiration from (successful) research in Image Analysis

1 Lowe 2004; 2 Matas et al. 2002; 3 Belongie et al. 2000;
Taking inspiration from (successful) research in Image Analysis
Taking inspiration from (successful) research in Image Analysis

Zeiler, Fergus 2013
Taking inspiration from (successful) research in Image Analysis

input  convolution  aggregation  convolution  aggregation  output

...and for shapes?
Taking inspiration from (successful) research in Image Analysis

... and for shapes?

Challenges:

- shift-invariant sliding window
- localized convolution
- dependence from the data structure (mesh, point clouds, ... )
• Review of shape descriptors
• Extending convolution to shapes
• Proposed architecture
• Results
• **Review of shape descriptors**
  • Extending convolution to shapes
  • Proposed architecture
  • Results
A signal processing view of spectral descriptors

\[ f(x) = \sum_{k \geq 1} \begin{pmatrix} 
\tau_1(\lambda_k) \\ \vdots \\ \tau_Q(\lambda_k) 
\end{pmatrix} \phi_k^2(x) \]
A signal processing view of spectral descriptors

\[ f(x) = \sum_{k \geq 1} \left( \begin{array}{c}
\tau_1(\lambda_k) \\
\vdots \\
\tau_Q(\lambda_k)
\end{array} \right) \phi_k^2(x) \]

\[ \tau_i(\lambda) = \exp(-t_i\lambda) \]

Collection of low-pass transfer functions

Sun, Ovsjanikov, Guibas 2009
A signal processing view of spectral descriptors

\[ f(x) = \sum_{k \geq 1} \left( \begin{array}{c} \tau_1(\lambda_k) \\ \vdots \\ \tau_Q(\lambda_k) \end{array} \right) \phi_k^2(x) \]

\[ \tau_i(\lambda) = \exp(-t_i\lambda) \]

Collection of low-pass transfer functions

\[ h_{t_1}(x, x) \quad h_{t_2}(x, x) \quad h_{t_Q}(x, x) \]
A signal processing view of spectral descriptors

\[ f(x) = \sum_{k \geq 1} \begin{pmatrix} \tau_1(\lambda_k) \\ \vdots \\ \tau_Q(\lambda_k) \end{pmatrix} \phi_k^2(x) \]

\[ \tau_i(\lambda) = \exp(-t_i\lambda) \]

\[ \tau_i(\lambda) = \exp\left(-\frac{(\log e - \log \lambda)^2}{2\sigma^2}\right) \]

Collection of **low-pass**

transfer functions

\[ h_{t1}(x, x) \quad h_{t2}(x, x) \quad h_{tQ}(x, x) \]

Collection of **band-pass**

transfer functions

Sun, Ovsjanikov, Guibas 2009; Aubry, Schlickewei, Cremers 2011
A signal processing view of spectral descriptors

\[ f(x) = \sum_{k \geq 1} \left( \begin{array}{c} \tau_1(\lambda_k) \\ \vdots \\ \tau_Q(\lambda_k) \end{array} \right) \phi_k^2(x) \]

\[ \tau_i(\lambda) = \exp(-t_i \lambda) \]

\[ \tau_i(\lambda) = \exp\left( -\frac{(\log e - \log \lambda)^2}{2\sigma^2} \right) \]

Collection of **low-pass**

transfer functions

\[ h_{t1}(x, x) \]
\[ h_{t2}(x, x) \]
\[ h_{tQ}(x, x) \]

Collection of **band-pass**

transfer functions

\[ p_{e1}(x) \]
\[ p_{e2}(x) \]
\[ p_{eQ}(x) \]

Sun, Ovsjanikov, Guibas 2009; Aubry, Schlickewei, Cremers 2011
A generic $Q$-dimensional spectral descriptor of the form

$$f_\tau(x) = \sum_{k \geq 1} \begin{pmatrix} \tau_1(\lambda_k) \\ \vdots \\ \tau_Q(\lambda_k) \end{pmatrix} \phi_k^2(x)$$

can be parametrized by frequency responses $\tau(\lambda) = (\tau_1(\lambda), \ldots, \tau_Q(\lambda))^\top$
Optimal spectral descriptor

A generic $Q$-dimensional spectral descriptor of the form

$$f_A(x) = \sum_{k \geq 1} A \begin{pmatrix} \beta_1(\lambda_k) \\ \vdots \\ \beta_M(\lambda_k) \end{pmatrix} \phi_k^2(x)$$

can be parametrized by frequency responses $\tau(\lambda) = (\tau_1(\lambda), \ldots, \tau_Q(\lambda))^\top$ represented in some fixed basis $\beta_1(\lambda), \ldots, \beta_M(\lambda)$ by an $Q \times M$ matrix $A$.
Optimal spectral descriptor

A generic $Q$-dimensional spectral descriptor of the form

$$ f_A(x) = A \sum_{k \geq 1} \begin{pmatrix} \beta_1(\lambda_k) \\ \vdots \\ \beta_M(\lambda_k) \end{pmatrix} \phi_k^2(x) $$

parametrized by linear combination coefficients $A$ of geometry vectors

$$ g(x) = (g_1(x), \ldots, g_M(x))^\top $$

Litman, Bronstein 2014
A generic \( Q \)-dimensional spectral descriptor of the form

\[
f_A(x) = A \sum_{k \geq 1} \begin{pmatrix} \beta_1(\lambda_k) \\ \vdots \\ \beta_M(\lambda_k) \end{pmatrix} \phi_k^2(x)
\]

parametrized by linear combination coefficients \( A \) of geometry vectors \( g(x) = (g_1(x), \ldots, g_M(x))^\top \)

\( \Rightarrow \) Hard to model axiomatically... yet easy to learn from examples!

Litman, Bronstein 2014
Optimal spectral descriptor

Given a set of points $x$, knowingly similar points (positive) $x_+$, and knowingly dissimilar points (negative) $x_-$. 

Litman, Bronstein 2014
Optimal spectral descriptor

Given a set of points $x$, knowingly similar points (positive) $x_+$, and knowingly dissimilar points (negative) $x_-$

Find optimal $A$ by minimizing the loss

$$
\min_{A: A^\top A = I} \gamma \mathbb{E}(\|f_A(x) - f_A(x_+)\|^2) - (1 - \gamma) \mathbb{E}(\|f_A(x) - f_A(x_-)\|^2)
$$

Litman, Bronstein 2014
Optimal spectral descriptor

Optimal transfer functions learned from positive and negative examples

\begin{align*}
  f^1_A(x) & \\
  f^2_A(x) & \\
  f^Q_A(x) & 
\end{align*}

Litman, Bronstein 2014
The need for context
Intrinsic shape context

Geodesic patches

Kokkinos, Bronstein, Bronstein, Litman 2012
Intrinsic shape context

Geodesic patches

Angular weight

Radial weight

Kokkinos, Bronstein, Bronstein, Litman 2012
Intrinsic shape context

Geodesic patches

ISC(x)

Angular weight

Radial weight

Kokkinos, Bronstein, Bronstein, Litman 2012
• Review of shape descriptors
• **Extending convolution to shapes**
• Proposed architecture
• Results
A function $f : [-\pi, \pi] \to \mathbb{R}$ can be written as **Fourier series**

$$f(x) = \sum_{w} \frac{1}{2\pi} \int_{-\pi}^{\pi} f(\xi) e^{-iwx} d\xi e^{iwx}$$
A function $f : [−\pi, \pi] \to \mathbb{R}$ can be written as **Fourier series**

$$
    f(x) = \sum_{w} \frac{1}{2\pi} \int_{-\pi}^{\pi} f(\xi) e^{-iwx} d\xi e^{iwx}
$$

$$
    \mathcal{F}(f)(w) = \langle f, e^{-iwx} \rangle_{L^2([−\pi, \pi])}
$$
A function $f : [-\pi, \pi] \to \mathbb{R}$ can be written as **Fourier series**

$$f(x) = \sum_w \frac{1}{2\pi} \int_{-\pi}^{\pi} f(\xi) e^{-iwx} d\xi e^{iwx}$$

$$\mathcal{F}(f)(w) = \langle f, e^{-iwx} \rangle_{L^2([-\pi, \pi])}$$

$$\begin{align*}
\alpha_1 \quad + \quad \alpha_2 \quad + \quad \alpha_3 \quad + \ldots
\end{align*}$$
A function \( f : [-\pi, \pi] \rightarrow \mathbb{R} \) can be written as **Fourier series**

\[
f(x) = \sum_w \frac{1}{2\pi} \int_{-\pi}^{\pi} f(\xi) e^{-iw\xi} d\xi e^{iwx}
\]

\[
\mathcal{F}(f)(w) = \langle f, e^{-iwx} \rangle_{L^2([-\pi, \pi])}
\]

Fourier basis = **Laplacian eigenfunctions**

\[
\Delta e^{-iwx} = w^2 e^{-iwx}
\]
A function $f : X \to \mathbb{R}$ can be written as **Fourier series**

$$f(x) = \sum_{k \geq 1} \int_X f(\xi) \phi_k(\xi) d\xi \phi_k(x)$$

$$\mathcal{F}(f)_k = \langle f, \phi_k \rangle_{L^2(X)}$$
Fourier analysis (non-Euclidean spaces)

A function \( f : X \rightarrow \mathbb{R} \) can be written as **Fourier series**

\[
  f(x) = \sum_{k \geq 1} \int_X f(\xi) \phi_k(\xi) d\xi \phi_k(x)
\]

\[
  \mathcal{F}(f)_k = \langle f, \phi_k \rangle_{L^2(X)}
\]

\[
  = \alpha_1 + \alpha_2 + \alpha_3 + \ldots
\]
A function $f: X \to \mathbb{R}$ can be written as **Fourier series**

$$f(x) = \sum_{k \geq 1} \int_X f(\xi)\phi_k(\xi)d\xi \phi_k(x)$$

$$\mathcal{F}(f)_k = \langle f, \phi_k \rangle_{L^2(X)}$$

**Fourier basis** = **Laplacian eigenfunctions**

$$\Delta_X \phi_k(x) = \lambda_k \phi_k(x)$$
Euclidean case: given $f, g : [-\pi, \pi] \to \mathbb{R}$, the Fourier transform diagonalizes the convolution operator

$$\mathcal{F}((f \star g)(x)) = \mathcal{F}(f(x)) \cdot \mathcal{F}(g(x))$$
Euclidean case: given \( f, g : [-\pi, \pi] \rightarrow \mathbb{R} \), the Fourier transform diagonalizes the convolution operator

\[
(f \ast g)(x) = \mathcal{F}^{-1}(\mathcal{F}(f(x)) \cdot \mathcal{F}(g(x)))
\]
Convolution

Euclidean case: given $f, g : [-\pi, \pi] \rightarrow \mathbb{R}$, the Fourier transform diagonalizes the convolution operator

$$(f \ast g)(x) = \mathcal{F}^{-1}(\mathcal{F}(f(x)) \cdot \mathcal{F}(g(x)))$$

Non-Euclidean case: given $f, g : L^2(X) \rightarrow \mathbb{R}$, the generalized convolution can be defined by analogy as

$$(f \ast g)(x) = \sum_{k \geq 1} \langle f, \phi_k \rangle_{L^2(X)} \langle g, \phi_k \rangle_{L^2(X)} \phi_k(x)$$
Euclidean case: given \( f, g : [-\pi, \pi] \rightarrow \mathbb{R} \), the Fourier transform diagonalizes the convolution operator

\[
(f \ast g)(x) = \mathcal{F}^{-1}(\mathcal{F}(f(x)) \cdot \mathcal{F}(g(x)))
\]

Non-Euclidean case: given \( f, g : L^2(X) \rightarrow \mathbb{R} \), the generalized convolution can be defined by analogy as

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

product in the Fourier domain
Euclidean case: given \( f, g : [-\pi, \pi] \to \mathbb{R} \), the Fourier transform diagonalizes the convolution operator

\[
(f \ast g)(x) = \mathcal{F}^{-1} \left( \mathcal{F}(f(x)) \cdot \mathcal{F}(g(x)) \right)
\]

Non-Euclidean case: given \( f, g : L^2(X) \to \mathbb{R} \), the **generalized convolution** can be defined by analogy as

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

where

- \( \mathcal{F}^{-1} \) is the inverse Fourier transform.
- The product \( \langle f, \phi_k \rangle_{L^2(X)} \langle g, \phi_k \rangle_{L^2(X)} \) is the product in the Fourier domain.
- \( \phi_k(x) \) is the inverse Fourier trasform of the \( k \)-th eigenfunction of the Fourier transform.
Convolution

Euclidean case: given $f, g : [-\pi, \pi] \to \mathbb{R}$, the Fourier transform diagonalizes the convolution operator

$$(f \ast g)(x) = \mathcal{F}^{-1}(\mathcal{F}(f(x)) \cdot \mathcal{F}(g(x)))$$

Non-Euclidean case: given $f, g : L^2(X) \to \mathbb{R}$, the generalized convolution can be defined by analogy as

$$(f \ast g)(x) = \sum_{k \geq 1} \langle f, \phi_k \rangle_{L^2(X)} \langle g, \phi_k \rangle_{L^2(X)} \phi_k(x)$$

product in the Fourier domain

inverse Fourier transform

not shift-invariant!
Uncertainty principle

Poor spatial localization $\iff$ Good frequency localization
Uncertainty principle

Poor spatial localization $\iff$ Good frequency localization
Good spatial localization $\iff$ Poor frequency localization
**Uncertainty principle**

Poor spatial localization $\iff$ Good frequency localization

Good spatial localization $\iff$ Poor frequency localization

$\omega_0 = \frac{2\pi}{c_0}$

Spatial

Frequency
Spatial localization: Windowed Fourier Transform

**Main idea:** compute Fourier transform of a signal locally enveloped by a window

\[ WFT(f(x))(\xi, \omega) = \int_{-\pi}^{\pi} f(x)w(x - \xi)e^{-i\omega x} \, dx \]
Spatial localization: Windowed Fourier Transform

**Main idea:** compute Fourier transform of a signal locally enveloped by a window

\[ WFT(f(x))(\xi, \omega) = \int_{-\pi}^{\pi} f(x)w(x - \xi)e^{-i\omega x} dx \]
Spatial localization: Windowed Fourier Transform

**Main idea:** compute Fourier transform of a signal locally enveloped by a window

\[
WFT(f(x))(\xi, \omega) = \int_{-\pi}^{\pi} f(x)w(x - \xi)e^{-i\omega x} dx
\]
**Spatial localization: Windowed Fourier Transform**

**Main idea:** compute Fourier transform of a signal locally enveloped by a window

\[
WFT(f(x))(\xi, \omega) = \int_{-\pi}^{\pi} f(x)w(x - \xi)e^{-i\omega x} \, dx
\]
Main idea: compute Fourier transform of a signal locally enveloped by a window

\[
WFT(f(x))(\xi, \omega) = \int_{-\pi}^{\pi} f(x) w(x - \xi) e^{-i\omega x} \, dx
\]

atom \( g_{\xi,\omega}(x) \)
Spatial localization: Windowed Fourier Transform

**Main idea:** compute Fourier transform of a signal locally enveloped by a window

\[
WFT(f(x))(\xi, \omega) = \int_{-\pi}^{\pi} f(x) w(x - \xi) e^{-i\omega x} \, dx
\]

\[
\langle f(x), g_{\xi, \omega}(x) \rangle_{L^2(X)}
\]
**Translation**: convolution with delta

\[(T_x f)(x) = (f \ast \delta_x)(x)\]
**Translation**: convolution with delta

\[(T_{x'} f)(x) = (f * \delta_{x'})(x)\]

\[= \sum_{k \geq 1} \langle f, \phi_k \rangle_{L^2(X)} \langle \delta_{x'}, \phi_k \rangle_{L^2(X)} \phi_k(x)\]
**Translation:** convolution with delta

\[
(T_{x'}, f)(x) = (f \ast \delta_{x'})(x)
\]

\[
= \sum_{k \geq 1} \langle f, \phi_k \rangle_{L^2(X)} \langle \delta_{x'}, \phi_k \rangle_{L^2(X)} \phi_k(x)
\]

\[
= \sum_{k \geq 1} \hat{f}_k \phi_k(x') \phi_k(x)
\]
**Translation**: convolution with delta

\[
(T_{x', f})(x) = (f \ast \delta_{x'})(x)
\]

\[
= \sum_{k \geq 1} \langle f, \phi_k \rangle_{L^2(X)} \langle \phi_k \rangle_{L^2(X)} \phi_k(x)
\]

\[
= \sum_{k \geq 1} \hat{f}_k \phi_k(x') \phi_k(x)
\]

**Modulation**: multiplication by basis function

\[
(M_k f)(x) = f(x) \phi_k(x)
\]

*Graph case: Shuman, Ricaud, Vandergheynst 2014*
Windowed Fourier transform on manifolds

**Translation**: convolution with delta

\[
(T_{x', f})(x) = (f * \delta_{x'})(x)
\]

\[
= \sum_{k \geq 1} \langle f, \phi_k \rangle_{L^2(X)} \langle \delta_{x'}, \phi_k \rangle_{L^2(X)} \phi_k(x)
\]

\[
= \sum_{k \geq 1} \hat{f}_k \phi_k(x') \phi_k(x)
\]

**Modulation**: multiplication by basis function

\[
(M_k f)(x) = f(x) \phi_k(x)
\]

**Windowed Fourier transform (WFT)**:

\[
(S f)_{x, k} = \langle f, M_k T_x g \rangle_{L^2(X)}
\]

atom

Graph case: Shuman, Ricaud, Vandergheynst 2014
**Translation:** convolution with delta

\[(T_{x'}, f)(x) = (f \ast \delta_{x'})(x)\]

\[= \sum_{k \geq 1} \langle f, \phi_k \rangle_{L^2(X)} \langle \delta_{x'}, \phi_k \rangle_{L^2(X)} \phi_k(x)\]

\[= \sum_{k \geq 1} \hat{f}_k \phi_k(x') \phi_k(x)\]

**Modulation:** multiplication by basis function

\[(M_k f)(x) = f(x) \phi_k(x)\]

**Windowed Fourier transform (WFT):**

\[(Sf)_{x,k} = \langle f, M_k T_x g \rangle_{L^2(X)} = \sum_{\ell \geq 1} \hat{g}_\ell \phi_\ell(x) \langle f, \phi_\ell \phi_k \rangle_{L^2(X)}\]

Graph case: Shuman, Ricaud, Vandergheynst 2014
Windowed Fourier transform on manifolds

$\hat{g}_1$

$\hat{g}_2$

eigenvalues
• Review of shape descriptors
• Extending convolution to shapes
• **Proposed architecture**
• Results
LSCNN architecture: Fully connected (FC) layer

\[ f_{q_{\text{out}}}^{\text{out}}(x) = \sum_{p=1}^{P} w_{qp} f_{p_{\text{in}}}^{\text{in}}(x), \quad q = 1, \ldots, Q \]

- Linear combination of \( P \) inputs into \( Q \) outputs
- Parametrized by \( QP \) coefficients
LSCNN architecture: ReLU layer

Typically, the ReLU non-linearity $\xi(t) = \max(t, 0)$ is used.

- Typically applied after FC layer.
- Fixed layer.

Mathematically,

$$f_{q}^{\text{out}}(x) = \xi(f_{q}^{\text{in}}(x)), \quad q = 1, \ldots, Q$$
LSCNN architecture: WFT layer

\[ f_{q_{\text{out}}} (x) = \sum_{p=1}^{P} \sum_{k=1}^{K} a_{qpk} |(Sf_{p_{\text{in}}})_{x,k}| \]

where for each input dimension WFT uses a different window

\[ (Sf_{p_{\text{in}}})_{x,k} = \sum_{\ell \geq 1} \gamma_p (\lambda \ell) \phi_{\ell}(x) \langle f_{p_{\text{in}}}, \phi_{\ell} \phi_k \rangle_{L^2(X)} \]

\[ \sum_{m=1}^{M} b_{pm} \beta_m (\lambda) \]

- Learn window for each input dimension (coefficients \( b_{pm} \))
- Learn bank of filters for each WFT (coefficients \( a_{qpk} \))
Previous descriptors are particular configurations of LSCNN!
Previous descriptors are particular configurations of LSCNN!

Aubry, Schlickewei, Cremers 2011
Previous descriptors are particular configurations of LSCNN!
Previous descriptors are particular configurations of LSCNN!
Architecture: Localized Spectral Convolutional Neural Network

**Input layer** \( M \)-dim

**FC layer**

**ReLU layer**

**WFT layer**

**Output layer** \( Q \)-dim

- \( f_{1}^{in} \)
- \( f_{2}^{in} \)
- \( \vdots \)
- \( f_{M}^{in} \)

- \( S_{1} \)
- \( S_{2} \)
- \( \vdots \)
- \( S_{Q} \)

- \( \xi \)
- \( \xi \)
- \( \vdots \)
- \( \xi \)

- \( \Sigma \)
- \( \Sigma \)
- \( \vdots \)
- \( \Sigma \)

- \( \text{abs} \)
- \( \text{abs} \)
- \( \vdots \)
- \( \text{abs} \)

- \( S_{1} \)
- \( S_{2} \)
- \( \vdots \)
- \( S_{Q} \)

- \( f_{1}^{out} \)
- \( f_{2}^{out} \)
- \( \vdots \)
- \( f_{Q}^{out} \)
- Review of shape descriptors
- Extending convolution to shapes
- Proposed architecture
- **Results**
Descriptor robustness

Heat Kernel Signature (HKS)
Descriptor robustness

Wave Kernel Signature (WKS)
Descriptor robustness

Optimal Spectral Descriptor (OSD)
Localized Spectral Convolutional Neural Network (LSCNN)
Results

Descriptor performance (training: FAUST, testing: FAUST)
Results

Descriptor performance (training: FAUST, testing: SCAPE)
Results

Cumulative Matching Characteristic

Receiver Operating Characteristic

Correspondence quality

Descriptor performance (training: SCAPE, testing: SCAPE)
Results

Descriptor performance (training: SCAPE, testing: FAUST)
Descriptor robustness (point clouds)

Heat Kernel Signature (HKS)
Descriptor robustness (point clouds)

Wave Kernel Signature (WKS)
Descriptor robustness (point clouds)

Localized Spectral Convolutional Neural Network (LSCNN)
Summary

- Convolutional Neural Networks on manifolds
- Generalizes previous approaches (HKW, WKS, OSD, ISC)
- State-of-the-art results
- Extensions to other shape representations (e.g. point clouds)
In collaboration with

J. Masci  M. Bronstein  S. Melzi  U. Castellani  P. Vandergheynst
Thank you!

www.inf.usi.ch/phd/boscaini
Experimental settings

- **Data:** FAUST\(^1\) and SCAPE\(^2\)

- **Settings**
  - **Input:** \( M = 150 \)-dimensional geometry vectors, computed using \( K = 300 \) Laplace-Beltrami eigenfunctions with B-spline bases
  - **Output:** \( Q = 16 \)-dimensional descriptors

- **Timings**
  - **Training:** \(< 2h\)
  - **Testing:** 30K vertices per second
  - **Precomputation:** \( \approx 10s \) for a shape with 7K vertices

---

\(^1\) Bogo et al. 2014, \(^2\) Anguelov et al. 2005
Geodesic polar coordinates

- Local system of **geodesic polar coordinates** at $x$
  - $\rho$-level set of geodesic distance function $d_{x}(x, \xi)$, truncated at $\rho_{0}$
  - Points along geodesic $\Gamma_{\theta}(x)$ emanating from $x$ in direction $\theta$
Geodesic polar coordinates

- Local system of **geodesic polar coordinates** at $x$
  - $\rho$-level set of geodesic distance function $d_X(x, \xi)$, truncated at $\rho_0$
  - Points along geodesic $\Gamma_{\theta}(x)$ emanating from $x$ in direction $\theta$

- Local chart: bijective map

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

from manifold to local coordinates $(\rho, \theta)$ around $x$
Geodesic polar coordinates

- **Local system of geodesic polar coordinates** at $x$
  - $\rho$-level set of geodesic distance function $d_X(x, \xi)$, truncated at $\rho_0$
  - Points along geodesic $\Gamma_\theta(x)$ emanating from $x$ in direction $\theta$

- **Local chart**: bijective map
  
  $$\Omega(x) : B_{\rho_0}(x) \to [0, \rho_0] \times [0, 2\pi)$$

  From manifold to local coordinates $(\rho, \theta)$ around $x$

- **Patch operator** applied to $f \in L^2(X)$
  
  $$(D(x)f)(\rho, \theta) = (f \circ \Omega^{-1}(x))(\rho, \theta)$$