Learning on manifolds and graphs with intrinsic CNNs

Michael Bronstein

University of Lugano
Switzerland

Tel Aviv University
Israel

Intel Corporation
Israel

3DDL NIPS Workshop, Barcelona, 9 December 2016


VIVID 9i
VIVID 9i

$100K

2005

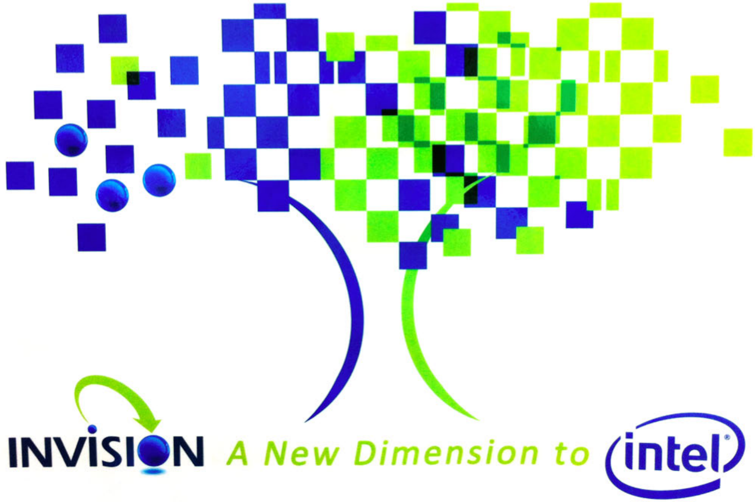
$100

2010

$20

2014


intel
REALSENSE
TECHNOLOGY



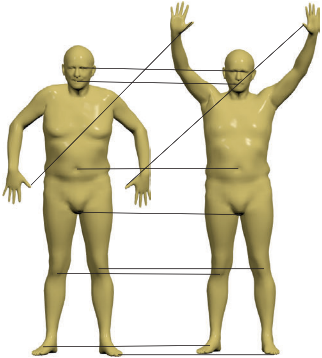
(Acquired by Intel in 2012)

Applications

Markerless motion capture

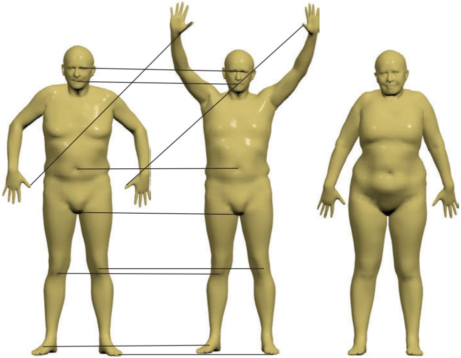
Gesture control

Basic problems: shape similarity and correspondence



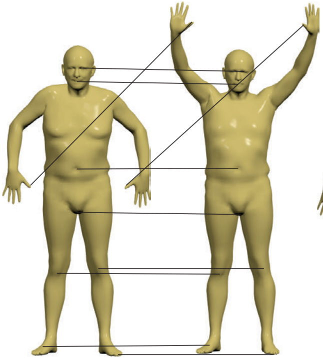
Isometric

Basic problems: shape similarity and correspondence



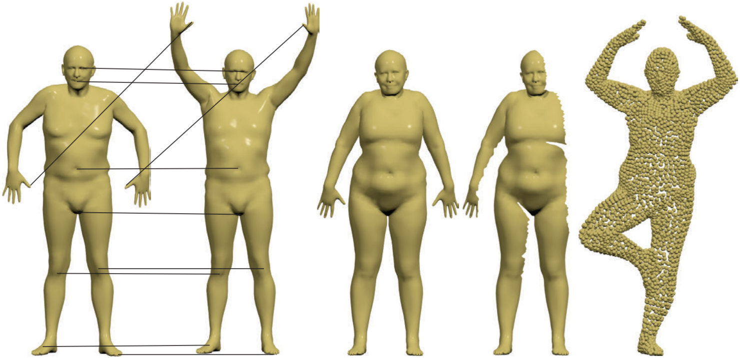
Isometric Non-isometric

Basic problems: shape similarity and correspondence



Isometric   Non-isometric   Partial

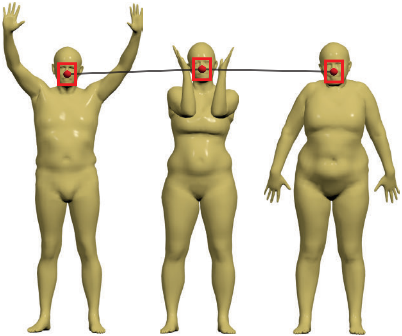
Basic problems: shape similarity and correspondence



Isometric Non-isometric Partial Different representation

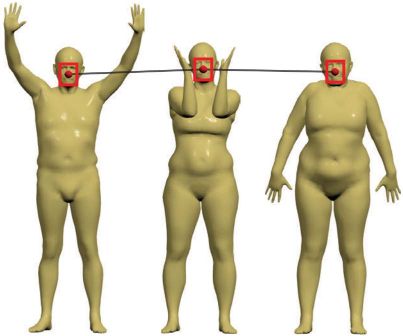
Task-specific features

Correspondence

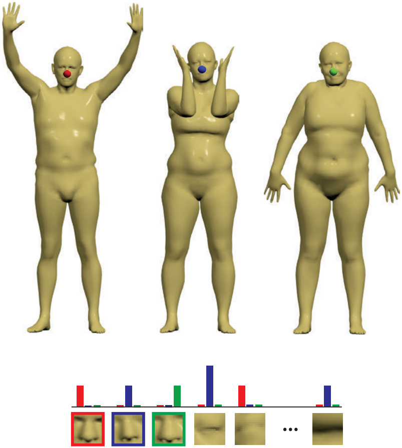


Task-specific features

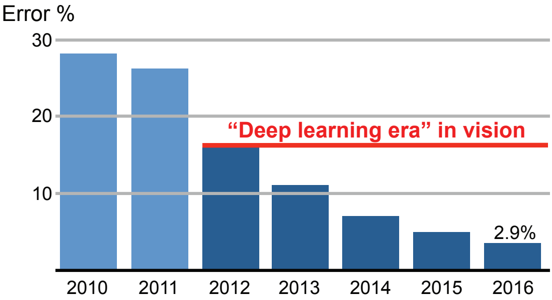
Correspondence



Similarity

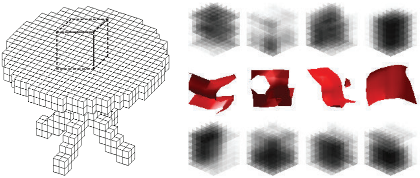


Deep learning in computer vision



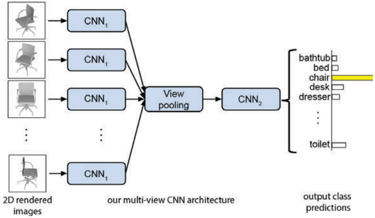
ImageNet ILSVRC Challenge

Deep learning in computer graphics



Volumetric[1]

Single view based[2]



Multiple view based[3]

[1]Wu et al. 2015; [2]Wei et al. 2016; [3]Su et al. 2015

Extrinsic vs Intrinsic CNNs

Extrinsic **Intrinsic**

What is convolution on manifolds?

Convolution

Euclidean

Spatial domain

$$(f \star g)(x) = \int_{-\pi}^{\pi} f(\xi) g(x - \xi) d\xi$$

Non-Euclidean

Convolution

Euclidean

Spatial domain

$$(f \star g)(x) = \int_{-\pi}^{\pi} f(\xi) g(x - \xi) d\xi$$

Spectral domain

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

'Convolution Theorem'

Non-Euclidean

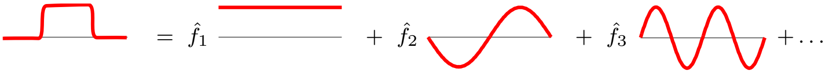
Convolution

Euclidean	**Non-Euclidean**
Spatial domain	
$$(f \star g)(x) = \int_{-\pi}^{\pi} f(\xi) g(x - \xi) d\xi$$	?
Spectral domain	
$$\widehat{(f \star g)}(\omega) = \hat{f}(\omega) \cdot \hat{g}(\omega)$$	?
'Convolution Theorem'	

Fourier analysis (Euclidean spaces)

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

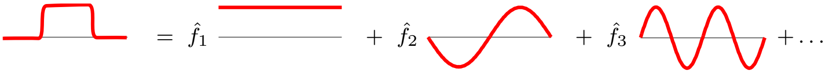
$$f(x) = \sum_{\omega} \frac{1}{2\pi} \int_{-\pi}^{\pi} f(\xi) e^{-ik\xi} d\xi \, e^{ikx}$$



Fourier analysis (Euclidean spaces)

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

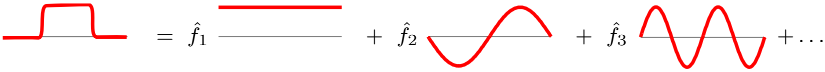
$$f(x) = \sum_{\omega} \underbrace{\frac{1}{2\pi} \int_{-\pi}^{\pi} f(\xi) e^{-ik\xi} d\xi}_{\hat{f}_k = \langle f, e^{ikx} \rangle_{L^2([-\pi,\pi])}} e^{ikx}$$



Fourier analysis (Euclidean spaces)

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

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



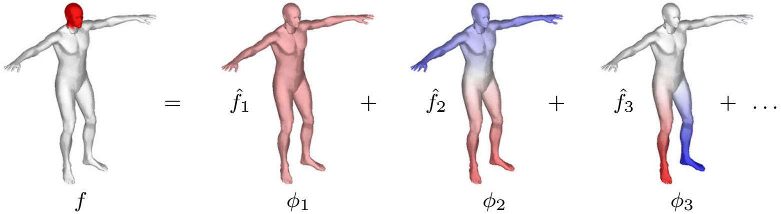
Fourier basis = Laplacian eigenfunctions: $\Delta e^{ikx} = k^2 e^{ikx}$

We define Laplacian as a positive semi-definite operator $\Delta = -\frac{d^2}{dx^2}$

Fourier analysis (non-Euclidean spaces)

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

$$f(x) = \sum_{k \geq 0} \underbrace{\int_{\mathcal{X}} f(\xi)\phi_k(\xi) d\xi}_{\hat{f}_k = \langle f, \phi_k \rangle_{L^2(\mathcal{X})}} \ \phi_k(x)$$



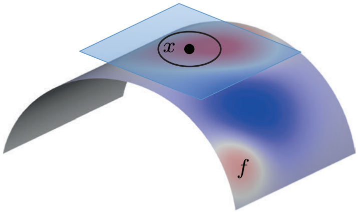
Fourier basis = Laplacian eigenfunctions: $\Delta \phi_k(x) = \lambda_k \phi_k(x)$

Laplacian operator

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

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

"difference between $f(x)$ and average value of f around x"

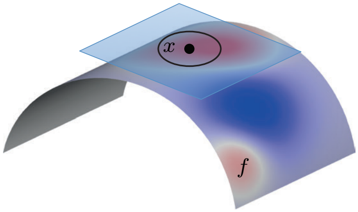


Laplacian operator

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

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

"difference between $f(x)$ and average value of f around x"



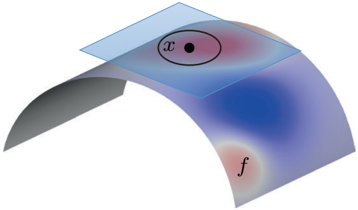
- Intrinsic (expressed solely in terms of the Riemannian metric)

Laplacian operator

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

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

"difference between $f(x)$ and average value of f around x"



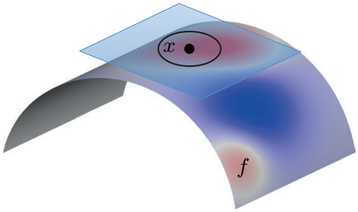
- Intrinsic (expressed solely in terms of the Riemannian metric)
- Isometry-invariant

Laplacian operator

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

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

"difference between $f(x)$ and average value of f around x"



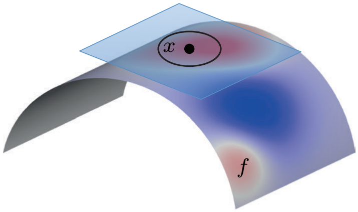
- Intrinsic (expressed solely in terms of the Riemannian metric)
- Isometry-invariant
- Self-adjoint $\langle \Delta f, g \rangle_{L^2(\mathcal{X})} = \langle f, \Delta g \rangle_{L^2(\mathcal{X})}$

Laplacian operator

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

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

"difference between $f(x)$ and average value of f around x"



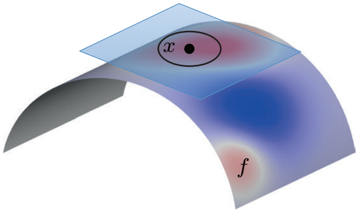
- Intrinsic (expressed solely in terms of the Riemannian metric)
- Isometry-invariant
- Self-adjoint $\langle \Delta f, g \rangle_{L^2(\mathcal{X})} = \langle f, \Delta g \rangle_{L^2(\mathcal{X})}$
  $\Rightarrow$ orthogonal eigenfunctions

Laplacian operator

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

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

"difference between $f(x)$ and average value of f around x"



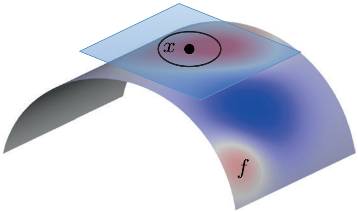
- Intrinsic (expressed solely in terms of the Riemannian metric)
- Isometry-invariant
- Self-adjoint $\langle \Delta f, g \rangle_{L^2(\mathcal{X})} = \langle f, \Delta g \rangle_{L^2(\mathcal{X})}$
  $\Rightarrow$ orthogonal eigenfunctions
- Positive semidefinite

Laplacian operator

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

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

"difference between $f(x)$ and average value of f around x"



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

Convolution

Euclidean

Spatial domain

$$(f \star g)(x) = \int_{-\pi}^{\pi} f(\xi) g(x - \xi) d\xi$$

Spectral domain

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

'Convolution Theorem'

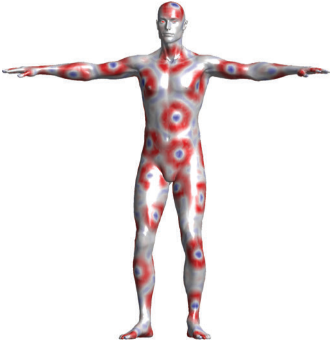
Non-Euclidean

?

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

Spectral convolution

Function f

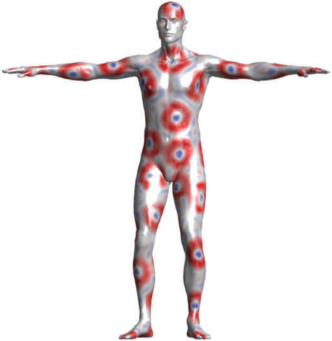


Filtered function $\tilde{f}$

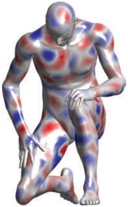
Henaff, Bruna, LeCun 2015; Defferrard, Bresson, Vandergheynst 2016

Spectral convolution

Function f



Filtered function $\tilde{f}$

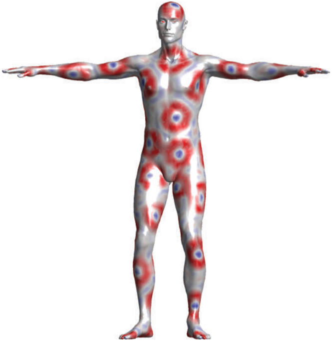


Same function,
same filter,
another shape

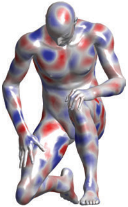
Henaff, Bruna, LeCun 2015; Defferrard, Bresson, Vandergheynst 2016

Spectral convolution

Function f



Filtered function $\tilde{f}$

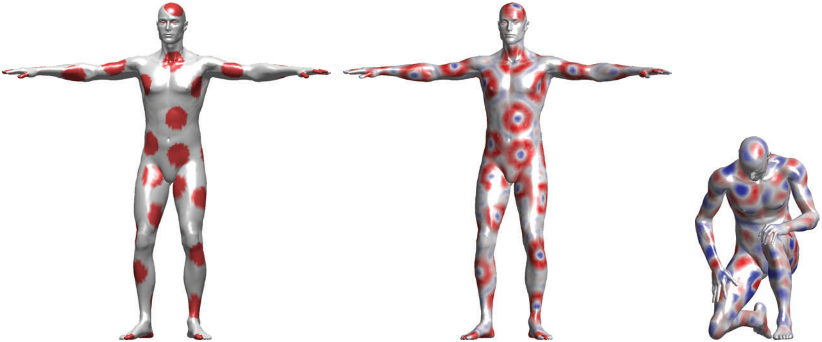


Same function,
same filter,
another shape

Filter is basis dependent

Henaff, Bruna, LeCun 2015; Defferrard, Bresson, Vandergheynst 2016

Spectral convolution

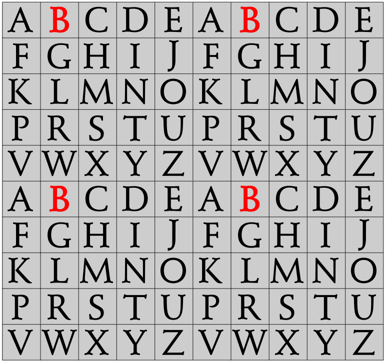


Function f | Filtered function $\tilde{f}$ | Same function, same filter, another shape

Filter is basis dependent $\Rightarrow$ **does not generalize across domains!**

Henaff, Bruna, LeCun 2015; Defferrard, Bresson, Vandergheynst 2016

Convolution in the spatial domain

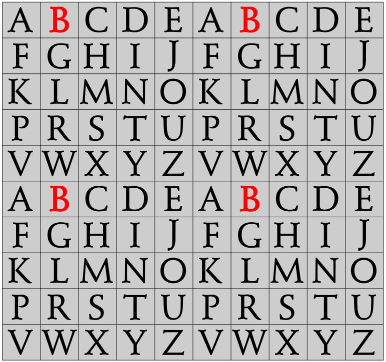


Euclidean

Non-Euclidean

- No canonical global system of coordinates

Convolution in the spatial domain

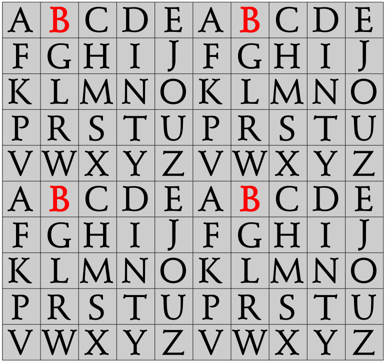


Euclidean

Non-Euclidean

- No canonical global system of coordinates
- No grid structure (no regular memory access)

Convolution in the spatial domain



Euclidean

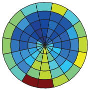
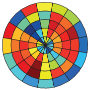
Non-Euclidean

- No canonical global system of coordinates
- No grid structure (no regular memory access)
- No shift-invariance (patch operator is position-dependent)

Convolution

Euclidean	**Non-Euclidean**
Spatial domain	
$$(f \star g)(x) = \int_{-\pi}^{\pi} f(\xi) g(x - \xi) d\xi$$	$$(f \star g)(x) = \int (D(x)f)(\mathbf{u}) g(\mathbf{u}) d\mathbf{u}$$
Spectral domain	
$$\widehat{(f \star g)}(\omega) = \hat{f}(\omega) \cdot \hat{g}(\omega)$$	$$\widehat{(f \star g)}_k = \langle f, \phi_k \rangle_{L^2(\mathcal{X})} \langle g, \phi_k \rangle_{L^2(\mathcal{X})}$$
'Convolution Theorem'	

Patch operator

$(f \star g)(x) = \int$  $\times$  $d\mathbf{u}$

$(D(x)f)(\mathbf{u})$ $\quad g(\mathbf{u})$

Masci, Boscaini, B, Vandergheynst 2015; Boscaini, Masci, Rodolà, B 2016

Heat diffusion on manifolds

$$f_t = -c\Delta f$$

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

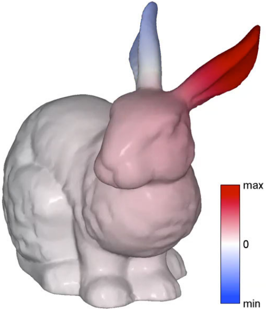
Heat diffusion on manifolds

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

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

Heat diffusion on manifolds

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

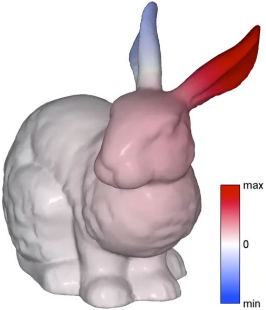


- $f(x,t)$ = amount of heat at point x at time t
- $f_0(x)$ = initial heat distribution

Solution of the heat equation expressed through the heat operator

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

Heat diffusion on manifolds



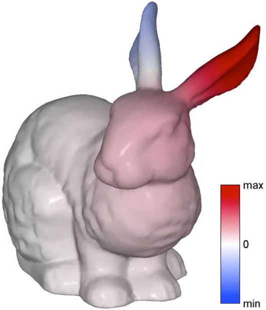
$$\begin{cases} f_t(x,t) = -\Delta f(x,t) \\ f(x,0) = f_0(x) \end{cases}$$

- $f(x,t)$ = 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

$$f(x,t) \quad = \quad e^{-t\Delta} f_0(x) = \sum_{k\geq 0} \langle f_0, \phi_k \rangle_{L^2(\mathcal{X})} e^{-t\lambda_k} \phi_k(x)$$

Heat diffusion on manifolds



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

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

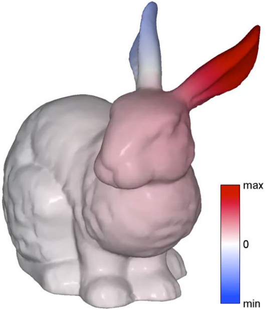
Solution of the heat equation expressed through the heat operator

$$\begin{aligned} f(x,t) &= e^{-t\Delta} f_0(x) = \sum_{k\geq 0} \langle f_0, \phi_k \rangle_{L^2(\mathcal{X})} e^{-t\lambda_k} \phi_k(x) \\ &= \int_{\mathcal{X}} f_0(\xi) \sum_{k\geq 0} e^{-t\lambda_k} \phi_k(x)\phi_k(\xi)\, d\xi \end{aligned}$$

Heat diffusion on manifolds

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

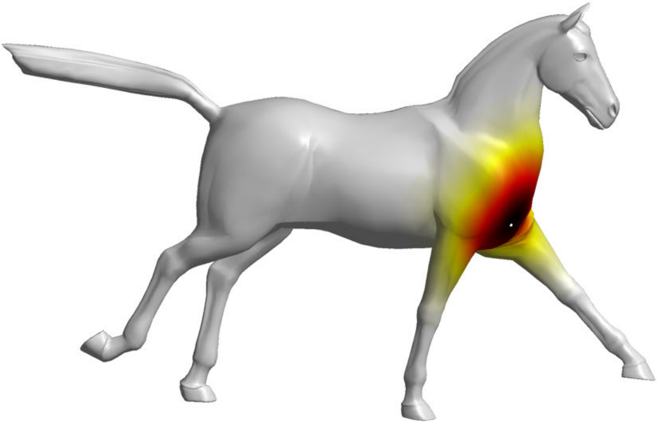
- $f(x,t)$ = amount of heat at point x at time t
- $f_0(x)$ = initial heat distribution



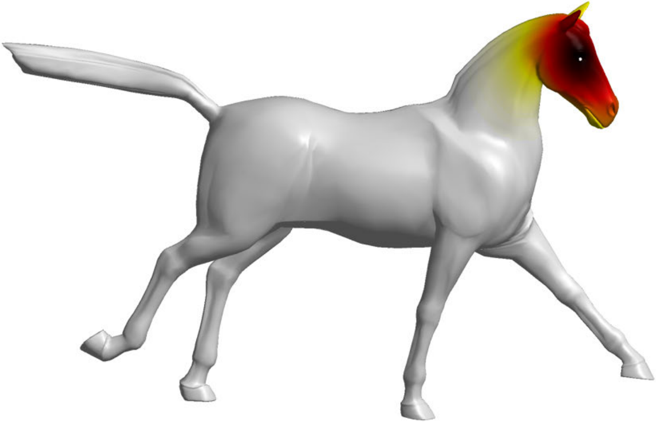
Solution of the heat equation expressed through the heat operator

$$\begin{aligned} f(x,t) &= e^{-t\Delta} f_0(x) = \sum_{k\geq 0} \langle f_0, \phi_k \rangle_{L^2(\mathcal{X})} e^{-t\lambda_k} \phi_k(x) \\ &= \int_{\mathcal{X}} f_0(\xi) \underbrace{\sum_{k\geq 0} e^{-t\lambda_k} \phi_k(x)\phi_k(\xi)}_{\text{heat kernel } h_t(x,\xi)} d\xi \end{aligned}$$

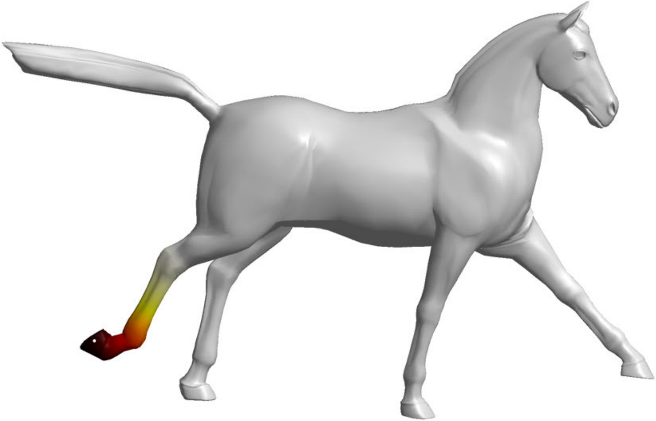
Heat kernels



Heat kernels



Heat kernels



Heat kernels

Homogeneous diffusion

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

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

Anisotropic diffusion

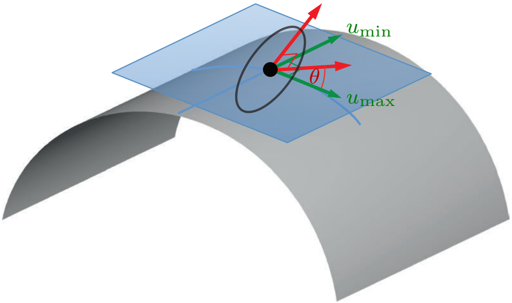
$$f_t(x) = -\mathrm{div}(A(x)\nabla f(x))$$

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

Anisotropic diffusion

Isotropic **Anisotropic**

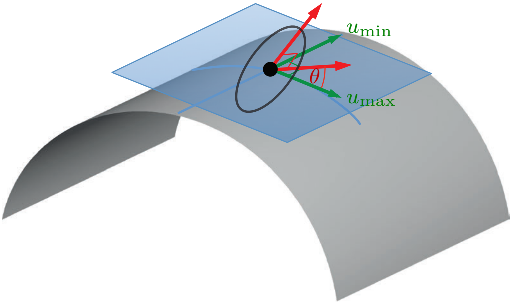
Anisotropic diffusion on manifolds



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

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

Anisotropic diffusion on manifolds



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

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

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

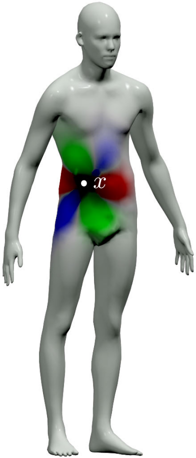
Anisotropic heat kernels

$$h_{\alpha\theta t}(x,\xi) = \sum_{k\geq 0} e^{-t\lambda_{\alpha\theta k}} \phi_{\alpha\theta k}(x)\phi_{\alpha\theta k}(\xi)$$

Scale t Orientation θ Elongation α

Boscaini, Masci, Rodolà, B, Cremers 2015

Intrinsic patch operator



Given a function $f \in L^2(\mathcal{X})$, the patch operator

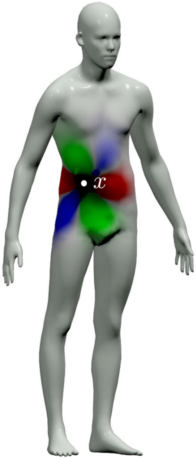
$$(D(x)f)(\theta, t) = \int_{\mathcal{X}} f(\xi) h_{\alpha\theta t}(x, \xi) d\xi$$

produces a local representation of f around point x

- θ = 'angular coordinate'
- t = 'radial coordinate'

Masci, Boscaini, B, Vandergheynst 2015; Boscaini, Masci, Rodolà, B 2016

Intrinsic patch operator



Given a function $f \in L^2(\mathcal{X})$, the patch operator

$$(D(x)f)(\theta, t) = \int_{\mathcal{X}} f(\xi) h_{\alpha\theta t}(x, \xi) d\xi$$

produces a local representation of f around point x

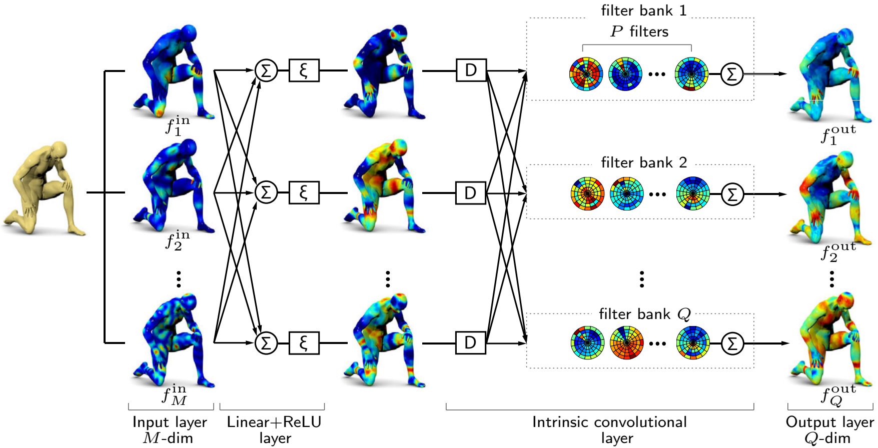
- $\theta =$ 'angular coordinate'
- $t =$ 'radial coordinate'

Intrinsic convolution

$$(f \star a)(x) = \sum_{\theta, t} (D(x)f)(\theta, t) g(\theta, t)$$

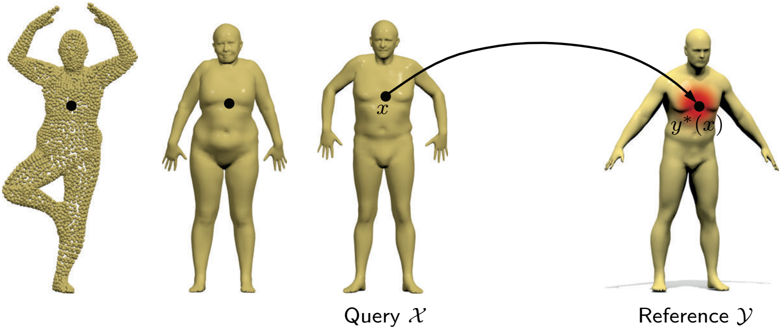
Masci, Boscaini, B, Vandergheynst 2015; Boscaini, Masci, Rodolà, B 2016

Toy Anisotropic CNN architecture



Masci, Boscaini, B, Vandergheynst 2015; Boscaini, Masci, Rodolà, B 2016

Learning shape correspondence

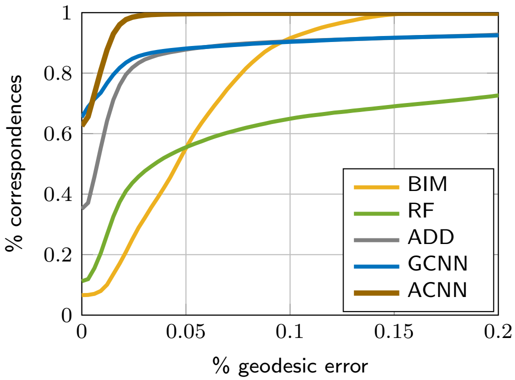


- Correspondence = labeling problem
- ACNN output $\mathbf{f}_{\mathbf{\Theta}}(x)$ = probability distribution on reference $\mathcal{Y}$
- Minimize logistic regression cost w.r.t. ACNN parameters $\mathbf{\Theta}$

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

Rodolà et al. 2014; Masci, Boscaini, B, Vandergheynst 2015; Boscaini, Masci, Rodolà, B 2016

Correspondence performance



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

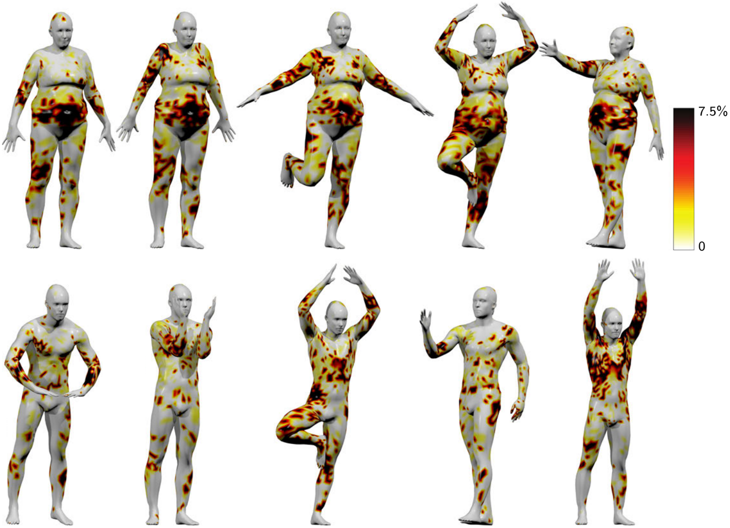
Methods: Kim et al. 2011 (BIM); Boscaini, Masci, Melzi, B, Castellani, Vandergheynst 2015 (LSCNN); Rodolà et al. 2014 (RF); Boscaini, Masci, Rodolà, B, Cremers 2015 (ADD); Masci, Boscaini, B, Vandergheynst 2015 (GCNN); Boscaini, Masci, Rodolà, B 2016 (ACNN); data: Bogo et al. 2014 (FAUST); benchmark: Kim et al. 2011

Correspondence error: Blended Intrinsic Map

Pointwise geodesic error (in % of geodesic diameter)

Kim, Lipman, Funkhouser 2011

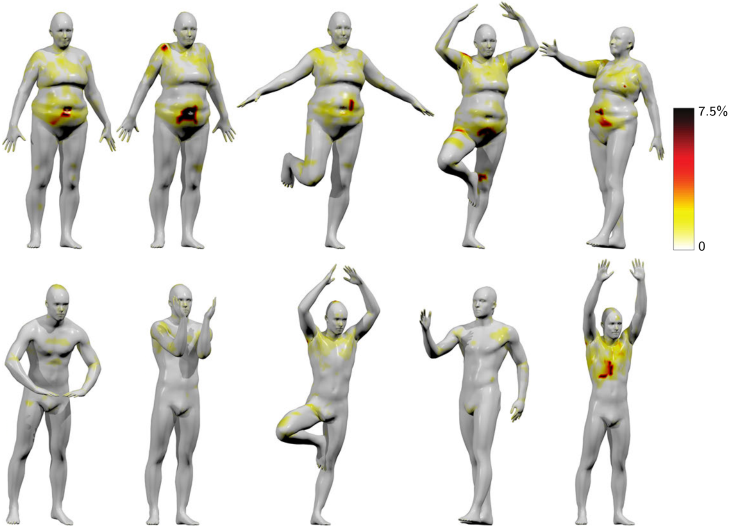
Correspondence error: GCNN



Pointwise geodesic error (in % of geodesic diameter)

Masci, Boscaini, B, Vandergheynst 2015

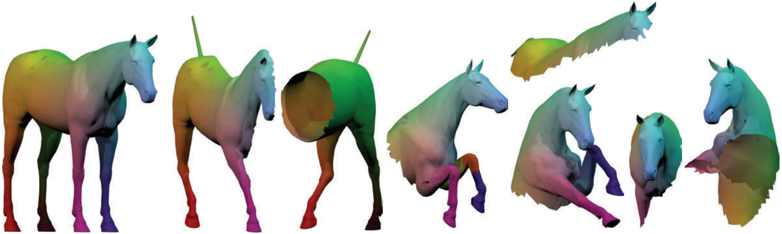
Correspondence error: ACNN



Pointwise geodesic error (in % of geodesic diameter)

Boscaini, Masci, Rodolà, Bronstein 2016

Partial correspondence with ACNN



Correspondence

Correspondence error

Boscaini, Masci, Rodolà, B 2016

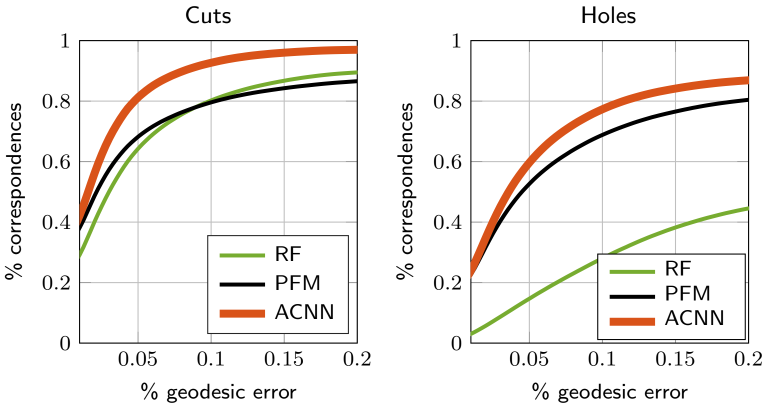
Partial correspondence with ACNN

Correspondence

Correspondence error

Boscaini, Masci, Rodolà, B 2016

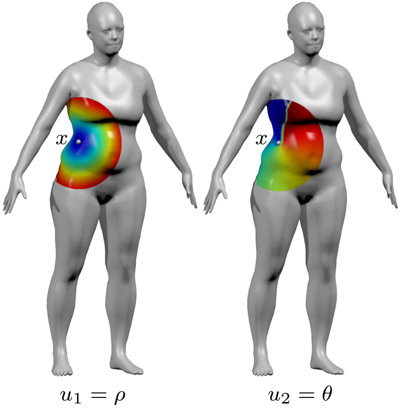
Partial correspondence performance



Methods: Rodolà et al. 2014 (RF); Rodolà et al. 2015 (PFM); Boscaini, Masci, Rodolà, B 2016 (ACNN); data: Cosmo et al. 2016 (SHREC); benchmark: Kim et al. 2011

Mixture Model Network (MoNet)

- Local geodesic coordinates
  $\mathbf{u}(x, y) = (\rho(x, y), \theta(x, y))$



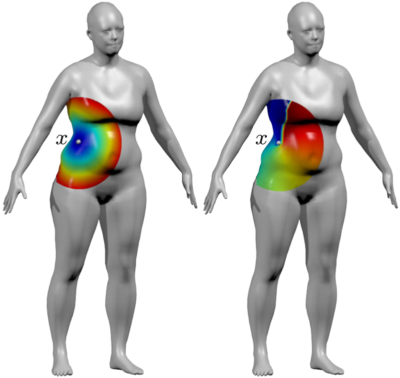
Monti, Boscaini, Masci, Rodolà, Svoboda, B 2016

Mixture Model Network (MoNet)

- Local geodesic coordinates
  $\mathbf{u}(x, y) = (\rho(x, y), \theta(x, y))$
- Gaussian weight functions

  $$w_k(\mathbf{u}) = \exp\left((\mathbf{u} - \boldsymbol{\mu}_k)^\top \boldsymbol{\Sigma}_k^{-1}(\mathbf{u} - \boldsymbol{\mu}_k)\right)$$

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



$u_1 = \rho$ $\qquad$ $u_2 = \theta$

Monti, Boscaini, Masci, Rodolà, Svoboda, B 2016

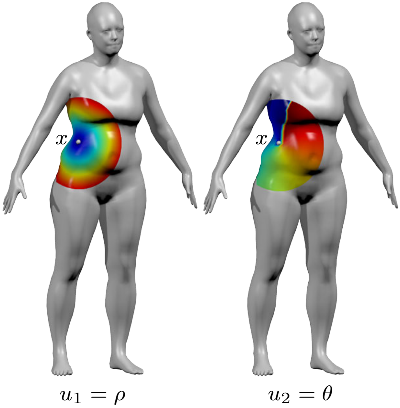
Mixture Model Network (MoNet)

- Local geodesic coordinates
  $\mathbf{u}(x,y) = (\rho(x,y), \theta(x,y))$
- Gaussian weight functions

  $$w_k(\mathbf{u}) = \exp\left((\mathbf{u}-\boldsymbol{\mu}_k)^\top \boldsymbol{\Sigma}_k^{-1}(\mathbf{u}-\boldsymbol{\mu}_k)\right)$$

  learnable covariance $\boldsymbol{\Sigma}$ and mean $\boldsymbol{\mu}$
- Patch operator

  $$(D(x)f)_k = \int_{\mathcal{X}} w_k(\mathbf{u}(x,y))f(y)dy$$



Monti, Boscaini, Masci, Rodolà, Svoboda, B 2016

Mixture Model Network (MoNet)

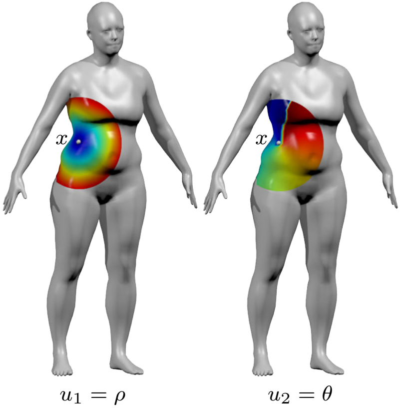
- Local geodesic coordinates
  $\mathbf{u}(x,y) = (\rho(x,y), \theta(x,y))$
- Gaussian weight functions

  $$w_k(\mathbf{u}) = \exp\left((\mathbf{u}-\boldsymbol{\mu}_k)^\top \boldsymbol{\Sigma}_k^{-1}(\mathbf{u}-\boldsymbol{\mu}_k)\right)$$

  learnable covariance $\boldsymbol{\Sigma}$ and mean $\boldsymbol{\mu}$
- Patch operator

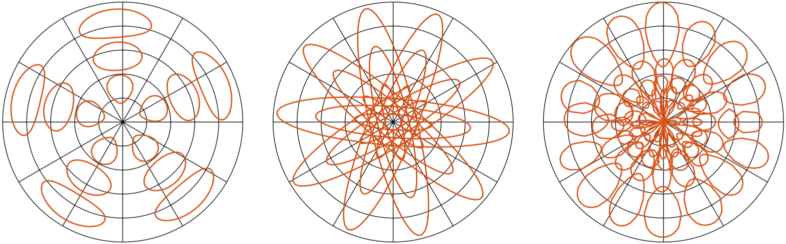
  $$(D(x)f)_k = \int_{\mathcal{X}} w_k(\mathbf{u}(x,y))f(y)dy$$
- Spatial convolution

  $$(f \star g)(x) = \sum_k (D(x)f)_k \cdot g_k$$



Monti, Boscaini, Masci, Rodolà, Svoboda, B 2016

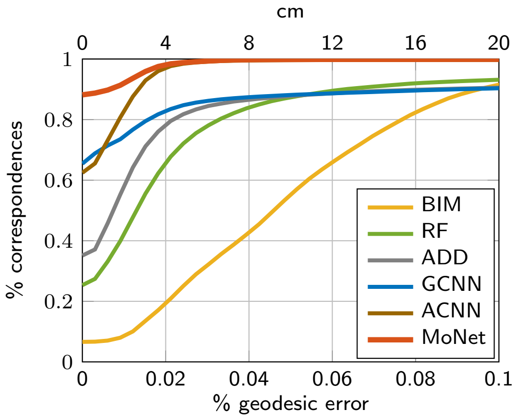
Patch operator weight functions



GCNN ACNN MoNet

Masci, Boscaini, B, Vandergheynst 2016 (GCNN); Boscaini, Masci, Rodolà, B 2016 (ACNN); Monti, Boscaini, Masci, Rodolà, Svoboda, B 2016 (MoNet)

Correspondence performance



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

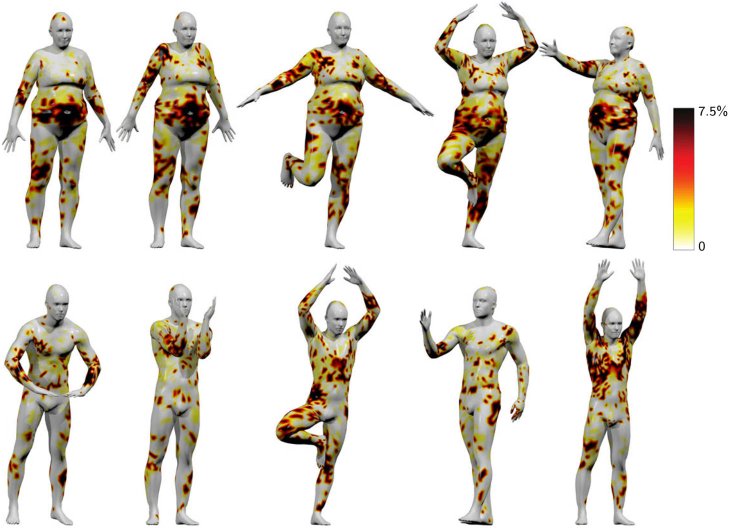
Methods: Kim et al. 2011 (BIM); Rodolà et al. 2014 (RF); Boscaini, Masci, Rodolà, B, Cremers 2015 (ADD); Masci, Boscaini, B, Vandergheynst 2015 (GCNN); Boscaini, Masci, Rodolà, B 2016 (ACNN); Monti, Boscaini, Masci, Rodolà, Svoboda, B 2016 (MoNet); data: Bogo et al. 2014 (FAUST); benchmark: Kim et al. 2011

Correspondence error: Blended Intrinsic Map

Pointwise geodesic error (in % of geodesic diameter)

Kim, Lipman, Funkhouser 2011

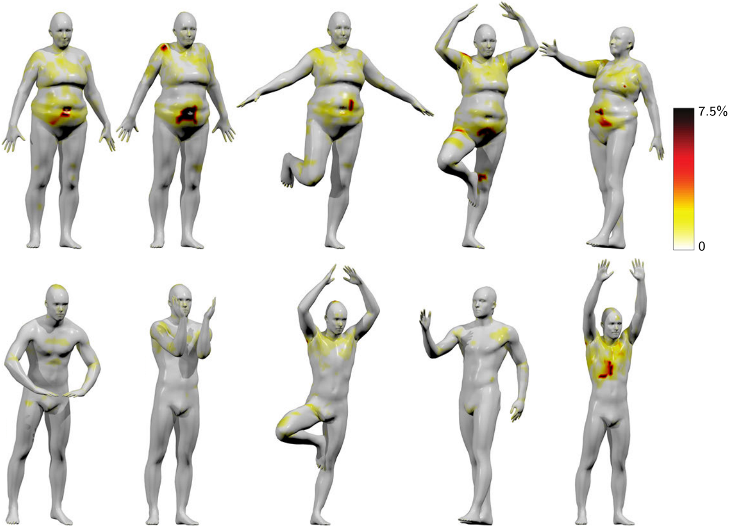
Correspondence error: GCNN



Pointwise geodesic error (in % of geodesic diameter)

Masci, Boscaini, B, Vandergheynst 2015

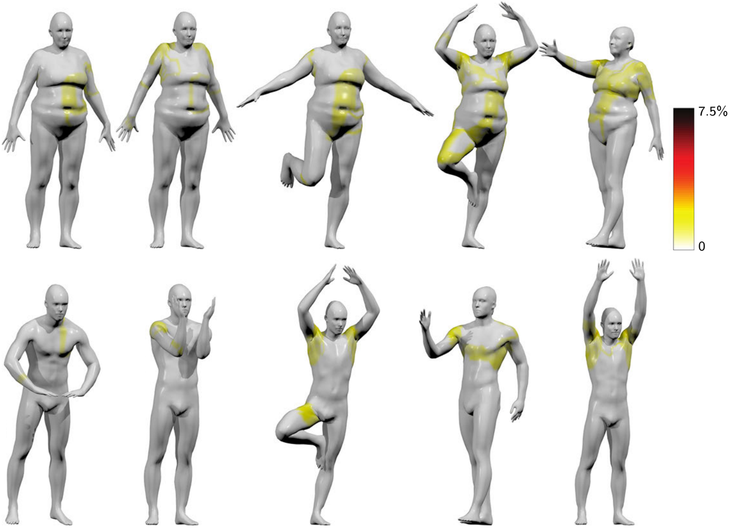
Correspondence error: ACNN



Pointwise geodesic error (in % of geodesic diameter)

Boscaini, Masci, Rodolà, B 2016

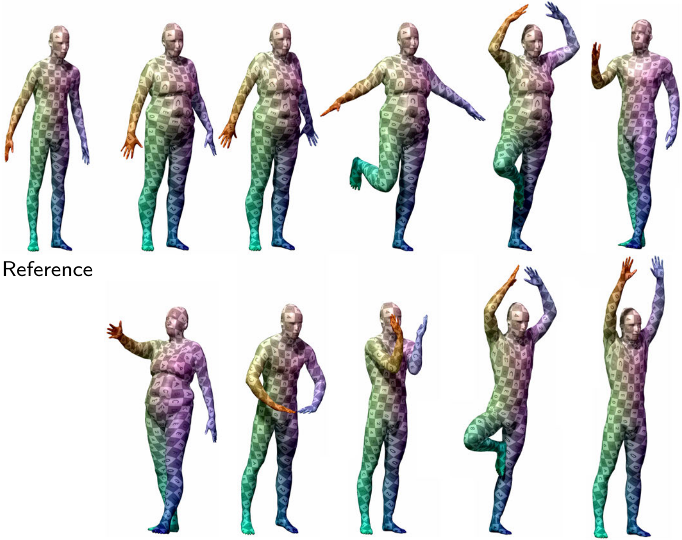
Correspondence error: MoNet



Pointwise geodesic error (in % of geodesic diameter)

Monti, Boscaini, Masci, Rodolà, Svoboda, B 2016

MoNet correspondence visualization

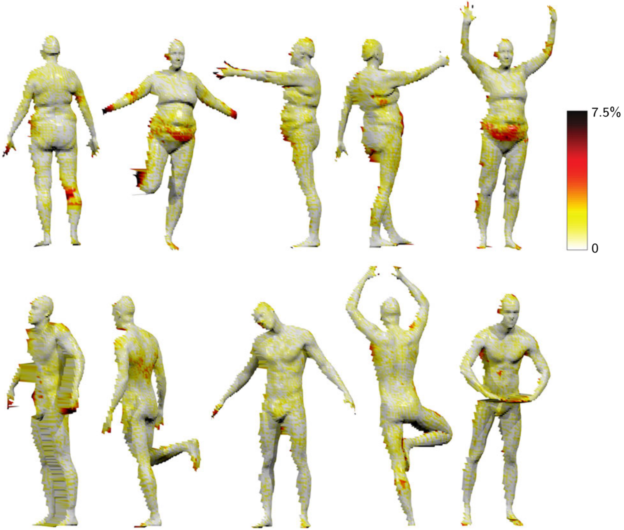


Reference

Texture transferred from reference to query shapes

Monti, Boscaini, Masci, Rodolà, Svoboda, B 2016

Correspondence with MoNet: Range images



Pointwise geodesic error (in % of geodesic diameter)

Monti, Boscaini, Masci, Rodolà, Svoboda, B 2016

Correspondence with MoNet: Range images

Correspondence visualization (similar colors encode corresponding points)
Training: FAUST / Testing: FAUST

Monti, Boscaini, Masci, Rodolà, Svoboda, B 2016

Correspondence with MoNet: Range images

Reference

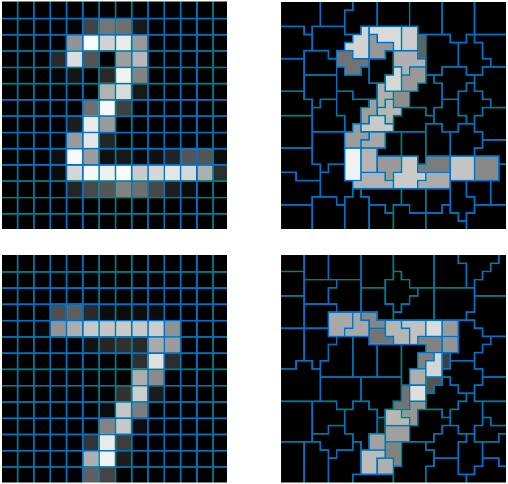
Correspondence visualization (similar colors encode corresponding points)
Training: FAUST / Testing: SCAPE+TOSCA

Monti, Boscaini, Masci, Rodolà, Svoboda, B 2016

Summary

- 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.

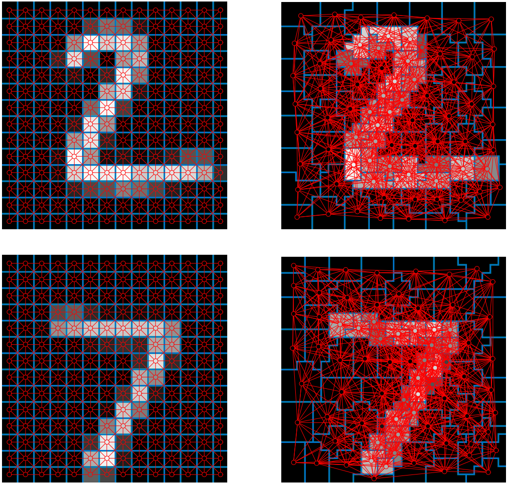
Learning on graphs: MNIST classification



Regular grid　　Superpixels

Monti, Boscaini, Masci, Rodolà, Svoboda, B 2016

Learning on graphs: MNIST classification



Regular grid   Superpixels

Monti, Boscaini, Masci, Rodolà, Svoboda, B 2016

Learning on graphs: MNIST classification

Dataset	**LeNet5**[1]	**Spectral CNN**[2]	**MoNet**[3]
*Full grid	99.33%	99.14%	99.19%
*$\frac{1}{4}$ grid	98.59%	97.51%	98.16%
300 Superpixels	-	88.05%	**97.30%**
150 Superpixels	-	80.94%	**96.75%**
75 Superpixels	-	75.62%	**91.11%**

Classification accuracy of different methods on MNIST dataset
*All images have the same graph

[1]LeCun et al. 1998; [2]Defferrard, Bresson, Vandergheynst 2016; [3]Monti, Boscaini, Masci, Rodolà, Svoboda, B 2016

Learning on graphs: citation networks

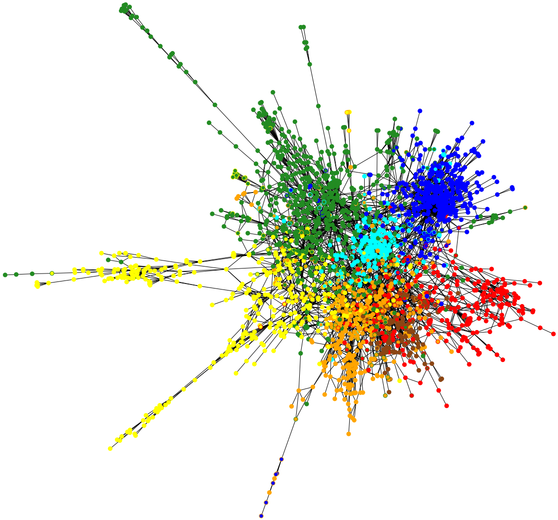


Figure: Monti, Boscaini, Masci, Rodolà, Svoboda, B 2016; data: Sen et al. 2008

Learning on graphs: citation networks

Method	**Cora**[1]	**PubMed**[2]
Manifold Regularization[3]	59.5%	70.7%
Semidefinite Embedding[4]	59.0%	71.1%
Label Propagation[5]	68.0%	63.0%
DeepWalk[6]	67.2%	65.3%
Planetoid[7]	75.7%	77.2%
Graph Convolutional Net[8]	81.59±0.42%	78.72±0.25%
MoNet[9]	**81.69**±0.48%	**78.81**±0.44%

Classification accuracy of different methods on citation network datasets

Data: [1,2]Sen et al. 2008; methods: [3]Belkin et al. 2006; [4]Weston et al. 2012; [5]Zhu et al. 2003; [6]Perozzi et al. 2014; [7]Yang et al. 2016; [8]Kipf, Welling 2016; [9]Monti, Boscaini, Masci, Rodolà, Svoboda, B 2016

F. Monti

D. Boscaini

J. Masci

E. Rodolà

J. Svoboda

Supported by

erc

KTI/CTI
DIE FÖRDERAGENTUR FÜR INNOVATION
L'AGENCE POUR LA PROMOTION DE L'INNOVATION
L'AGENZIA PER LA PROMOZIONE DELL'INNOVAZIONE
THE INNOVATION PROMOTION AGENCY

HASLERSTIFTUNG

Google

NVIDIA

intel

Thank you!