Deep learning on geometric data

Michael Bronstein

University of Lugano

Intel Corporation

Prague, 12 November 2015

INVISION A New Dimension to intel

(Acquired by Intel in 2012)


intel
REALSENSE
TECHNOLOGY

Different form factor computers featuring Intel RealSense 3D camera

Deluge of geometric data

KINECT for XBOX 360 SoftKinetic

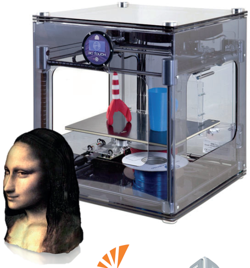
intel REALSENSE

3D sensors

Google 3D warehouse

shapeways

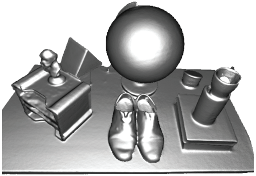
Repositories



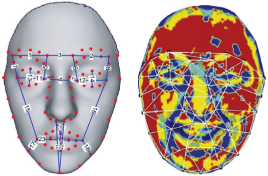
Stratasys 3DSYSTEMS

3D printers

Applications

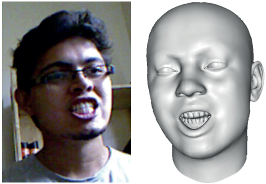


Reconstruction

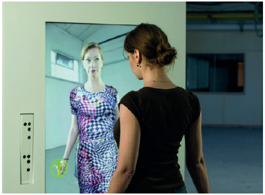


Recognition

Retrieval



Avatars

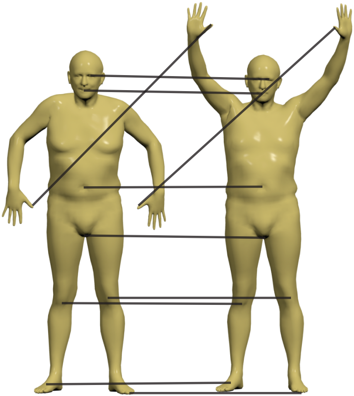


Virtual dressing

Gesture control

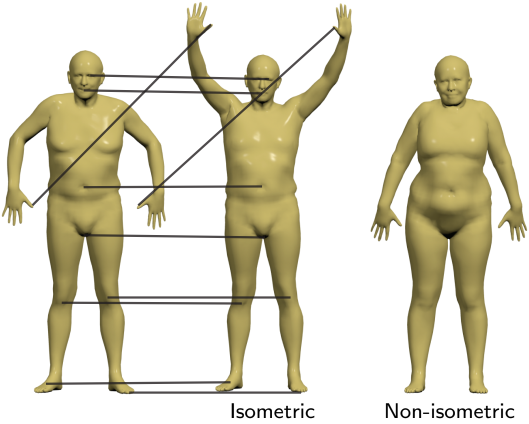
Images: Davison et al. 2011; Zafeiriou et al. 2012; Kim et al. 2013; Faceshift; Fashion3D; Minority report

Basic problems: shape similarity and correspondence

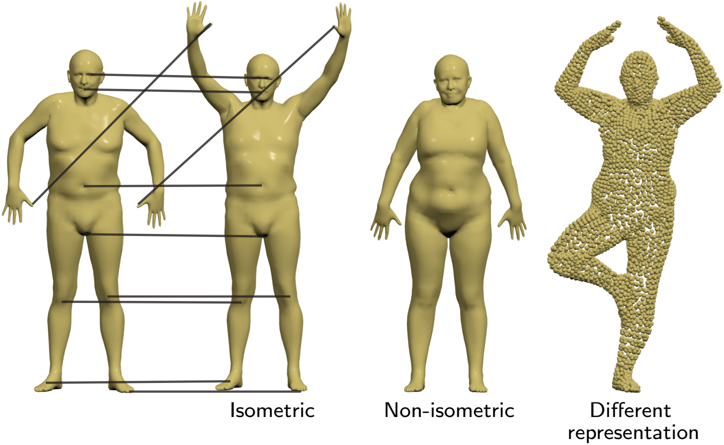


Isometric

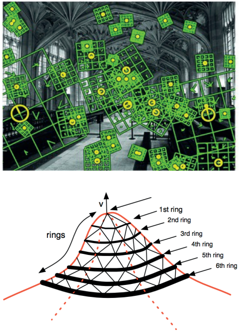
Basic problems: shape similarity and correspondence



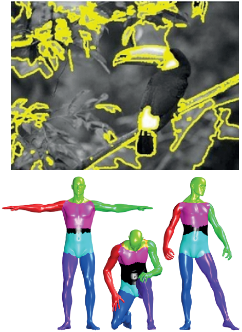
Basic problems: shape similarity and correspondence



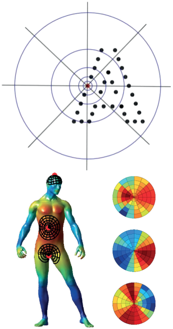
3D feature descriptors



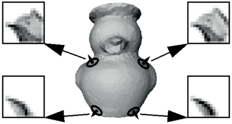
SIFT[1] / MeshHOG[2]



MSER[3] / ShapeMSER[4]



(Intrinsic[6]) Shape context[5]



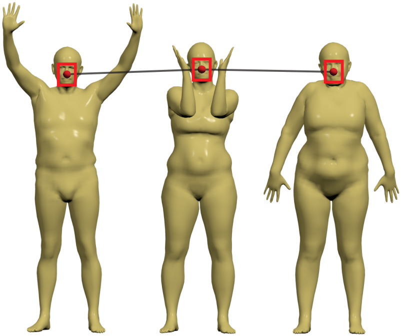
Spin image[7]

Heat kernel signature[8]

[1] Lowe 2004; [2] Zaharescu et al. 2009; [3] Matas et al. 2002; [4] Litman et al. 2010; [5] Belongie et al. 2000; [6] Kokkinos et al. 2012; [7] Johnson et al. 1999; [8] Sun et al. 2009

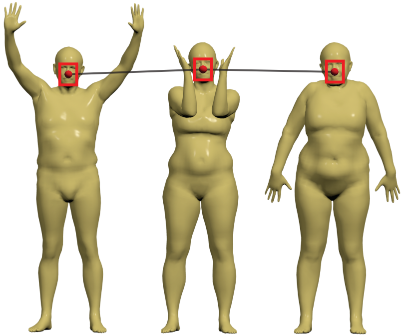
Task-specific features

Correspondence

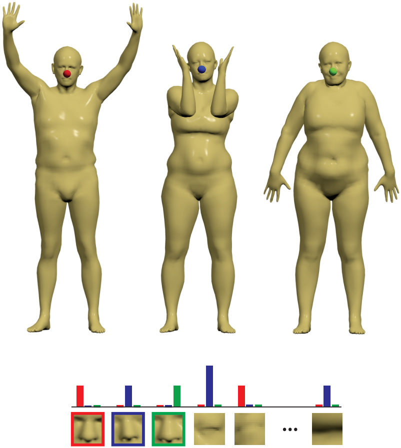


Task-specific features

Correspondence



Similarity



Deep learning (r)evolution

2012

ImageNet Classification with Deep Convolutional Neural Networks

Alex Krizhevsky
University of Toronto
kriz@cs.utoronto.ca

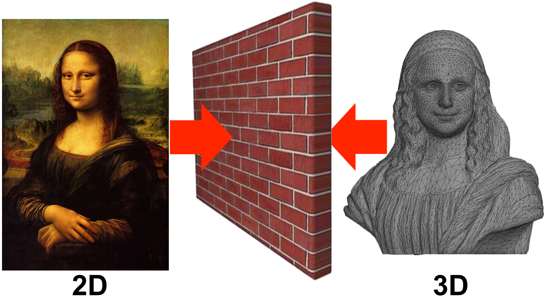
Ilya Sutskever
University of Toronto
ilya@cs.utoronto.ca

Geoffrey E. Hinton
University of Toronto
hinton@cs.utoronto.ca

2014

VERY DEEP CONVOLUTIONAL NETWORKS FOR LARGE-SCALE IMAGE RECOGNITION

Karen Simonyan* **& Andrew Zisserman+**
Visual Geometry Group, Department of Engineering Science, University of Oxford
{karen,az}@robots.ox.ac.uk


2D
3D

Outline

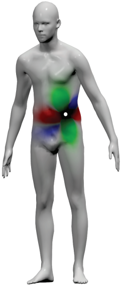
- Background: Laplacians and spectral analysis on manifolds
- Spectral descriptors (heat- and wave-kernel signatures)
- Convolutional neural networks on manifolds

Outline

- Background: Laplacians and spectral analysis on manifolds
- Spectral descriptors (heat- and wave-kernel signatures)
- Convolutional neural networks on manifolds

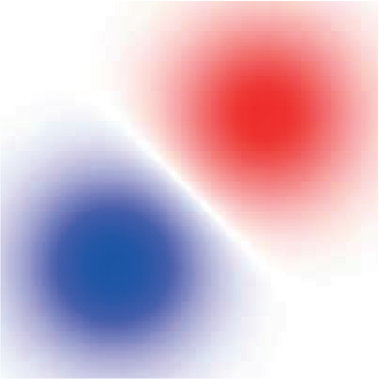
Geodesic convolution

Windowed Fourier transform



Anisotropic diffusion

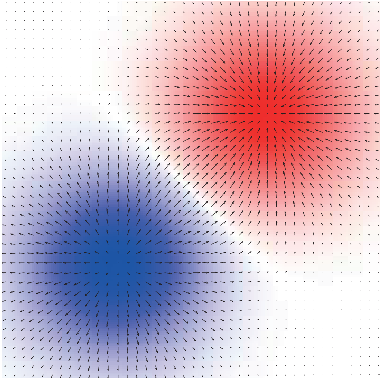
Laplacian in one minute



Smooth scalar field f

Laplacian in one minute

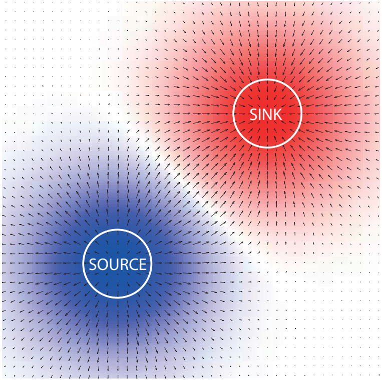
- Gradient $\nabla f(x)$ = 'direction of the steepest increase of f at x'



Smooth scalar field f

Laplacian in one minute

- Gradient $\nabla f(x) =$ 'direction of the steepest increase of f at x'
- Divergence $\text{div}(F(x)) =$ 'density of an outward flux of F from an infinitesimal volume around x'



Smooth vector field F

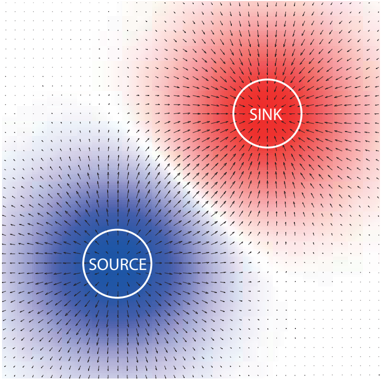
Laplacian in one minute

- Gradient $\nabla f(x) =$ 'direction of the steepest increase of f at x'
- Divergence $\text{div}(F(x)) =$ 'density of an outward flux of F from an infinitesimal volume around x'

  Divergence theorem:

  $$\int_V \text{div}(\text{F}) dV = \int_{\partial V} \langle F, \hat{n} \rangle dS$$

  '$\sum$ sources + sinks = net flow'



Smooth vector field F

Laplacian in one minute

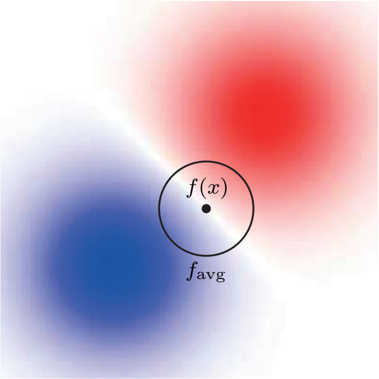
- Gradient $\nabla f(x) =$ 'direction of the steepest increase of f at x'

- Divergence $\mathrm{div}(F(x)) =$ 'density of an outward flux of F from an infinitesimal volume around x'

  Divergence theorem:

  $$\int_V \mathrm{div}(\mathrm{F}) dV = \int_{\partial V} \langle F, \hat{n} \rangle dS$$

  '$\sum$ sources + sinks = net flow'



- Laplacian $\Delta f(x) = -\mathrm{div}(\nabla f(x))$
  'difference between $f(x)$ and the average of f on an infinitesimal sphere around x' (consequence of the Divergence theorem)

*We define Laplacian with negative sign

Physical application: heat equation

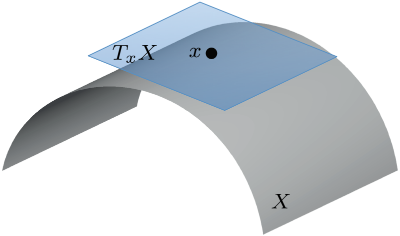
$$f_t = -c\Delta f$$

Newton's law of cooling: rate of change of the temperature of an object is proportional to the difference between its own temperature and the temperature of the surrounding

c [m^2/sec] $=$ thermal diffusivity constant (assumed $= 1$)

Riemannian geometry in one minute

- Tangent plane $T_x X$ = local Euclidean representation of manifold (surface) X around x



*We assume manifolds without boundary for simplicity

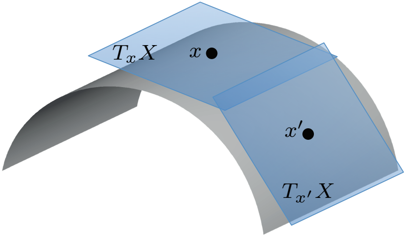
Riemannian geometry in one minute

- Tangent plane $T_x X$ = local Euclidean representation of manifold (surface) X around x

- Riemannian metric

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

depending smoothly on x



*We assume manifolds without boundary for simplicity

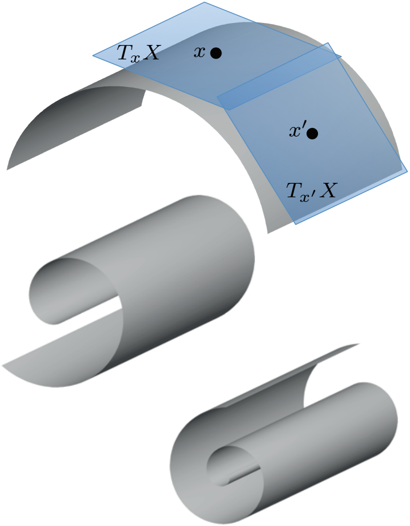
Riemannian geometry in one minute

- Tangent plane $T_x X$ = local Euclidean representation of manifold (surface) X around x
- Riemannian metric

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

depending smoothly on x

Isometry = metric-preserving shape deformation



*We assume manifolds without boundary for simplicity

Riemannian geometry in one minute

- Tangent plane $T_x X$ = local Euclidean representation of manifold (surface) X around x

- Riemannian metric

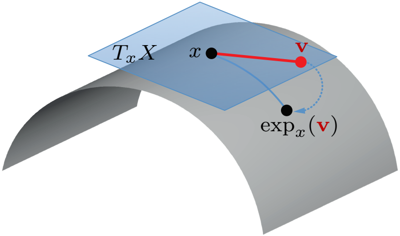
$$\langle \cdot, \cdot \rangle_{T_x X} : T_x X \times T_x X \to \mathbb{R}$$

depending smoothly on x

Isometry = metric-preserving shape deformation

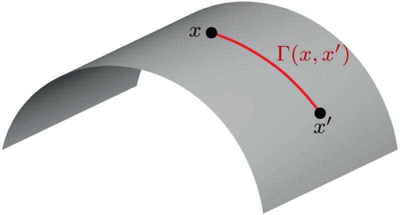
- Exponential map

$$\exp_x : T_x X \to X$$



*We assume manifolds without boundary for simplicity

Riemannian geometry in one minute



- Tangent plane $T_x X$ = local Euclidean representation of manifold (surface) X around x

- Riemannian metric

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

  depending smoothly on x

  Isometry = metric-preserving shape deformation

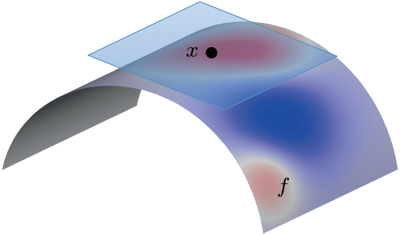
- Exponential map

  $$\exp_x : T_x X \to X$$

- Geodesic = shortest path on X between x and x'

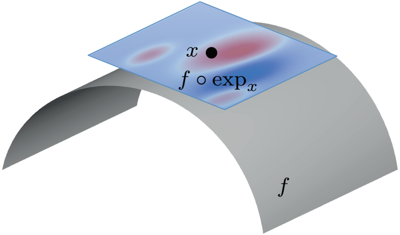
*We assume manifolds without boundary for simplicity

Laplace-Beltrami operator



Smooth field $f : X \to \mathbb{R}$

Laplace-Beltrami operator



Smooth field $f \circ \exp_x : T_xX \to \mathbb{R}$

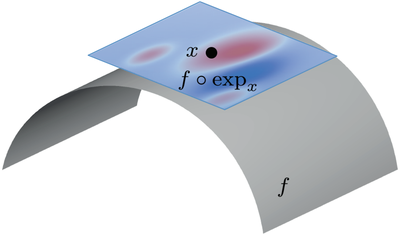
Laplace-Beltrami operator

- Intrinsic gradient

$$\nabla_X f(x) = \nabla(f \circ \exp_x)(\mathbf{0})$$

Taylor expansion

$$(f \circ \exp_x)(\mathbf{v}) \approx f(x) + \langle \nabla_X f(x), \mathbf{v} \rangle_{T_x X}$$



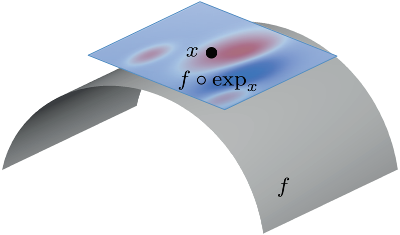
Laplace-Beltrami operator

- Intrinsic gradient

$$\nabla_X f(x) = \nabla(f \circ \exp_x)(\mathbf{0})$$

Taylor expansion

$$(f \circ \exp_x)(\mathbf{v}) \approx f(x) + \langle \nabla_X f(x), \mathbf{v} \rangle_{T_x X}$$



- Laplace-Beltrami operator

$$\Delta_X f(x) = \Delta(f \circ \exp_x)(\mathbf{0})$$

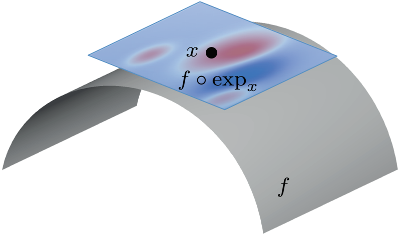
Laplace-Beltrami operator

- Intrinsic gradient

$$\nabla_X f(x) = \nabla(f \circ \exp_x)(\mathbf{0})$$

Taylor expansion

$$(f \circ \exp_x)(\mathbf{v}) \approx f(x) + \langle \nabla_X f(x), \mathbf{v} \rangle_{T_x X}$$



- Laplace-Beltrami operator

$$\Delta_X f(x) = \Delta(f \circ \exp_x)(\mathbf{0})$$

  - Intrinsic (expressed solely in terms of the Riemannian metric)

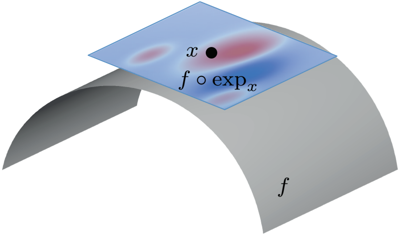
Laplace-Beltrami operator

- Intrinsic gradient

$$\nabla_X f(x) = \nabla(f \circ \exp_x)(\mathbf{0})$$

Taylor expansion

$$(f \circ \exp_x)(\mathbf{v}) \quad \approx \quad f(x) + \langle \nabla_X f(x), \mathbf{v} \rangle_{T_x X}$$

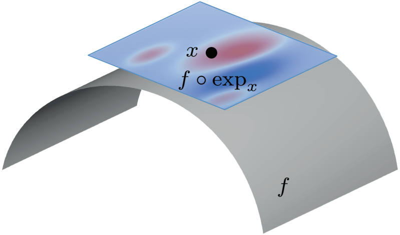


- Laplace-Beltrami operator

$$\Delta_X f(x) = \Delta(f \circ \exp_x)(\mathbf{0})$$

  - Intrinsic (expressed solely in terms of the Riemannian metric)
  - Isometry-invariant

Laplace-Beltrami operator



- Intrinsic gradient

$$\nabla_X f(x) = \nabla(f \circ \exp_x)(\mathbf{0})$$

Taylor expansion

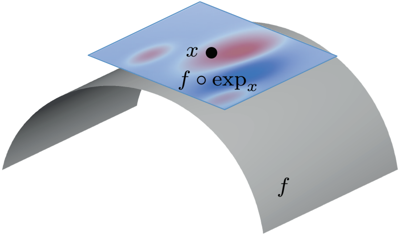
$$(f \circ \exp_x)(\mathbf{v}) \approx f(x) + \langle \nabla_X f(x), \mathbf{v} \rangle_{T_x X}$$

- Laplace-Beltrami operator

$$\Delta_X f(x) = \Delta(f \circ \exp_x)(\mathbf{0})$$

  - Intrinsic (expressed solely in terms of the Riemannian metric)
  - Isometry-invariant
  - Self-adjoint $\langle \Delta_X f, g \rangle_{L^2(X)} = \langle f, \Delta_X g \rangle_{L^2(X)}$

Laplace-Beltrami operator



- Intrinsic gradient

$$\nabla_X f(x) = \nabla(f \circ \exp_x)(\mathbf{0})$$

Taylor expansion

$$(f \circ \exp_x)(\mathbf{v}) \approx f(x) + \langle \nabla_X f(x), \mathbf{v} \rangle_{T_x X}$$

- Laplace-Beltrami operator

$$\Delta_X f(x) = \Delta(f \circ \exp_x)(\mathbf{0})$$

  - Intrinsic (expressed solely in terms of the Riemannian metric)
  - Isometry-invariant
  - Self-adjoint $\langle \Delta_X f, g \rangle_{L^2(X)} = \langle f, \Delta_X g \rangle_{L^2(X)} \Rightarrow$ orthogonal eigenfunctions

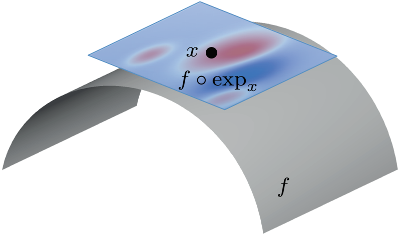
Laplace-Beltrami operator

- Intrinsic gradient

$$\nabla_X f(x) = \nabla(f \circ \exp_x)(\mathbf{0})$$

Taylor expansion

$$(f \circ \exp_x)(\mathbf{v}) \approx f(x) + \langle \nabla_X f(x), \mathbf{v} \rangle_{T_x X}$$

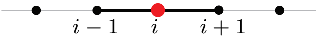


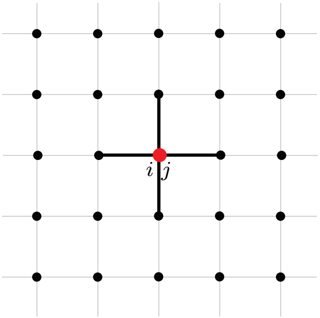
- Laplace-Beltrami operator

$$\Delta_X f(x) = \Delta(f \circ \exp_x)(\mathbf{0})$$

  - Intrinsic (expressed solely in terms of the Riemannian metric)
  - Isometry-invariant
  - Self-adjoint $\langle \Delta_X f, g \rangle_{L^2(X)} = \langle f, \Delta_X g \rangle_{L^2(X)}$ ⇒ orthogonal eigenfunctions
  - Positive semidefinite ⇒ non-negative eigenvalues

Discrete Laplacian (Euclidean)





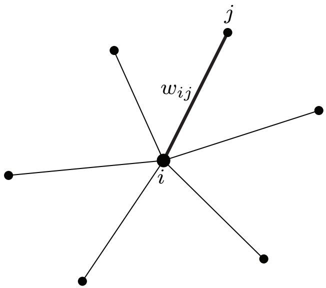
One-dimensional

$$(\Delta f)_i \approx 2f_i - f_{i-1} - f_{i+1}$$

Two-dimensional

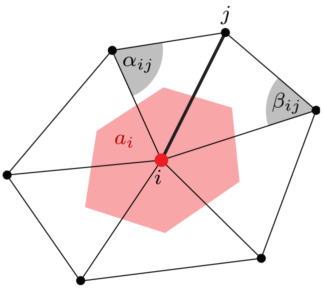
$$\begin{aligned}(\Delta f)_{ij} &\approx 4f_{ij} - f_{i-1,j} - f_{i+1,j} \\ &- f_{i,j-1} - f_{i,j+1}\end{aligned}$$

Discrete Laplacian (non-Euclidean)



Undirected graph (V, E)

$$(\Delta f)_i \approx \sum_{(i,j)\in E} w_{ij}(f_i - f_j)$$



Triangular mesh (V, E, F)

$$(\Delta f)_i \approx \frac{1}{a_i} \sum_{(i,j)\in E} \frac{\cot \alpha_{ij} + \cot \beta_{ij}}{2}(f_i - f_j)$$

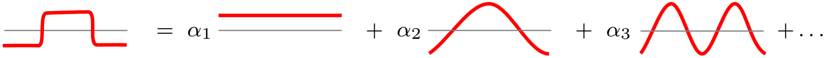
a_i = local area element

Tutte 1963; MacNeal 1949; Duffin 1959; Pinkall, Polthier 1993

Fourier analysis (Euclidean spaces)

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

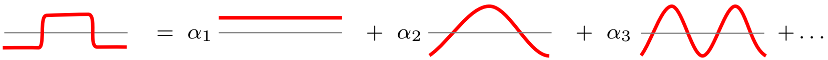
$$f(x) = \sum_{\omega} \quad \frac{1}{2\pi} \int_{-\pi}^{\pi} f(\xi) e^{i\omega\xi} d\xi \quad e^{-i\omega x}$$



Fourier analysis (Euclidean spaces)

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

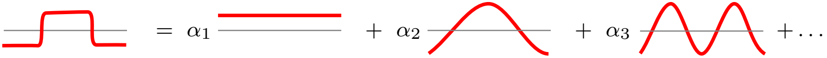
$$f(x) = \sum_{\omega} \underbrace{\frac{1}{2\pi} \int_{-\pi}^{\pi} f(\xi) e^{i\omega\xi} d\xi}_{\hat{f}(\omega) = \langle f, e^{-i\omega x} \rangle_{L^2([-\pi,\pi])}} \; e^{-i\omega x}$$



Fourier analysis (Euclidean spaces)

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

$$f(x) = \sum_{\omega} \underbrace{\frac{1}{2\pi} \int_{-\pi}^{\pi} f(\xi) e^{i\omega\xi} d\xi}_{\hat{f}(\omega) = \langle f, e^{-i\omega x} \rangle_{L^2([-\pi,\pi])}} \; e^{-i\omega x}$$

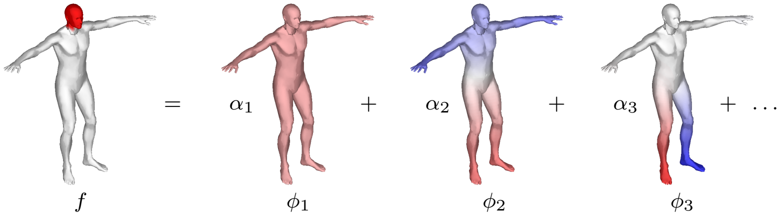


Fourier basis = Laplacian eigenfunctions: $\Delta e^{-i\omega x} = \omega^2 e^{-i\omega x}$

Fourier analysis (non-Euclidean spaces)

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

$$f(x) = \sum_{k \geq 1} \underbrace{\int_X f(\xi)\phi_k(\xi)d\xi}_{\hat{f}_k = \langle f, \phi_k \rangle_{L^2(X)}} \; \phi_k(x)$$



Fourier basis = Laplacian eigenfunctions: $\Delta_X \phi_k(x) = \lambda_k \phi_k(x)$

Heat diffusion on manifolds

$$\begin{cases} f_t(x,t) = -\Delta_X f(x,t) \\ f(x,0) = f_0(x) \end{cases}$$

- $f(x,t)$ = amount of heat at point x at time t
- $f_0(x)$ = initial heat distribution

Heat diffusion on manifolds

$$\begin{cases} f_t(x,t) = -\Delta_X f(x,t) \\ f(x,0) = f_0(x) \end{cases}$$

- $f(x,t)$ = amount of heat at point x at time t
- $f_0(x)$ = initial heat distribution

Solution of the heat equation expressed through the heat operator

$$f(x,t) = e^{-t\Delta_X} f_0(x)$$

Heat diffusion on manifolds

$$\begin{cases} f_t(x,t) = -\Delta_X f(x,t) \\ f(x,0) = f_0(x) \end{cases}$$

- $f(x,t)$ = amount of heat at point x at time t
- $f_0(x)$ = initial heat distribution

Solution of the heat equation expressed through the heat operator

$$f(x,t) \quad = \quad e^{-t\Delta_X} f_0(x) = \sum_{k \geq 1} \langle f_0, \phi_k \rangle_{L^2(X)} e^{-t\lambda_k} \phi_k(x)$$

Heat diffusion on manifolds

$$\begin{cases} f_t(x,t) = -\Delta_X f(x,t) \\ f(x,0) = f_0(x) \end{cases}$$

- $f(x,t)$ = amount of heat at point x at time t
- $f_0(x)$ = initial heat distribution

Solution of the heat equation expressed through the heat operator

$$\begin{aligned} f(x,t) &= e^{-t\Delta_X} f_0(x) = \sum_{k\geq 1} \langle f_0, \phi_k \rangle_{L^2(X)} e^{-t\lambda_k} \phi_k(x) \\ &= \int_X f_0(\xi) \sum_{k\geq 1} e^{-t\lambda_k} \phi_k(x) \phi_k(\xi)\, d\xi \end{aligned}$$

Heat diffusion on manifolds

$$\begin{cases} f_t(x,t) = -\Delta_X f(x,t) \\ f(x,0) = f_0(x) \end{cases}$$

- $f(x,t)$ = amount of heat at point x at time t
- $f_0(x)$ = initial heat distribution

Solution of the heat equation expressed through the heat operator

$$\begin{aligned} f(x,t) &= e^{-t\Delta_X} f_0(x) = \sum_{k\geq 1} \langle f_0, \phi_k \rangle_{L^2(X)} e^{-t\lambda_k} \phi_k(x) \\ &= \int_X f_0(\xi) \underbrace{\sum_{k\geq 1} e^{-t\lambda_k} \phi_k(x)\phi_k(\xi)}_{\text{heat kernel } h_t(x,\xi)} d\xi \end{aligned}$$

Heat diffusion on manifolds

$$\begin{cases} f_t(x,t) = -\Delta_X f(x,t) \\ f(x,0) = f_0(x) \end{cases}$$

- $f(x,t)$ = amount of heat at point x at time t
- $f_0(x)$ = initial heat distribution

Solution of the heat equation expressed through the heat operator

$$\begin{aligned} f(x,t) &= e^{-t\Delta_X} f_0(x) = \sum_{k\geq 1} \langle f_0, \phi_k \rangle_{L^2(X)} e^{-t\lambda_k} \phi_k(x) \\ &= \int_X f_0(\xi) \underbrace{\sum_{k\geq 1} e^{-t\lambda_k} \phi_k(x)\phi_k(\xi)}_{\text{heat kernel } h_t(x,\xi)} d\xi \end{aligned}$$

- "impulse response" to a delta-function at ξ

Heat diffusion on manifolds

$$\begin{cases} f_t(x,t) = -\Delta_X f(x,t) \\ f(x,0) = f_0(x) \end{cases}$$

- $f(x,t)$ = amount of heat at point x at time t
- $f_0(x)$ = initial heat distribution

Solution of the heat equation expressed through the heat operator

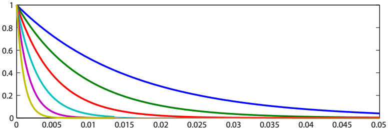
$$\begin{aligned} f(x,t) &= e^{-t\Delta_X} f_0(x) = \sum_{k\geq 1} \langle f_0, \phi_k \rangle_{L^2(X)} e^{-t\lambda_k} \phi_k(x) \\ &= \int_X f_0(\xi) \underbrace{\sum_{k\geq 1} e^{-t\lambda_k} \phi_k(x)\phi_k(\xi)}_{\text{heat kernel } h_t(x,\xi)} d\xi \end{aligned}$$

- "impulse response" to a delta-function at ξ
- "how much heat is transferred from point x to ξ in time t"

Spectral descriptors

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

Heat Kernel Signature (HKS)

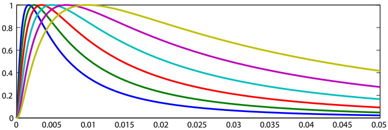


Low-pass filter bank

$$\tau_i(\lambda) = \exp(-\lambda t_i)$$

Heat autodiffusivity

Wave Kernel Signature (WKS)



Band-pass filter bank

$$\tau_i(\lambda) = \exp\left(-\frac{(\log e_i - \log \lambda)^2}{\sigma^2}\right)$$

Probability of a quantum particle

Sun, Ovsjanikov, Guibas 2009; Aubry, Schlickewei, Cremers 2011

Optimal spectral descriptors

$$\mathbf{f}_{\boldsymbol{\tau}}(x) \quad = \quad \sum_{k \geq 1} \begin{pmatrix} \tau_1(\lambda_k) \\ \vdots \\ \tau_Q(\lambda_k) \end{pmatrix} \phi_k^2(x)$$

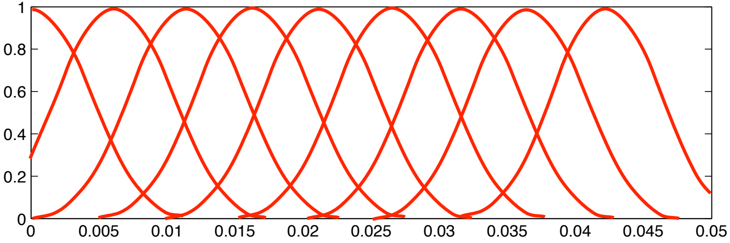
parametrized by frequency responses $\boldsymbol{\tau}(\lambda) = (\tau_1(\lambda), \ldots, \tau_Q(\lambda))^\top$

Litman, Bronstein 2014

Optimal spectral descriptors

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

parametrized by frequency responses $\boldsymbol{\tau}(\lambda) = (\tau_1(\lambda), \ldots, \tau_Q(\lambda))^\top$ represented in some fixed basis $\beta_1(\lambda), \ldots, \beta_M(\lambda)$ by an $Q \times M$ matrix $\mathbf{A}$



Litman, Bronstein 2014

Optimal spectral descriptors

$$\mathbf{f}_{\mathbf{A}}(x) = \mathbf{A} \underbrace{\sum_{k \geq 1} \begin{pmatrix} \beta_1(\lambda_k) \\ \vdots \\ \beta_M(\lambda_k) \end{pmatrix} \phi_k^2(x)}_{\mathbf{g}(x)}$$

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

Litman, Bronstein 2014

Optimal spectral descriptors

$$\mathbf{f}_{\mathbf{A}}(x) \quad = \quad \mathbf{A} \underbrace{\sum_{k \geq 1} \begin{pmatrix} \beta_1(\lambda_k) \\ \vdots \\ \beta_M(\lambda_k) \end{pmatrix} \phi_k^2(x)}_{\mathbf{g}(x)} = \mathbf{A}\mathbf{g}(x)$$

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

Litman, Bronstein 2014

Optimal spectral descriptors

$$\mathbf{f}_{\mathbf{A}}(x) \;=\; \mathbf{A} \underbrace{\sum_{k\geq 1} \begin{pmatrix} \beta_1(\lambda_k) \\ \vdots \\ \beta_M(\lambda_k) \end{pmatrix} \phi_k^2(x)}_{\mathbf{g}(x)} = \mathbf{A}\mathbf{g}(x)$$

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

- Optimal $\mathbf{A}$ in the spirit of Wiener filter:
  - attenuate frequencies with large noise content (deformation)
  - pass frequencies with large signal content (discriminative geometric features)

Litman, Bronstein 2014

Optimal spectral descriptors

$$\mathbf{f}_{\mathbf{A}}(x) = \mathbf{A} \underbrace{\sum_{k \geq 1} \begin{pmatrix} \beta_1(\lambda_k) \\ \vdots \\ \beta_M(\lambda_k) \end{pmatrix} \phi_k^2(x)}_{\mathbf{g}(x)} = \mathbf{A}\mathbf{g}(x)$$

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

- Optimal $\mathbf{A}$ in the spirit of Wiener filter:
  - attenuate frequencies with large noise content (deformation)
  - pass frequencies with large signal content (discriminative geometric features)
- Hard to model axiomatically...

Litman, Bronstein 2014

Optimal spectral descriptors

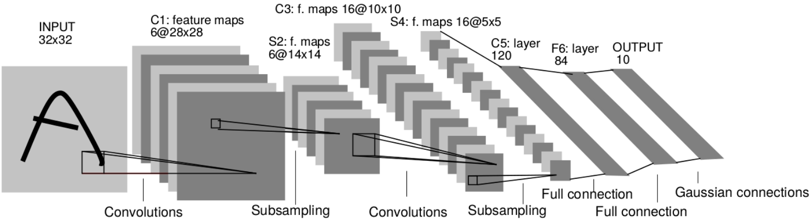
$$\mathbf{f}_{\mathbf{A}}(x) = \mathbf{A} \underbrace{\sum_{k \geq 1} \begin{pmatrix} \beta_1(\lambda_k) \\ \vdots \\ \beta_M(\lambda_k) \end{pmatrix} \phi_k^2(x)}_{\mathbf{g}(x)} = \mathbf{A}\mathbf{g}(x)$$

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

- Optimal $\mathbf{A}$ in the spirit of Wiener filter:
  - attenuate frequencies with large noise content (deformation)
  - pass frequencies with large signal content (discriminative geometric features)
- Hard to model axiomatically...
- ...yet easy to learn from examples!

Litman, Bronstein 2014

Convolutional neural networks

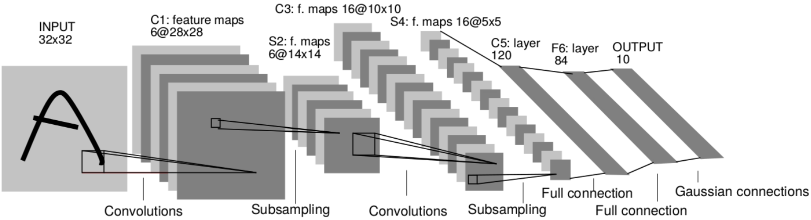


Typical CNN architecture

- Combination of convolution and pooling layers

Fukushima 1980; LeCun et al. 1989; Image: H. Wang

Convolutional neural networks

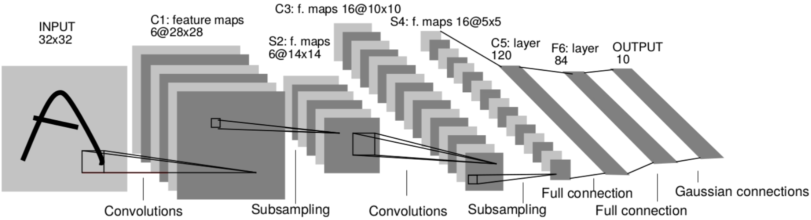


Typical CNN architecture

- Combination of convolution and pooling layers
- Learn hierarchical abstractions from data with little prior knowledge

Fukushima 1980; LeCun et al. 1989; Image: H. Wang

Convolutional neural networks

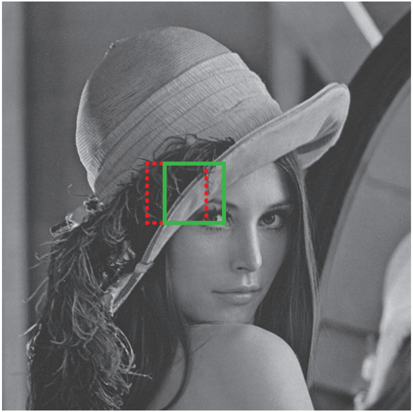


Typical CNN architecture

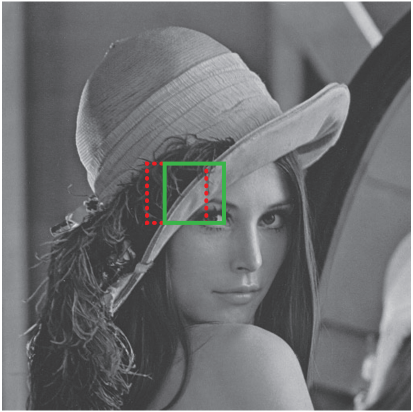
- Combination of convolution and pooling layers
- Learn hierarchical abstractions from data with little prior knowledge
- State-of-the-art performance in a wide range of applications

Fukushima 1980; LeCun et al. 1989; Image: H. Wang

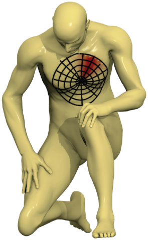
Convolution



Convolution

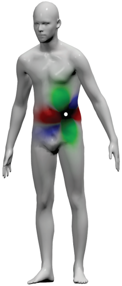


?



Geodesic convolution

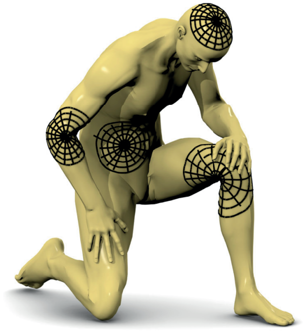
Windowed Fourier transform



Anisotropic diffusion

Geodesic polar coordinates

- Local system of geodesic polar coordinates at x
  - ρ-level set of geodesic distance function $d_X(x,\xi)$, truncated at ρ_0
  - points along geodesic $\Gamma_\theta(x)$ emanating from x in direction θ



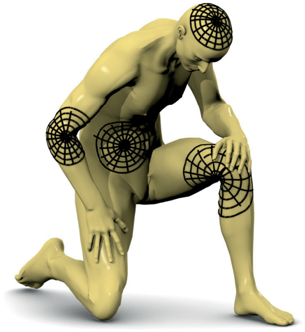
Kokkinos, Bronstein[2], Litman 2012

Geodesic polar coordinates

- Local system of geodesic polar coordinates at x
  - ρ-level set of geodesic distance function $d_X(x,\xi)$, truncated at ρ_0
  - points along geodesic $\Gamma_\theta(x)$ emanating from x in direction θ
- Local chart: bijective map

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

from manifold to local coordinates (ρ,θ) around x



Kokkinos, Bronstein^2, Litman 2012

Geodesic polar coordinates

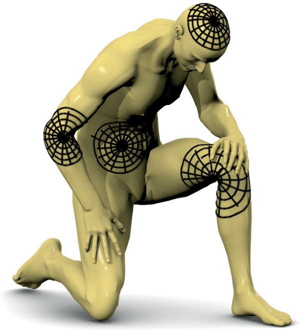
- Local system of geodesic polar coordinates at x
  - ρ-level set of geodesic distance function $d_X(x, \xi)$, truncated at ρ_0
  - points along geodesic $\Gamma_\theta(x)$ emanating from x in direction θ
- Local chart: bijective map

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

  from manifold to local coordinates (ρ, θ) around x

- Patch operator applied to $f \in L^2(X)$

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



Kokkinos, Bronstein^2, Litman 2012

Patch operator construction

$$(D(x)f)(\rho,\theta) = \frac{\int_X v_\rho(x,\xi) v_\theta(x,\xi) f(\xi) d\xi}{\int_X v_\rho(x,\xi) v_\theta(x,\xi) d\xi}$$

Radial weight

$$v_\rho(x,\xi) \propto e^{-(d_X(x,\xi)-\rho)^2/\sigma_\rho^2}$$

Angular weight

$$v_\theta(x,\xi) \propto e^{-d_X^2(\Gamma(x,\theta),\xi)/\sigma_\theta^2}$$

Kokkinos, Bronstein^2, Litman 2012

Geodesic convolution

- Geodesic convolution $=$ apply filter a to patches extracted from $f \in L^2(X)$ in local geodesic polar coordinates

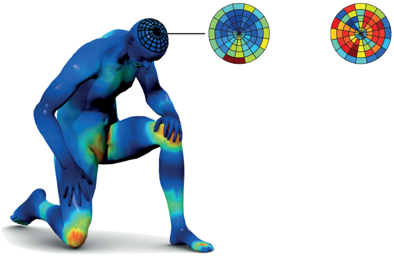
$$(f \star a)(x) \quad = \quad \sum_{\theta, r} (D(x)f)(r, \theta)\ a(\theta, r)$$

Masci, Boscaini, Bronstein, Vandergheynst 2015

Geodesic convolution

- Geodesic convolution = apply filter a to patches extracted from $f \in L^2(X)$ in local geodesic polar coordinates

$$(f \star a)(x) = \sum_{\theta, r} \underbrace{(D(x)f)(r,\theta)}\ \underbrace{a(\theta, r)}$$

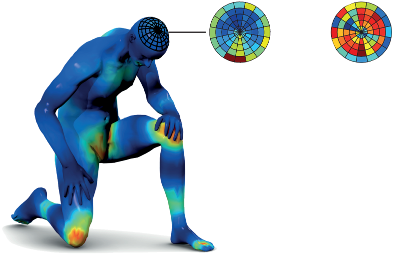


Masci, Boscaini, Bronstein, Vandergheynst 2015

Geodesic convolution

- Geodesic convolution = apply filter a to patches extracted from $f \in L^2(X)$ in local geodesic polar coordinates

$$(f \star a)(x) = \sum_{\theta,r} \underbrace{(D(x)f)(r,\theta)}\ \underbrace{a(\theta+\Delta\theta, r)}$$



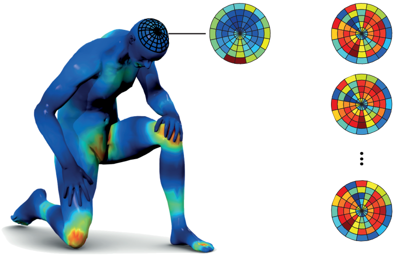
- Angular coordinate origin is arbitrary = rotation ambiguity!

Masci, Boscaini, Bronstein, Vandergheynst 2015

Geodesic convolution

- Geodesic convolution = apply filter a to patches extracted from $f \in L^2(X)$ in local geodesic polar coordinates

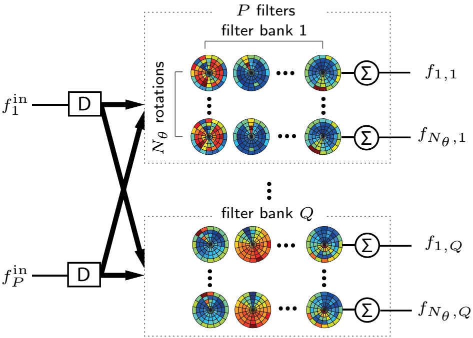
$$(f \star a)(x) = \sum_{\theta,r} \underbrace{(D(x)f)(r,\theta)}\ \underbrace{a(\theta+\Delta\theta, r)}$$



- Angular coordinate origin is arbitrary = rotation ambiguity!
- Keep all possible rotations

Masci, Boscaini, Bronstein, Vandergheynst 2015

Geodesic Convolution layer

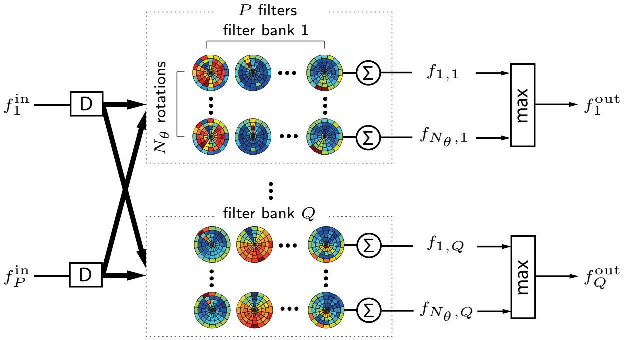


$$f^{\text{out}}_{\Delta\theta,q}(x) \quad = \quad \sum_{p=1}^{P} (f_p \star a_{\Delta\theta,qp})(x), \qquad q = 1,\ldots,Q$$

- $a_{\Delta\theta,qp}(\theta, r) = a_{qp}(\theta + \Delta\theta, r)$ are coefficients of pth filter in qth filter bank rotated by $\Delta\theta$

Masci, Boscaini, Bronstein, Vandergheynst 2015

Geodesic Convolution layer

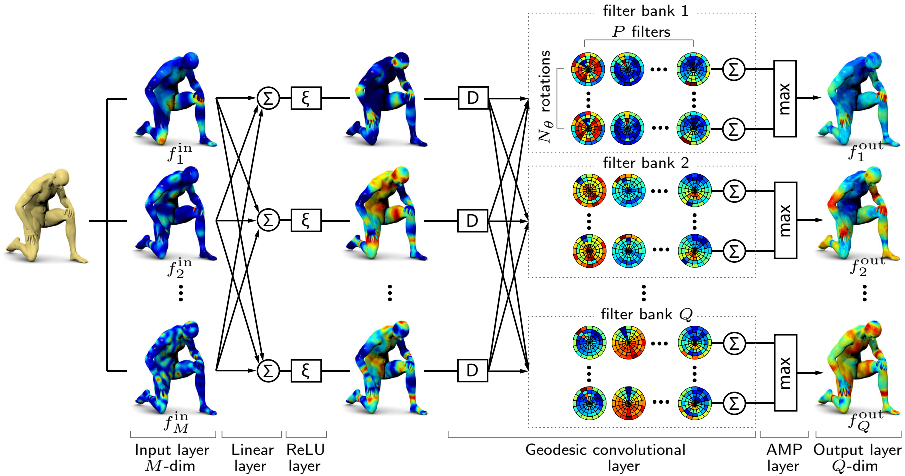


$$f_q^{\text{out}}(x) = \max_{\Delta\theta} \sum_{p=1}^{P} (f_p \star a_{\Delta\theta,qp})(x), \qquad q = 1, \ldots, Q$$

- $a_{\Delta\theta,qp}(\theta, r) = a_{qp}(\theta + \Delta\theta, r)$ are coefficients of pth filter in qth filter bank rotated by $\Delta\theta$
- Angular max pooling to remove rotation ambiguity

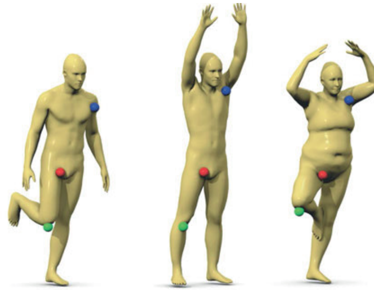
Masci, Boscaini, Bronstein, Vandergheynst 2015

Toy ShapeNet architecture



Masci, Boscaini, Bronstein, Vandergheynst 2015

Learning local descriptors with ShapeNet



- As similar as possible on positives $\mathcal{T}^+$
- As dissimilar as possible on negatives $\mathcal{T}^-$
- Minimize siamese loss w.r.t. ShapeNet parameters $\mathbf{\Theta}$

$$\begin{aligned} \ell(\mathbf{\Theta}) &= (1-\gamma) \sum_{(x,x^+)\in\mathcal{T}^+} \|\mathbf{f}_{\mathbf{\Theta}}(x) - \mathbf{f}_{\mathbf{\Theta}}(x^+)\| \\ &+ \gamma \sum_{(x,x^-)\in\mathcal{T}^-} \max\{\mu - \|\mathbf{f}_{\mathbf{\Theta}}(x) - \mathbf{f}_{\mathbf{\Theta}}(x^-)\|, 0\} \end{aligned}$$

Masci, Boscaini, Bronstein, Vandergheynst 2015

Descriptor robustness

HKS descriptor distance

Masci, Boscaini, Bronstein, Vandergheynst 2015; data: Bronstein et al. 2008 (TOSCA); Anguelov et al. 2005 (SCAPE); Bogo et al. 2014 (FAUST)

Descriptor robustness

WKS descriptor distance

Masci, Boscaini, Bronstein, Vandergheynst 2015; data: Bronstein et al. 2008 (TOSCA); Anguelov et al. 2005 (SCAPE); Bogo et al. 2014 (FAUST)

Descriptor robustness

Optimal Spectral descriptor distance

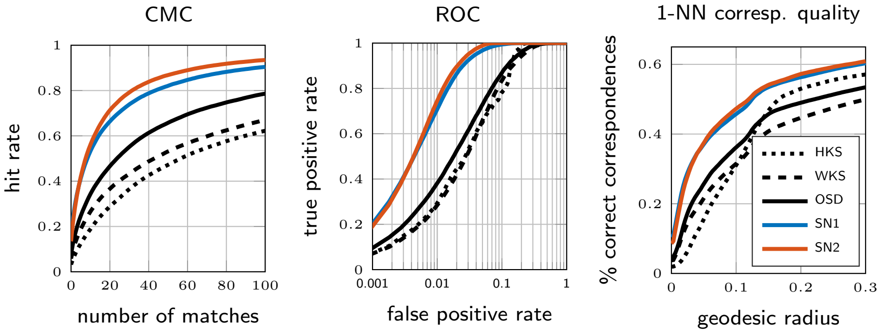
Masci, Boscaini, Bronstein, Vandergheynst 2015; data: Bronstein et al. 2008 (TOSCA); Anguelov et al. 2005 (SCAPE); Bogo et al. 2014 (FAUST)

Descriptor robustness

ShapeNet descriptor distance

Masci, Boscaini, Bronstein, Vandergheynst 2015; data: Bronstein et al. 2008 (TOSCA); Anguelov et al. 2005 (SCAPE); Bogo et al. 2014 (FAUST)

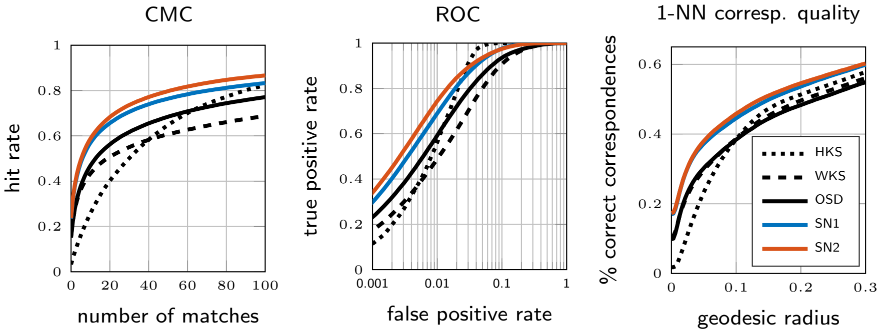
Descriptor performance



Descriptor performance using symmetric Princeton benchmark
(training and testing: FAUST)

Methods: Sun et al. 2009 (HKS); Aubry et al. 2011 (WKS); Litman, Bronstein 2014 (OSD); Masci, Boscaini, Bronstein, Vandergheynst 2015 (ShapeNet); data: Bogo et al. 2014 (FAUST); benchmark: Kim et al. 2011

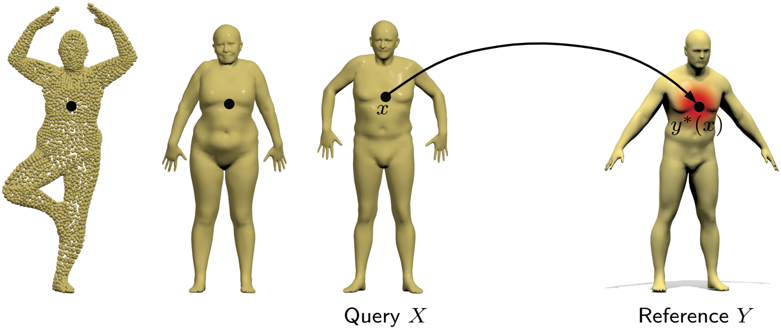
Descriptor performance



Descriptor performance using symmetric Princeton benchmark
(training: FAUST, testing: TOSCA)

Methods: Sun et al. 2009 (HKS); Aubry et al. 2011 (WKS); Litman, Bronstein 2014 (OSD); Masci, Boscaini, Bronstein, Vandergheynst 2015 (ShapeNet); data: Bogo et al. 2014 (FAUST); Bronstein et al. 2008 (TOSCA); benchmark: Kim et al. 2011

Learning shape correspondence with ShapeNet

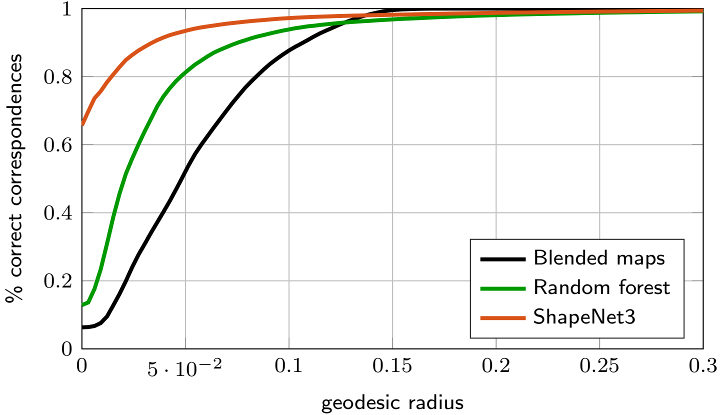


- Correspondence = labeling problem
- ShapeNet output $\mathbf{f}_{\mathbf{\Theta}}(x)$ = probability distribution on reference Y
- Minimize logistic regression cost w.r.t. ShapeNet parameters $\mathbf{\Theta}$

$$\ell(\mathbf{\Theta}) = -\sum_{(x,y^*(x))\in\mathcal{T}} \langle \delta_{y^*(x)}, \log \mathbf{f}_{\mathbf{\Theta}}(x) \rangle_{L^2(Y)}$$

Rodolà et al. 2014; Masci, Boscaini, Bronstein, Vandergheynst 2015

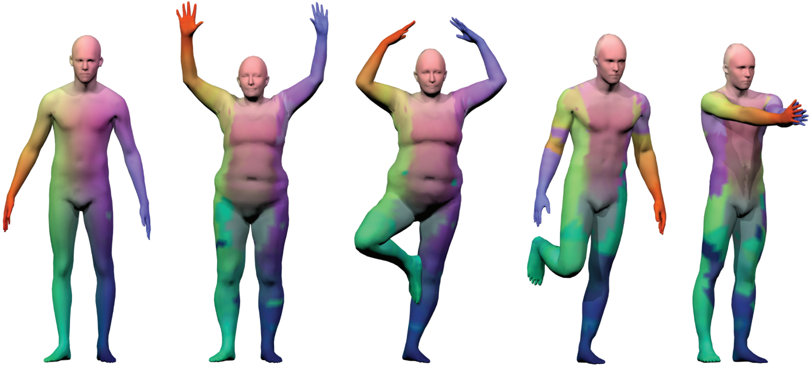
ShapeNet correspondence performance



Correspondence evaluated using symmetric Princeton benchmark
(training and testing: FAUST)

Masci, Boscaini, Bronstein, Vandergheynst 2015; Rodolà et al. 2014; Kim et al. 11

Correspondence examples: Random forest

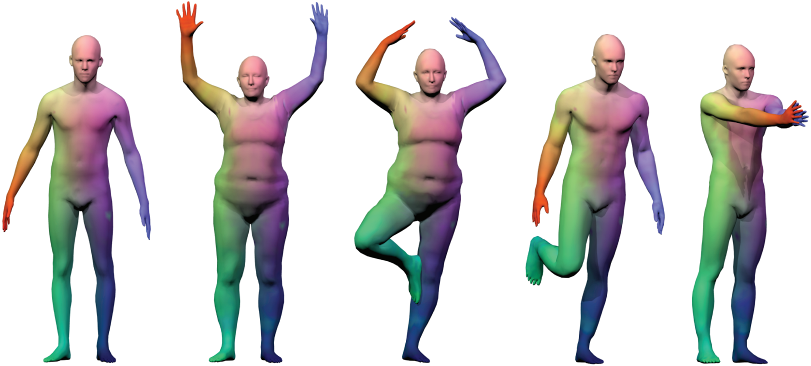


Correspondence found using random forest

(similar colors encode corresponding points)

Rodolà et al. 2014

Correspondence examples: ShapeNet

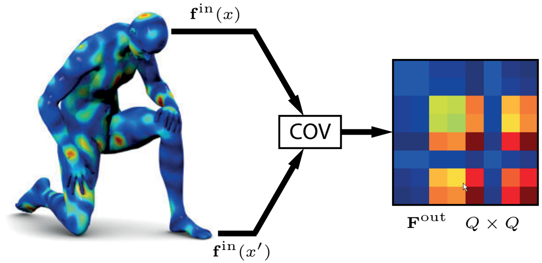


Correspondence found using ShapeNet

(similar colors encode corresponding points)

Masci, Boscaini, Bronstein, Vandergheynst 2015

From local to global features: covariance layer



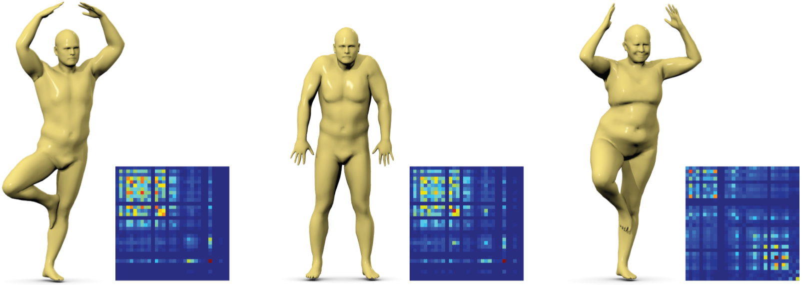
$$\mathbf{F}^{\text{out}} = \int_X (\mathbf{f}^{\text{in}}(x) - \boldsymbol{\mu}_{\mathbf{f}^{\text{in}}})(\mathbf{f}^{\text{in}}(x) - \boldsymbol{\mu}_{\mathbf{f}^{\text{in}}})^\top dx$$

$$\boldsymbol{\mu}_{\mathbf{f}^{\text{in}}} = \int_X \mathbf{f}_{\text{in}}(x) dx$$

- Aggregates local features into a global shape descriptor

Tuzel et al. 2006; Masci, Boscaini, Bronstein, Vandergheynst 2015

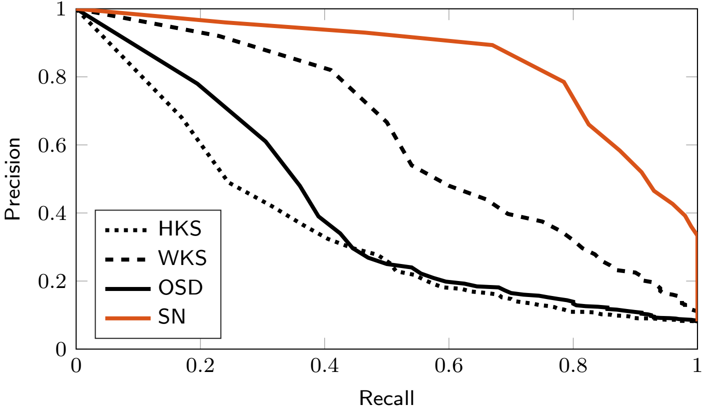
Learning shape similarity with ShapeNet



- Global shape descriptor using covariance layer in ShapeNet $\mathbf{F}_{\boldsymbol{\Theta}}$
- As similar as possible on positives $\mathcal{T}^{+}$
- As dissimilar as possible on negatives $\mathcal{T}^{-}$
- Minimize siamese loss w.r.t. ShapeNet parameters $\boldsymbol{\Theta}$

Rodolà et al. 2014; Masci, Boscaini, Bronstein, Vandergheynst 2015

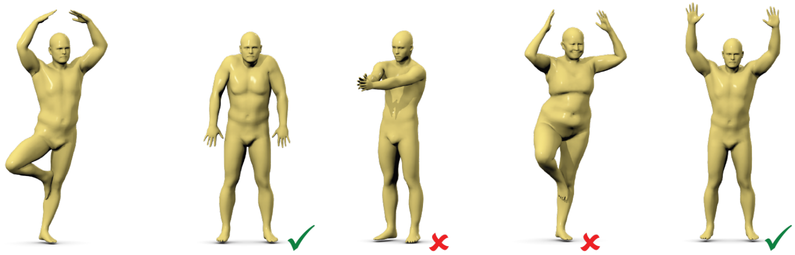
ShapeNet retrieval performance



1-layer ShapeNet; Training and testing: FAUST

Masci, Boscaini, Bronstein, Vandergheynst 2015

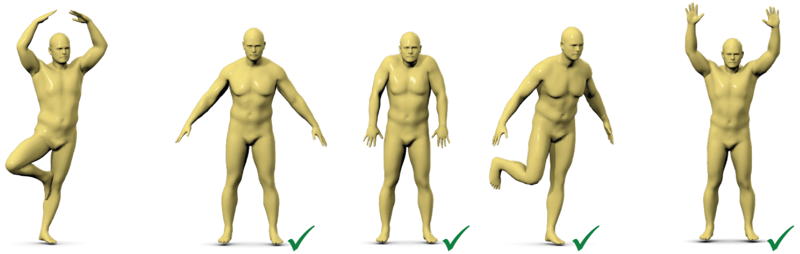
Retrieval examples: HKS



Shape retrieval using similarity computed with HKS

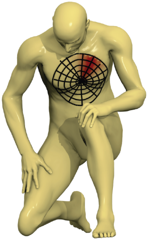
Masci, Boscaini, Bronstein, Vandergheynst 2015; data: Pickup et al. 2014

Retrieval examples: ShapeNet



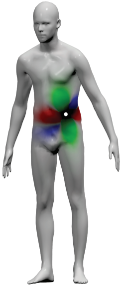
Shape retrieval using similarity computed with ShapeNet

Masci, Boscaini, Bronstein, Vandergheynst 2015; data: Pickup et al. 2014



Geodesic
convolution

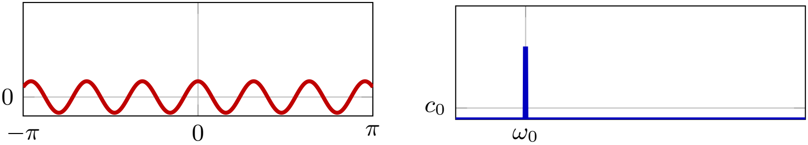
Windowed Fourier
transform



Anisotropic
diffusion

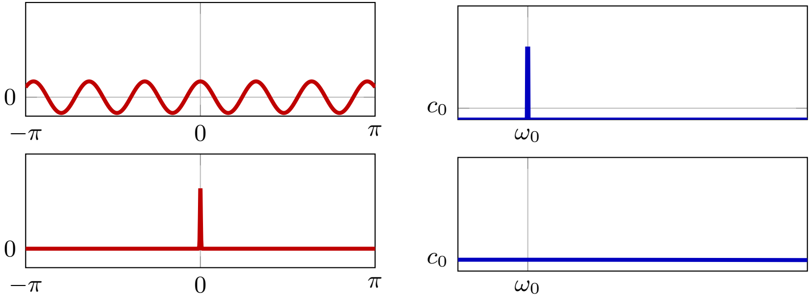
Uncertainty principle

Spatial localization $\times$ Frequency localization $= \text{const}$



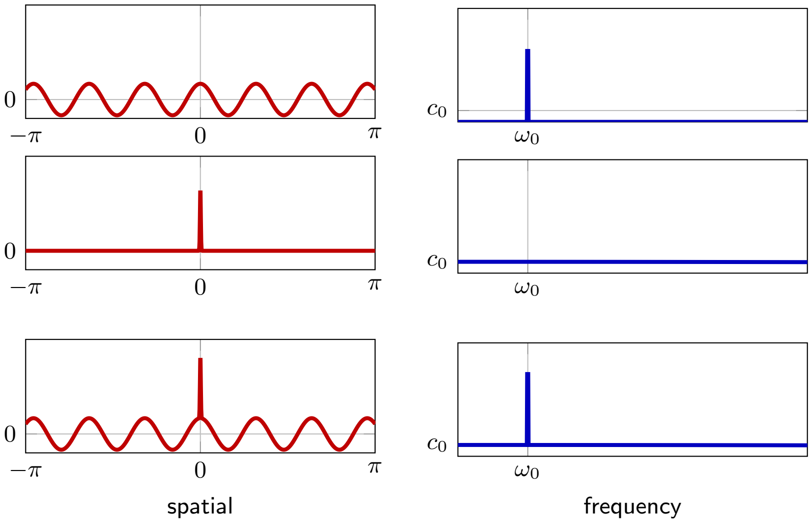
Uncertainty principle

Spatial localization $\times$ Frequency localization $= \mathrm{const}$



Uncertainty principle

Spatial localization $\times$ Frequency localization $= \mathrm{const}$



Windowed Fourier transform (WFT)

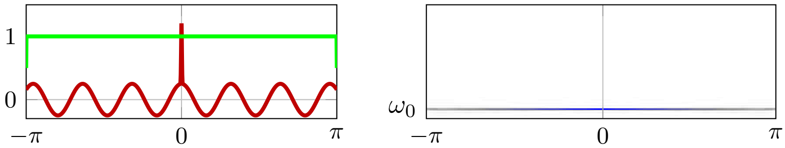
Localize Fourier transform in a window

$$\mathrm{WFT}(f(x))(\xi,\omega) = \frac{1}{2\pi}\int_{-\pi}^{\pi} f(x)\, w(x-\xi) e^{-i\omega x}\, dx$$

Windowed Fourier transform (WFT)

Localize Fourier transform in a window

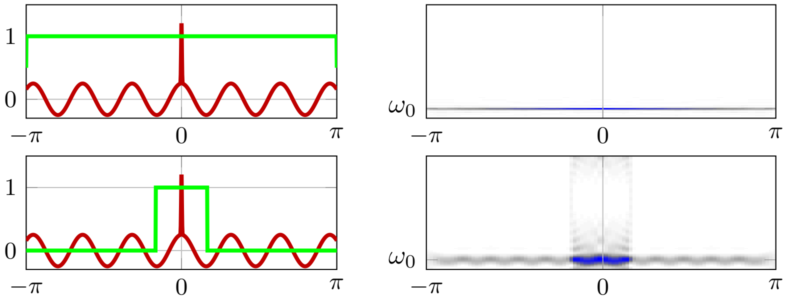
$$\mathrm{WFT}(f(x))(\xi,\omega) = \frac{1}{2\pi}\int_{-\pi}^{\pi} f(x)\, w(x-\xi) e^{-i\omega x}\, dx$$



Windowed Fourier transform (WFT)

Localize Fourier transform in a window

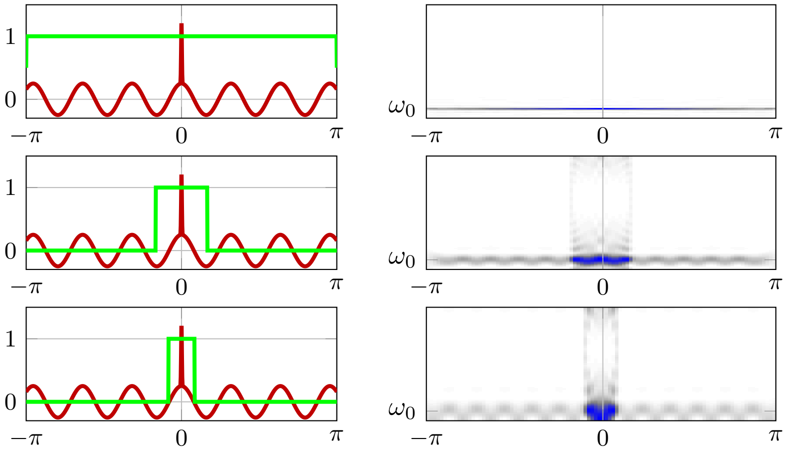
$$\mathrm{WFT}(f(x))(\xi,\omega) = \frac{1}{2\pi}\int_{-\pi}^{\pi} f(x)\, w(x-\xi) e^{-i\omega x}\, dx$$



Windowed Fourier transform (WFT)

Localize Fourier transform in a window

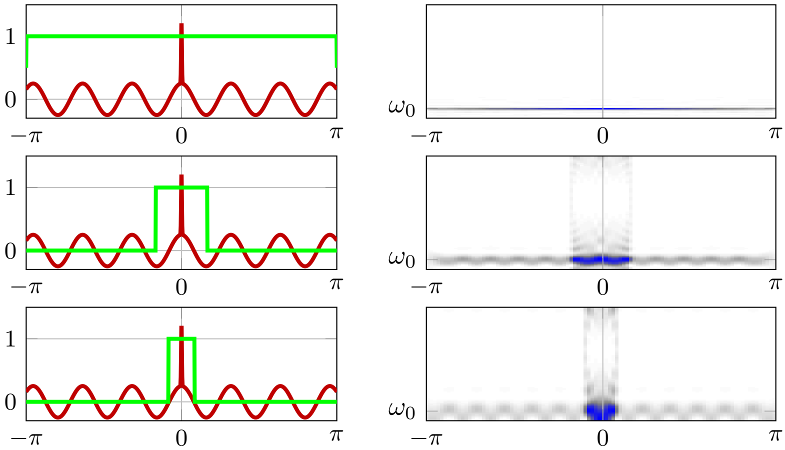
$$\mathrm{WFT}(f(x))(\xi,\omega) = \frac{1}{2\pi}\int_{-\pi}^{\pi} f(x)\, w(x-\xi) e^{-i\omega x}\, dx$$



Windowed Fourier transform (WFT)

Localize Fourier transform in a window

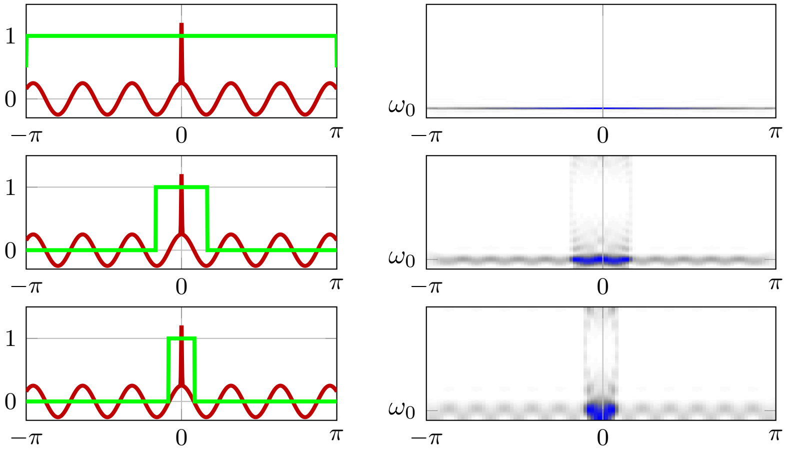
$$\mathrm{WFT}(f(x))(\xi,\omega) = \frac{1}{2\pi}\int_{-\pi}^{\pi} f(x)\underbrace{w(x-\xi)e^{-i\omega x}}_{\text{atom } g_{\xi,\omega}(x)}dx$$



Windowed Fourier transform (WFT)

Localize Fourier transform in a window

$$\mathrm{WFT}(f(x))(\xi,\omega) = \langle f, g_{\xi,\omega}\rangle_{L^2([-\pi,\pi])}$$



Windowed Fourier transform on manifolds

Translation: convolution with delta

$$(T_{x'}f)(x) \quad = \quad (f \star \delta_{x'})(x)$$

Shuman et al. 2014

Windowed Fourier transform on manifolds

Translation: convolution with delta

$$(T_{x'} f)(x) \quad = \quad (f \star \delta_{x'})(x) = \sum_{k \geq 1} \langle f, \phi_k \rangle_{L^2(X)} \langle \delta_{x'}, \phi_k \rangle_{L^2(X)} \phi_k(x)$$

Shuman et al. 2014

Windowed Fourier transform on manifolds

Translation: convolution with delta

$$\begin{aligned}(T_{x'}f)(x) &= (f \star \delta_{x'})(x) = \sum_{k\geq 1} \langle f, \phi_k \rangle_{L^2(X)} \langle \delta_{x'}, \phi_k \rangle_{L^2(X)} \phi_k(x) \\ &= \sum_{k \geq 1} \hat{f}_k \phi_k(x') \phi_k(x)\end{aligned}$$

Shuman et al. 2014

Windowed Fourier transform on manifolds

Translation: convolution with delta

$$\begin{aligned}(T_{x'}f)(x) &= (f\star\delta_{x'})(x) = \sum_{k\geq 1}\langle f,\phi_k\rangle_{L^2(X)}\langle\delta_{x'},\phi_k\rangle_{L^2(X)}\phi_k(x)\\ &= \sum_{k\geq 1}\hat{f}_k\phi_k(x')\phi_k(x)\end{aligned}$$

Modulation: multiplication by basis function

$$(M_k f)(x) = f(x)\phi_k(x)$$

Shuman et al. 2014

Windowed Fourier transform on manifolds

Translation: convolution with delta

$$\begin{aligned}(T_{x'}f)(x) &= (f \star \delta_{x'})(x) = \sum_{k\geq 1} \langle f, \phi_k \rangle_{L^2(X)} \langle \delta_{x'}, \phi_k \rangle_{L^2(X)} \phi_k(x) \\ &= \sum_{k\geq 1} \hat{f}_k \phi_k(x') \phi_k(x)\end{aligned}$$

Modulation: multiplication by basis function

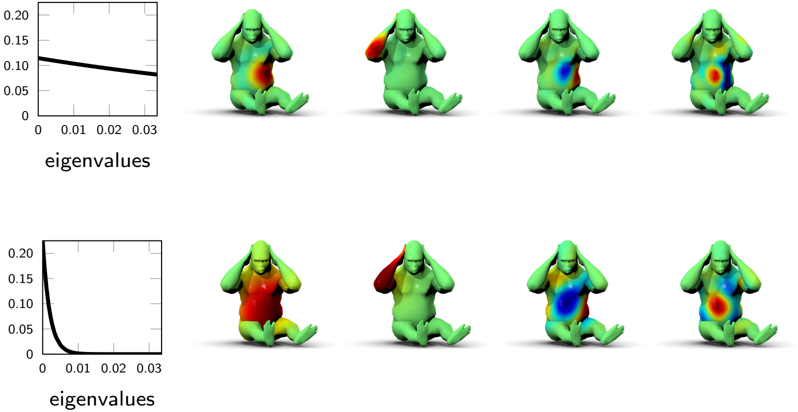
$$(M_k f)(x) = f(x)\phi_k(x)$$

Windowed Fourier transform:

$$(Sf)_{x,k} = \langle f, M_k T_x g \rangle_{L^2(X)} = \sum_{l\geq 1} \hat{g}_l \phi_l(x) \langle f, \phi_l \phi_k \rangle_{L^2(X)}$$

Shuman et al. 2014

Windowed Fourier transform on manifolds



Examples of WFT atoms $g_{x,k}$ for different windows $\hat{g}$

$$g_{x',k}(x) \quad = \quad (M_k T_{x'} g)(x) = \phi_k(x) \sum_{l \geq 1} \hat{g}_l \phi_l(x') \phi_l(x)$$

Boscaini, Masci, Melzi, Bronstein, Castellani, Vandergheynst 2015

Learning WFT windows

$$f_q^{\text{out}}(x) = \sum_{p=1}^{P}\sum_{k=1}^{K} a_{qpk}|(Sf_p^{\text{in}})_{x,k}|, \quad q = 1,\ldots,Q$$

where for each input dimension WFT uses a different window

$$(Sf_p^{\text{in}})_{x,k} = \sum_{l\geq 1} \underbrace{\gamma_p(\lambda_l)}_{\sum_{m=1}^{M} b_{pm}\beta_m(\lambda)} \phi_l(x)\langle f_p^{\text{in}}, \phi_l\phi_k\rangle_{L^2(X)}, \qquad p = 1,\ldots,P$$

- Learn window for each input dimension (coefficients b_{pm})

Boscaini, Masci, Melzi, Bronstein, Castellani, Vandergheynst 2015

Learning WFT windows

$$f_q^{\text{out}}(x) = \sum_{p=1}^{P}\sum_{k=1}^{K} a_{qpk}|(Sf_p^{\text{in}})_{x,k}|, \quad q = 1,\ldots,Q$$

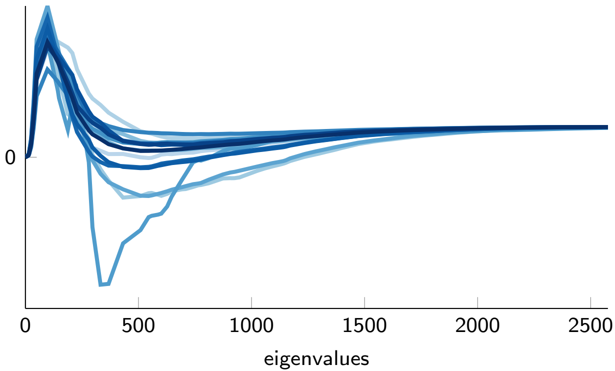
where for each input dimension WFT uses a different window

$$(Sf_p^{\text{in}})_{x,k} = \sum_{l\geq 1} \underbrace{\gamma_p(\lambda_l)}_{\sum_{m=1}^{M} b_{pm}\beta_m(\lambda)} \phi_l(x)\langle f_p^{\text{in}}, \phi_l\phi_k\rangle_{L^2(X)}, \quad p = 1,\ldots,P$$

- Learn window for each input dimension (coefficients b_{pm})
- Learn bank of filters for each WFT (coefficients a_{qpk})

Boscaini, Masci, Melzi, Bronstein, Castellani, Vandergheynst 2015

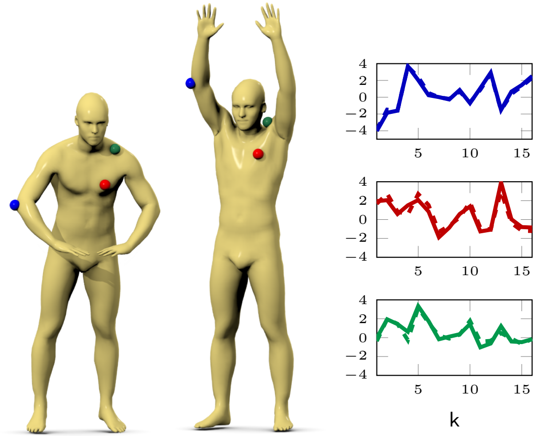
Example of learned WFT windows



WFT windows $\gamma_p(\lambda)$ learned on FAUST dataset

Boscaini, Masci, Melzi, Bronstein, Castellani, Vandergheynst 2015

Localized Spectral CNN (LSCNN) descriptors



Example of learned LSCNN shape descriptors

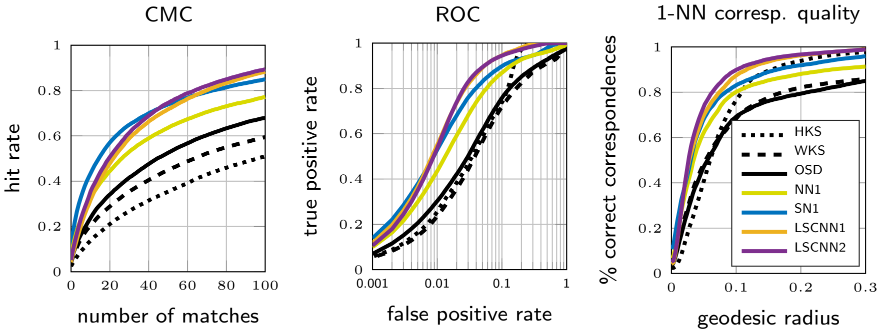
Boscaini, Masci, Melzi, Bronstein, Castellani, Vandergheynst 2015

LSCNN descriptor robustness: point clouds

LSCNN descriptor distance

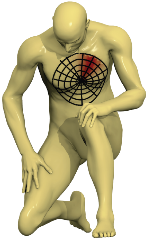
Boscaini, Masci, Melzi, Bronstein, Castellani, Vandergheynst 2015

LSCNN descriptor performance



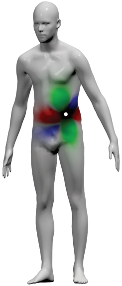
Descriptor performance using symmetric Princeton benchmark
(training and testing: FAUST)

Methods: Sun et al. 2009 (HKS); Aubry et al. 2011 (WKS); Litman, Bronstein 2014 (OSD); Masci, Boscaini, Bronstein, Vandergheynst 2015 (ShapeNet); Boscaini, Masci, Melzi, Bronstein, Castellani, Vandergheynst 2015 (NN, LSCNN); data: Bogo et al. 2014 (FAUST); benchmark: Kim et al. 2011



Geodesic convolution

Windowed Fourier transform



Anisotropic diffusion

Homogeneous diffusion

$$f_t(x) = -\operatorname{div}(c\nabla f(x))$$

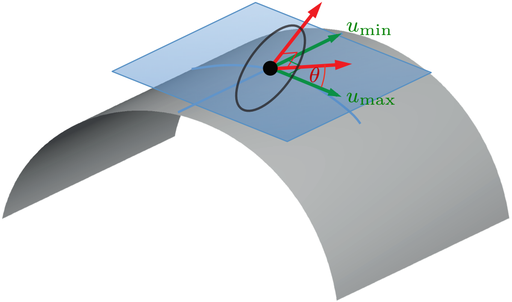
$c =$ thermal diffusivity constant describing heat conduction properties of the material (diffusion speed is equal everywhere)

Anisotropic diffusion

$$f_t(x) = -\mathrm{div}(A(x)\nabla f(x))$$

$A(x) =$ heat conductivity tensor describing heat conduction properties of the material (diffusion speed is position + direction dependent)

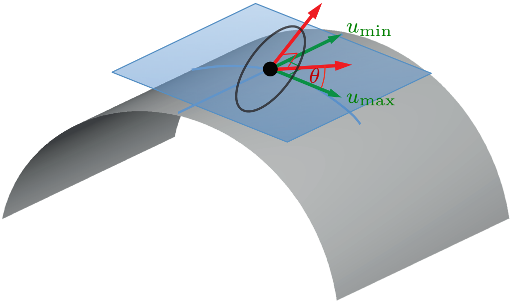
Anisotropic diffusion on manifolds



$$f_t(x) = -\mathrm{div}_X \left(R_\theta \begin{pmatrix} \alpha & \\ & 1 \end{pmatrix} R_\theta^\top \nabla_X f(x) \right)$$

Andreux et al. 2014; Boscaini, Masci, Rodolà, Bronstein, Cremers 2015

Anisotropic diffusion on manifolds



$$f_t(x) = -\mathrm{div}_X \Big(\underbrace{R_\theta \begin{pmatrix} \alpha & \\ & 1 \end{pmatrix} R_\theta^\top}_{D_{\alpha\theta}(x)} \nabla_X f(x) \Big)$$

- Anisotropic Laplacian $\Delta_{\alpha\theta} f(x) = \mathrm{div}_X \left(D_{\alpha\theta}(x) \nabla_X f(x) \right)$
- $\theta =$ orientation w.r.t. max curvature direction
- $\alpha =$ 'elongation'

Andreux et al. 2014; Boscaini, Masci, Rodolà, Bronstein, Cremers 2015

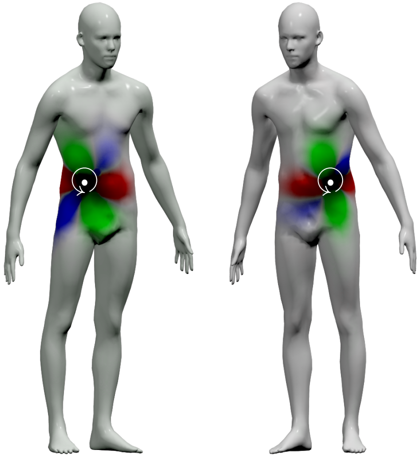
Anisotropic heat kernels

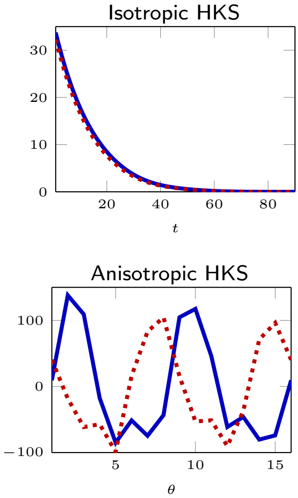
$$h_{\alpha\theta t}(x,\xi) = \sum_{k\geq 1} e^{-t\lambda_{\alpha\theta k}} \phi_{\alpha\theta k}(x)\phi_{\alpha\theta k}(\xi)$$

Examples of anisotropic heat kernels $h_{\alpha\theta t}$ for different values of t, θ and α

Boscaini, Masci, Rodolà, Bronstein, Cremers 2015

Isotropic vs Anisotropic HKS





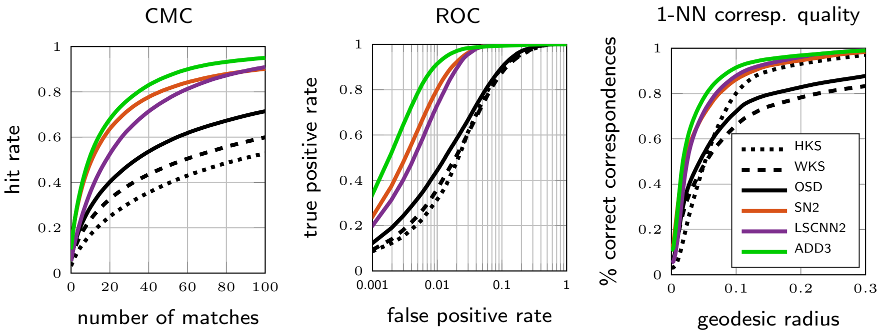
Boscaini, Masci, Rodolà, Bronstein, Cremers 2015

Anisotropic Diffusion Descriptor (ADD) robustness

Anisotropic Diffusion Descriptor (ADD) distance

Boscaini, Masci, Rodolà, Bronstein, Cremers 2015

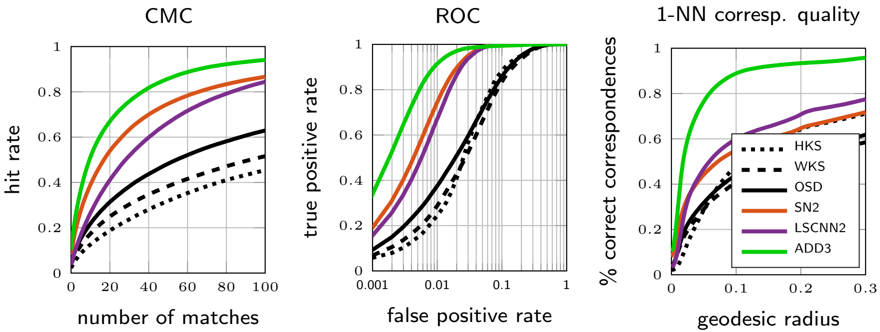
Anisotropic Diffusion Descriptor (ADD) performance



Descriptor performance using symmetric Princeton benchmark
(training and testing: FAUST)

Methods: Sun et al. 2009 (HKS); Aubry et al. 2011 (WKS); Litman, Bronstein 2014 (OSD); Masci, Boscaini, Bronstein, Vandergheynst 2015 (ShapeNet); Boscaini, Masci, Melzi, Bronstein, Castellani, Vandergheynst 2015 (LSCNN); Boscaini, Masci, Rodolà, Bronstein, Cremers 2015 (ADD); data: Bogo et al. 2014 (FAUST); benchmark: Kim et al. 2011

Anisotropic Diffusion Descriptor (ADD) performance



Descriptor performance using asymmetric Princeton benchmark
(training and testing: FAUST)

Methods: Sun et al. 2009 (HKS); Aubry et al. 2011 (WKS); Litman, Bronstein 2014 (OSD); Masci, Boscaini, Bronstein, Vandergheynst 2015 (ShapeNet); Boscaini, Masci, Melzi, Bronstein, Castellani, Vandergheynst 2015 (LSCNN); Boscaini, Masci, Rodolà, Bronstein, Cremers 2015 (ADD); data: Bogo et al. 2014 (FAUST); benchmark: Kim et al. 2011

ADD correspondence example: meshes

Correspondence found using ADD

(similar colors encode corresponding points)

Boscaini, Masci, Rodolà, Bronstein, Cremers 2015

ADD correspondence example: point clouds

Correspondence found using ADD

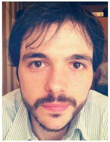
(similar colors encode corresponding points)

Boscaini, Masci, Rodolà, Bronstein, Cremers 2015

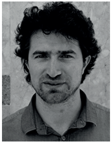
Summary

- First construction of generalizable intrinsic convolutional neural networks
- Learnable, task-specific, intrinsic features
- State-of-the-art performance in a variety of applications in 3D shape analysis
- Beyond shapes: graphs, social networks, etc.


D. Boscaini


J. Masci


P. Vandergheynst


U. Castellani


S. Melzi


E. Rodolà


D. Cremers

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

Funded by

Thank you!