# Laplacian Spectral Functions, Kernels, and Distances

Theory & Applications to Geometry Processing
& Shape Analysis

**Giuseppe Patanè**
CNR-IMATI, Genova - Italy
patane@ge.imati.cnr.it

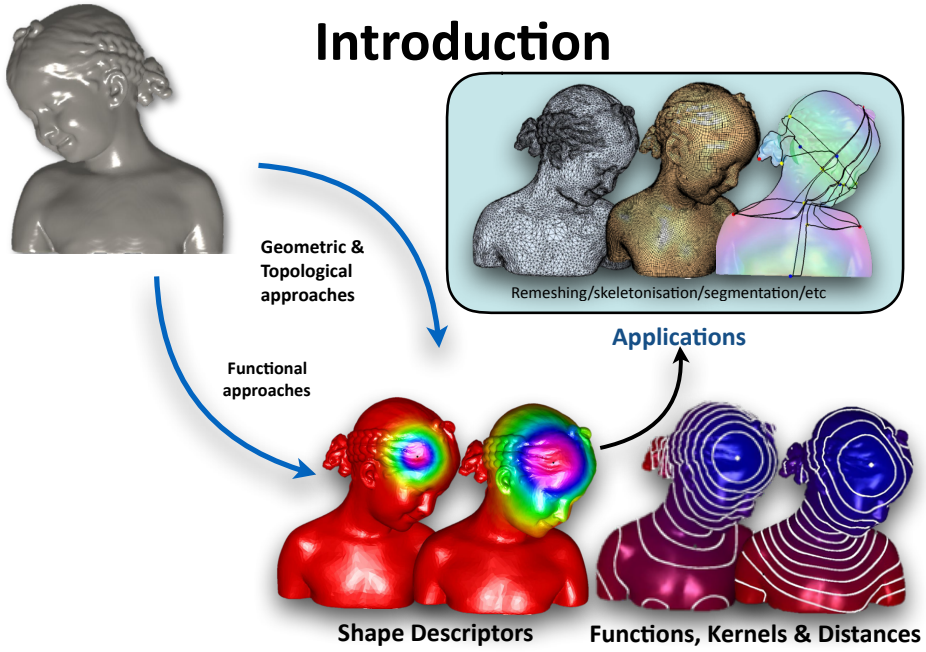
Consiglio Nazionale delle Ricerche

# Myself

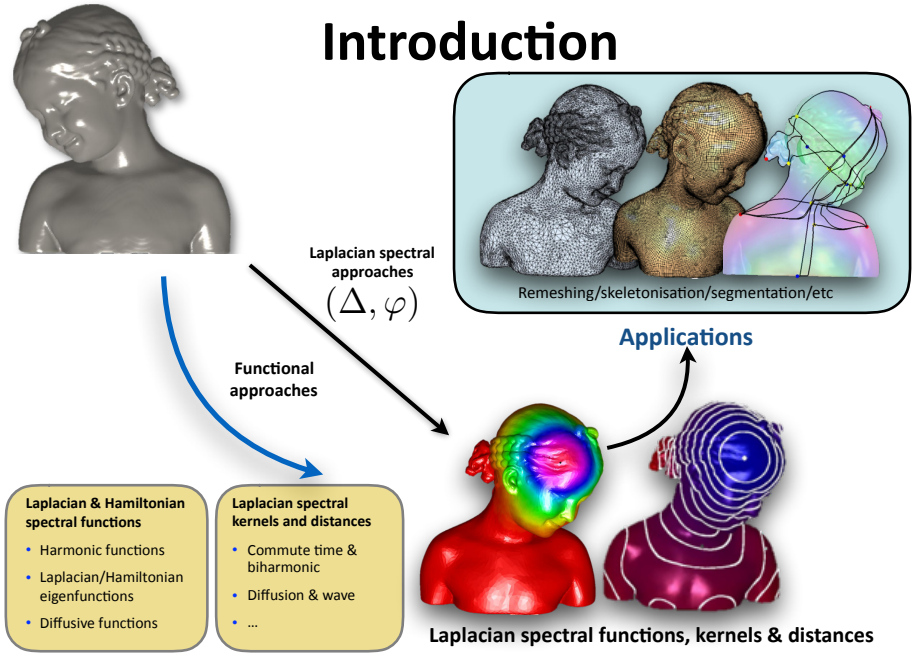
- **Researcher at CNR-IMATI** & Member of the "Shape and Semantics Modelling Group" (since 2001)
- **Responsible of the research line "Numerical Geometry and Signal Processing"**
- **National Scientific Habilitation**
  - Full Professor in *Computer Science* (INF01)
  - Full Professor in *Systems for Information Processing* (ENG-09/H1)
- **My research and training interests** lie at the intersection of
  - **Computer Science:** Computer Graphics & Multimedia, Machine Learning
  - **Engineering**: Information & Signal Processing
  - **Applied Mathematics**: Numerical Analysis
- **ERC Sector: PE6 Computer Science & Informatics**

Consiglio Nazionale delle Ricerche

# Introduction

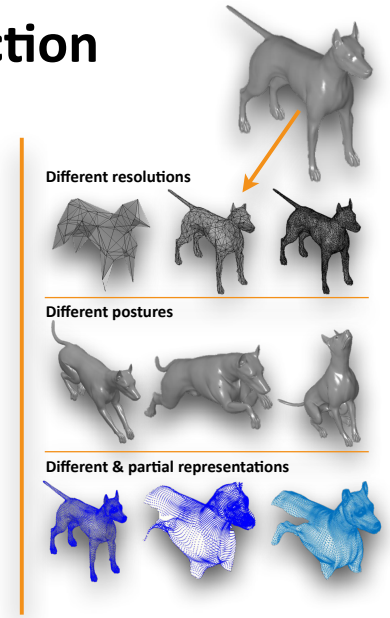



# Introduction



# Introduction

- **Target properties** of the Laplacian spectral functions, kernels, and distances
  - **smoothness & orthonormality**
  - **intrinsic definition**; ie., independent of data embedding/representation
  - **multi-scale definition**, in order to encode local and global shape features
  - **invariance to shape transformations**; eg., isometries for pose invariance
  - **compact support** & **localisation at feature/ seed points** for encoding local geometry properties & saving memory space
  - **efficient, stable,** and **parameter-free computation**

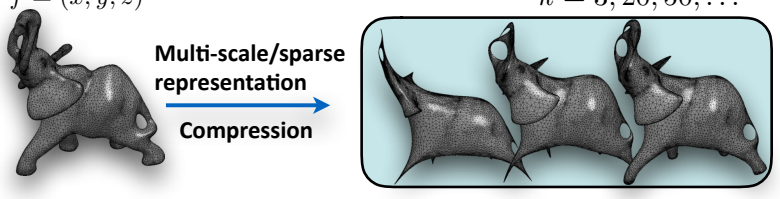



# Introduction

- Working on the **space of scalar functions** defined on the input domain (eg., surface, volume), we can address
  - **multi-scale signal representations and denoising**, by projecting the input signals/data on a set of (*multi-scale*) basis functions
  - **sparse representations**, by choosing a low number of basis functions in order to achieve a target approximation accuracy
  - **compression**, by quantising the representation coefficients

$$f = \sum_{i=1}^{k} \alpha_i \varphi_i$$

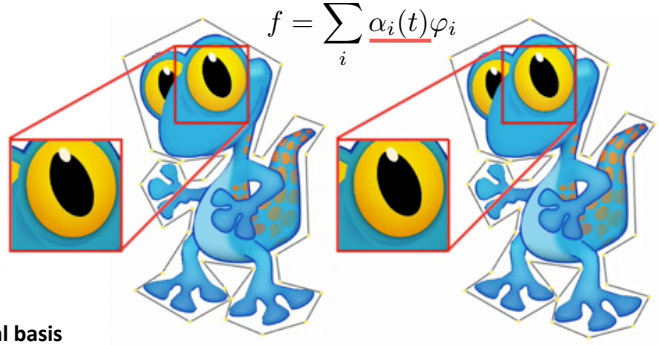
$f = (x, y, z)$ $\qquad k = 3, 20, 50, \ldots$




# Introduction

- Working on the **space of scalar functions** defined on the input domain (eg., surface, volume), we can address
  - **shape deformations**, by modifying the coefficients that express the *geometry* of the input surface in terms of **geometry-driven** or **shape-intrinsic basis functions** (eg., harmonic barycentric coordinates)

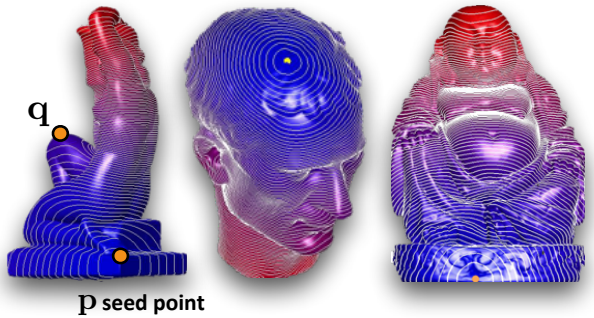
$$f = \sum_i \alpha_i(t) \varphi_i$$



**Global basis** **Local basis**

# Introduction

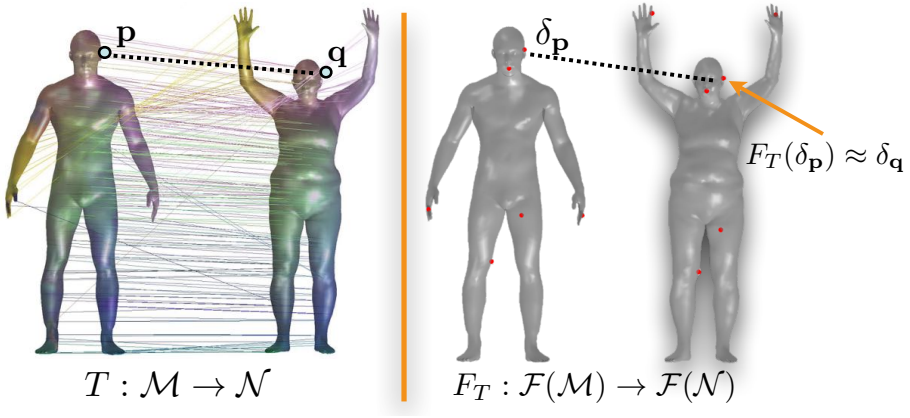
- Working on the **space of scalar functions** defined on the input domain (eg., surface, volume), we can address
  - the definition of **Laplacian spectral kernels and distances**, as a filtered combination of the Laplacian spectral basis




$$d^2(\mathbf{p}, \mathbf{q}) := \sum_i \alpha_i |\varphi_i(\mathbf{p}) - \varphi_i(\mathbf{q})|^2$$

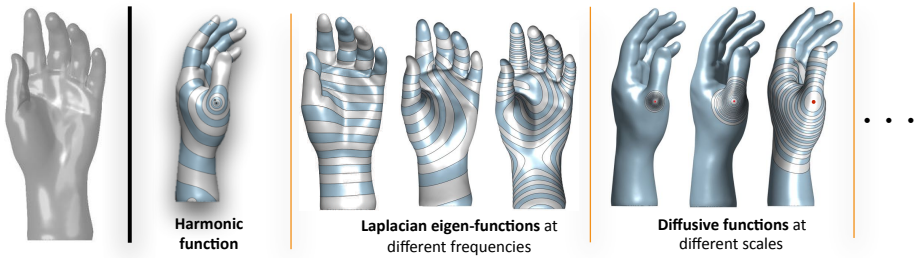
# Introduction

- Working on the **space of scalar functions** defined on the input domain (eg., surface, volume), we can address
  - **shape correspondence and comparison**, by expressing the problem with respect to a basis and converting it to a linear or least-squares problem



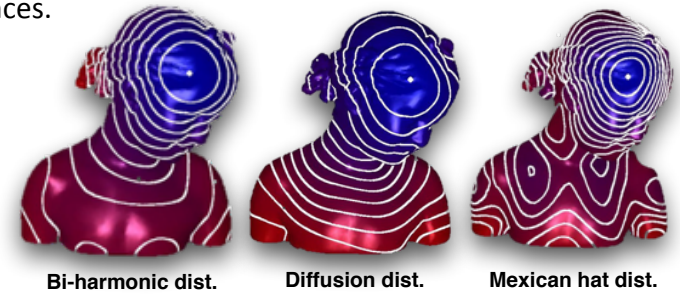

# Goals

- Review of previous work on the definition, discretisation, and computation of **Laplacian & Hamiltonian spectral functions**
  - harmonic functions
  - Laplacian/Hamiltonian eigenfunctions
  - diffusive functions, as solutions to the heat equation
  - …



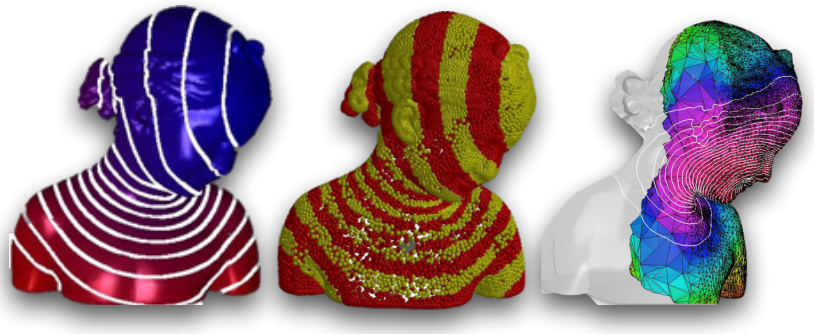

# Goals

- Review of previous work on the definition, discretisation, and computation of **Laplacian spectral kernels and distances** (eg. commute time, biharmonic, wave kernel distances) by
  - filtering the Laplacian spectrum
  - generalising results on the heat diffusion kernels and distances.



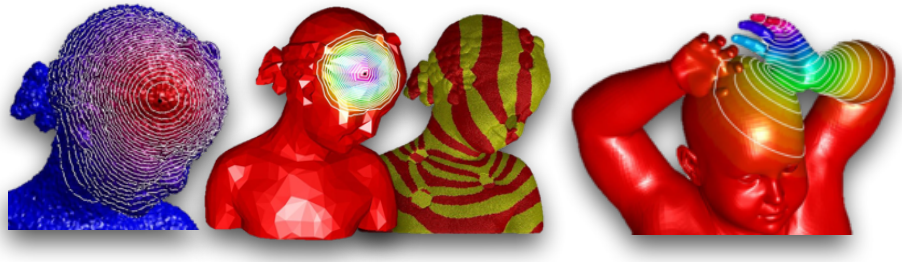

# Goals

- Our review will be “independent” of
  - data dimensionality (surface, volume, nD data)
  - discretisation of the input domain (mesh, point set) and the Laplace-Beltrami operator



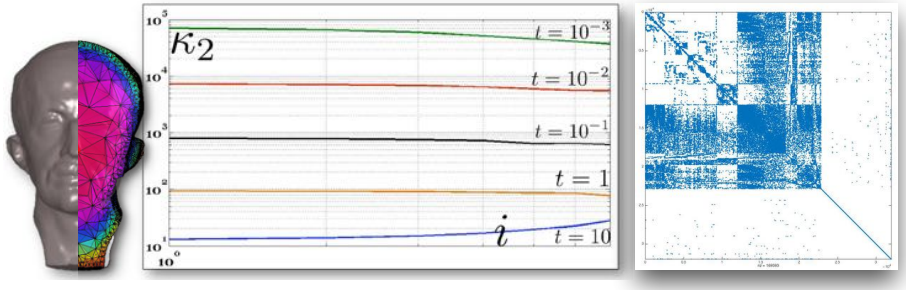
# Goals

- Analysis of preview work on the computation of the Laplacian spectral functions, kernels, and distances in terms of
  - robustness with respect to the discretisation of the input domain: connectivity, sampling, and smoothness (eg. geometric/topological noise)



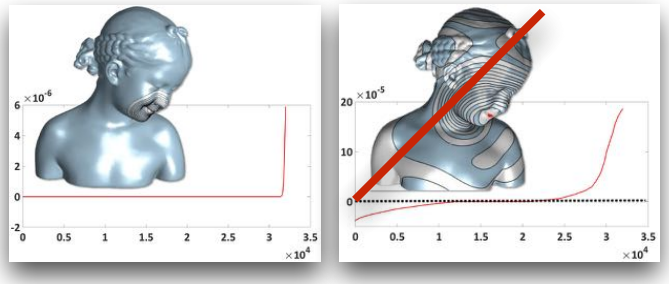
# Goals

- Analysis of preview work on the computation of the Laplacian spectral functions, kernels, and distances in terms of
  - numerical properties (eg., sparsity, conditioning number) of the Laplacian matrix and filter behaviour

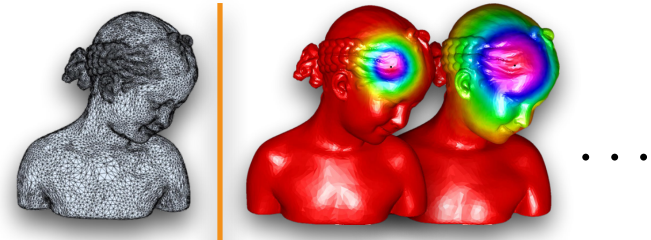



# Goals

- Analysis of preview work on the computation of the Laplacian spectral functions, kernels, and distances in terms of
  - numerical accuracy/stability: convergence, Gibbs phen.
  - computational cost & storage overhead
  - selection of parameters & heuristics



# Goals



$\mathcal{M}$ **n points**

**3D Shape**

$\Phi \in \mathbb{R}^{n \times n}$

**Space of scalar functions defined on M**

- In the space of scalar functions defined on M, we represent
  - point-wise or piecewise linear **scalar functions as vectors**
  - **linear operators as matrices**
- Numerical linear algebra is the main tool for addressing applications in spectral geometry processing and shape analysis

# Goals

- **Focus & Novelty:** unified review of the definition, discretisation, and computation of Laplacian spectral functions, kernels, and distances, independent of the data dimensionality and discretisation of both the input domain and the Laplace-Beltrami operator
- **Applications to geometry processing & shape analysis**
  - Geodesics & signal approximation
  - Diffusion smoothing, distances & descriptors
  - Laplacian spectral kernels & distances for shape comparison
- **Previous STARs** have addressed
  - the comparison of different discrete Laplacians[Zhang07]
  - Laplacian spectral smoothing[Taubin99]
  - surface coding & spectral partitioning[Karni00]
  - shape deformation based on differential coordinates[Sorkine06]
  - applications to shape modeling & geometry processing[Lèvy06]
  - diffusion shape analysis[Bronstein12] & comparison[Biasotti15]

# Outline

- **Laplacian & Hamiltonian operators**
  - Continuous & discrete differential operators
  - Harmonic functions
  - Laplacian & Hamiltonian eigenfunctions
  - Heat diffusion kernels
  - Properties & computation
- **Applications**
  - Geometry processing
  - Shape analysis
- **Laplacian spectral kernels & distances**
  - Properties & computation
- **Conclusions**

# Laplacian & Hamiltonian Operators

Consiglio Nazionale delle Ricerche

# Laplacian equations

- Continuous case

$$\Delta := \mathrm{div}(\mathrm{grad})$$

- Harmonic equation

$$\Delta f = 0$$

- Laplacian eigenvalue problem

$$\Delta f = \lambda f$$

- Heat diffusion equation

$$(\partial_t + \Delta) F(\cdot, t) = 0$$

# Discrete Laplacians

- **Aim:** review of previous work on the discretisation of the Laplace-Beltrami operator through a **unified representation of the discrete Laplacians** that is **independent of the**
  - "**dimensionality**" of the input domain: surfaces, volumes, nD data
  - **discretisation** of the input domain: graphs, triangle/ polygonal/tetrahedral meshes, point sets
  - **Laplacian weights**, as entries of the Laplacian matrix.

# Discrete Laplacians

- We represent the Laplacian matrix for graphs, meshes, and point sets in a "unified" way as

$$\tilde{\mathbf{L}} = \mathbf{B}^{-1}\mathbf{L}$$

**L** sparse, symm., positive semi-definite, **L1=0**

**B** sparse, symm., positive definite

**B-scalar product** $\langle \mathbf{f},\mathbf{g}\rangle_{\mathbf{B}} := \mathbf{f}^{\top}\mathbf{B}\mathbf{g}$ on the space of scalar functions defined on the input domain

- **Main properties**
  - Positive semi-definiteness: $\langle \tilde{\mathbf{L}}\mathbf{f}, \mathbf{f}\rangle_{\mathbf{B}} = \mathbf{f}^{\top}\mathbf{L}\mathbf{f} \geq 0$
  - Null eigenvalue: $\tilde{\mathbf{L}}\mathbf{1} = \mathbf{0}$
  - **B**-self-adjointness: $\langle \tilde{\mathbf{L}}\mathbf{f}, \mathbf{g}\rangle_{\mathbf{B}} = \langle \mathbf{f}, \tilde{\mathbf{L}}\mathbf{g}\rangle_{\mathbf{B}}$

# Discrete Laplacians

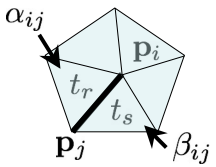
- Linear FEM Laplacian matrix[Reuter06] on triangle meshes

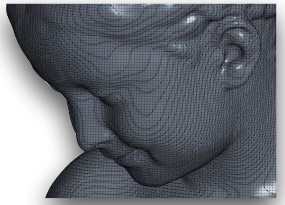
**Stiffness matrix**

$$L(i,j) := \begin{cases} w(i,j) := -\frac{\cot\alpha_{ij} + \cot\beta_{ij}}{2} & j \in N(i) \\ -\sum_{k\in N(i)} w(i,j) & i = j \\ 0 & \text{else} \end{cases}$$

**Mass matrix**

$$B(i,j) := \begin{cases} \frac{|t_r|+|t_s|}{12} & j \in N(i), \\ \frac{\sum_{k\in N(i)}|t_k|}{6} & i = j, \\ 0 & \text{else}, \end{cases}$$



$\tilde{\mathbf{L}} = \mathbf{B}^{-1}\mathbf{L}$

- Voronoi-cotg on triangle[Desbrun99,Pinkall99] & polygonal meshes[Alexa11,Herholz11], Curvature-based Laplacians[Aflalo2013]
- Anisotropic Voronoi-cotg weights[Andreux14,Shi08,Kim13]

# Discrete Laplacians

- According to[Aflalo2013], we consider the curvature-based Laplacian

$$\tilde{\mathbf{L}} := \mathbf{K}^{-1}\mathbf{A}^{-1}\mathbf{L}$$

  - **A** is the diagonal matrix whose $i^{th}$ component is the sum of the areas of the triangles that contain the vertex i (**area mass matrix**)
  - **K** is the diagonal matrix whose entries are the **regularised Gaussian curvature** at the vertices (**curvature-based weight matrix**)
  - **L** is the **stiffness matrix** with cotangent weights.

# Discrete Laplacians

- Laplacian matrix on point sets[Belkin03-06-08,Liu12]

$$L(i,j) = \begin{cases} \frac{1}{4\pi t^2} \exp\left(-\frac{\|\mathbf{p}_i - \mathbf{p}_j\|_2^2}{4t}\right) & i \neq j \\ -\sum_{k \neq i} L(i,k) & i = j \end{cases}$$

$$B(i,i) = v_i$$

**Area of the approximated Voronoi cell**

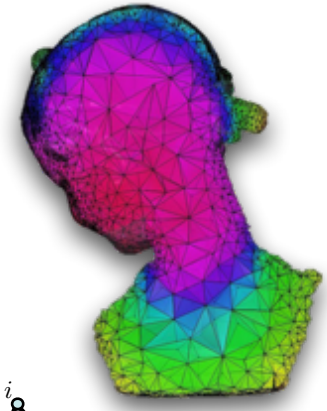
$$\tilde{\mathbf{L}} = \mathbf{B}^{-1}\mathbf{L}$$

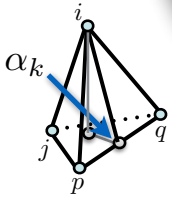
# Discrete Laplacians

$$L(i,j) = \begin{cases} \frac{1}{6} \sum_{k=1}^{n} l_k \cot \alpha_k & j \in N(i) \\ -\sum_{k \neq i} L(i,k) & i = j \end{cases}$$

**B** encodes tetrahedral volumes

$$\tilde{\mathbf{L}} = \mathbf{B}^{-1}\mathbf{L}$$






# Discrete Laplacians

- **B area-driven matrix**
  - – Linear FEM weights
    - [Reuter2006,Vallet2008]
  - – Voronoi-cotg weights
    - [Desbrun1999,Aflalo2013]
  - – Mean-value weights
    - [Floater2003]

  **T-mesh**
  - – Polygonal weights
    - [Alexa2011]

  **Polyg-mesh**
  - – Voronoi-exp weights
    - [Liu2012]

  **Points**
- **B=I** (Euclidean product)
  - – Cotg weights
    - [Pinkall1999]

  **T-mesh**
  - – Exp weights
    - [Belkin2003-06-08]

  **Points**

- **B volume-driven matrix**
  - – Volumetric cotg-weights
    - [Liao09,Tong03]

  **Tet-mesh**

$$\tilde{\mathbf{L}} = \mathbf{B}^{-1}\mathbf{L}$$

**Unified representation of the Laplacian matrix on surfaces, volumes, and n-dimensional data**

# Laplace-Beltrami operator

- The **anisotropic Laplace-Beltrami operator** is defined as

$$\Delta_{\mathbf{D}} := \operatorname{div}(\mathbf{D}\nabla f)$$

where the tensor **D** is a 2x2 matrix that
  - – applies to vectors belonging to the tangent planes to the surface at its points
  - – controls the direction and strength of the derivation from the isotropic case (**D**:=**I**, **Laplace-Beltrami operator**).
- **Tensor**[Shi08,Andreux14]

$$\mathbf{D} := \operatorname{diag}(\varphi_\alpha(\kappa_m), \varphi_\alpha(\kappa_M)), \qquad \varphi_\alpha(s) := (1 + \alpha|s|)^{-1}$$

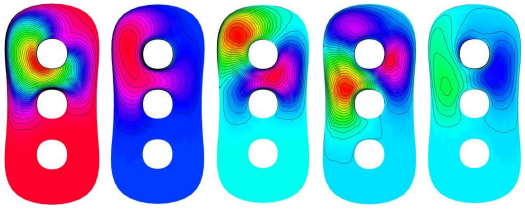
$\kappa_m, \kappa_M$ minimum and maximum curvature

# Hamiltonian operator

- The Hamiltonian operator is defined as

$$\mathcal{H} := \Delta + aV$$

  - V is a potential function, which is aimed at localising the behaviour of the Hamiltonian eigenfunctions in specific regions of the input domain
  - the trade-off parameter *a* controls the global and local support of the Hamiltonian eigenfunctions
- Discretisation $\Delta \rightarrow \tilde{\mathbf{L}} \qquad \mathbf{H} := \tilde{\mathbf{L}} + a\mathbf{V}$



**Hamiltonian eigenfunctions**

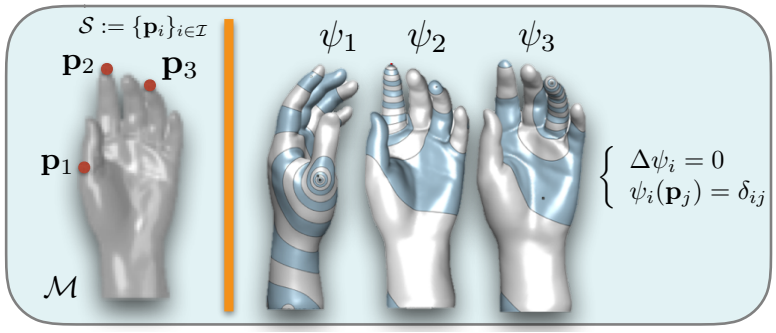
# Harmonic Functions

Consiglio Nazionale delle Ricerche

# Harmonic functions

- Harmonic functions as solutions to the Laplace equation (eg., Dirichlet boundary conditions)

$$f : \mathcal{M} \rightarrow \mathbb{R} \qquad \begin{cases} \Delta f = 0 & \mathcal{M} \\ f = g & \partial\mathcal{M} \end{cases}$$




# Laplacian Eigenfunctions

Consiglio Nazionale delle Ricerche

# Laplacian eigenfunctions

- **Properties of the Laplace-Beltrami operator**
  - **self-adjoint**: $\langle \Delta f, g \rangle_2 = \langle f, \Delta g \rangle_2$, $\forall f, g$
  - **positive semi-definite**: $\langle \Delta f, f \rangle_2 \geq 0$, $\forall f$. In particular, the Laplacian eigenvalues are positive
  - **locality**: the value $\Delta f(\mathbf{p})$ does not depend on $f(\mathbf{q})$, for any couple of distinct points **p**, **q**
  - **null eigenvalue**: the smallest Laplacian eigenvalue is null and the corresponding eigenfunction $\phi$, $\Delta\phi = 0$, is constant
- **Laplacian eigenbasis**

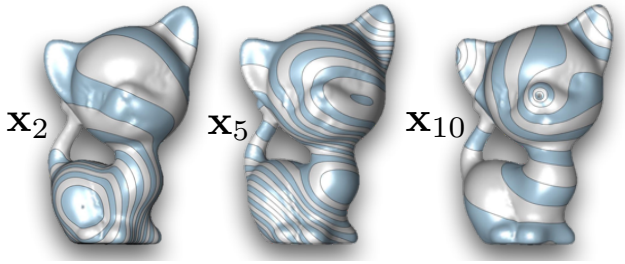
$$\Delta \rightarrow (\lambda_n, \phi_n)_{n=0}^{+\infty}, \qquad \Delta\phi_n = \lambda_n \phi_n$$

# Laplacian eigenfunctions

- The **generalised Laplacian eigensystem** of (**L**,**B**)

$$\mathbf{L}\mathbf{x}_i = \lambda_i \mathbf{B}\mathbf{x}_i \qquad \langle \mathbf{x}_i, \mathbf{x}_j \rangle_{\mathbf{B}} = \delta_{ij} \qquad \lambda_i \leq \lambda_{i+1}$$

defines a set of n linearly independent functions that
  - can be used for the solution to discrete differential equations involving the Laplacian matrix (eg., heat equation)
  - have a different behaviour: eigenfunctions associated with small/ large eigenvalues have a smooth/irregular behaviour




$$\tilde{\mathbf{L}} = \mathbf{B}^{-1}\mathbf{L}$$

# Laplacian spectrum - Computation

- The **O(n²) computation time and storage overhead of the whole Laplacian spectrum** are addressed by computing only **k eigenpairs** - k<<n: O(kn) comput. & storage cost[Golub89]
  - **shift method** computes spectral bands centred around a given eigenvalue
  - **inverse method** computes k smaller/larger eigenvalues
  - **power method** improves the convergence speed of the computation, by considering a power of the input matrix
- **Numerically unstable computations** of the Laplacian eigenpairs are due to
  - **multiple eigenvalues**, associated with high dimensional eigenspaces
  - **switched and/or numerically close eigenvalues** with respect to the approximation accuracy of the solver of the Laplacian eigenproblem

and are independent of the quality of the discretisation of the input domain.

# Laplacian spectrum - Stability

- Perturb the input Laplacian matrix $\tilde{\mathbf{L}} + \epsilon\mathbf{E}$ and compute the eigenvalue of the new problem

$$(\lambda(\epsilon), \mathbf{x}(\epsilon)): \qquad (\tilde{\mathbf{L}} + \epsilon\mathbf{E})\mathbf{x}(\epsilon) = \lambda(\epsilon)\mathbf{x}(\epsilon), \qquad \lambda(0) = \lambda,\ \mathbf{x}(0) = \mathbf{x}$$

whose initial conditions are the eigenpairs of (**L**,**B**).

- The size of the derivative $\lambda'(0)$ indicates the variation that a Laplacian eigenvalue undergoes when the Laplacian matrix is perturbed.
- For a **single eigenvalue**, the upper bound

$$|\lambda'(0)| \leq \|\mathbf{E}\mathbf{x}\|_{\mathbf{B}} \leq \|\mathbf{E}\|_{\mathbf{B}}$$

shows that **its computation is stable**.

# Laplacian spectrum - Stability

- Considering an **eigenvalue of multiplicity m, m>1,** and the approximation

$$\lambda(\delta) \approx \lambda^m + \mathcal{O}(\delta^{1/m})$$

  a perturbation of order $10^{-m}$ induces a change of order 0.1.
- For the perturbation of Laplacian eigenvectors[Golub89],

$$\|\mathbf{x}_i - \mathbf{x}_j\|_2 \leq \epsilon \sum_{j \neq i} \frac{\left|\mathbf{x}_i^\top \mathbf{E} \mathbf{x}_j\right|}{\lambda_i - \lambda_j} + \mathcal{O}(\epsilon^2)$$

  close eigenvalues generally induce *numerical instabilities*.
- Laplacian eigenspaces are generally stable to perturbations (—>projection operator)

# Applications

- **Spectral graph theory & Machine Learning**
  - Graph partitioning[Chung97,Fiedler93,Mohar93,Koren03]
  - Reduction of the bandwidth of sparse matrices[Golub89,Alpert99,Diaz02]
  - Dimensionality reduction with spectral embeddings[Belkin03,Xiao10]
- **Shape analysis**
  - Shape segmentation[Liu07,Zhang05]
  - Shape correspondence[Jain07,Jain&ZhangK07]
  - Shape comparison[Marini11,Reuter05-06-07,Rustamov07,Wardetzky07,Jain06-07]
  - Spectral kernels and distances
    - bi-harmonic kernels/distances[Lipman10,Rustamov11]
    - diffusion kernels/ distances[Bronstein10-11,Coifman06,Gebal09,Lafon06,Luo09,Hammond11,Patanè10]
    - wave kernels/distances[Bronstein11,Aubry11]

# Applications

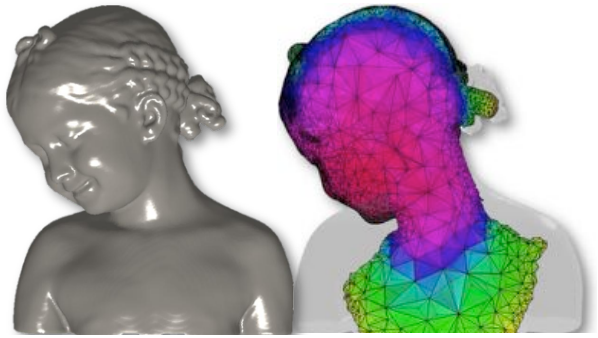
- **Geometry processing**
  - Data reduction[Belkin03-08] & compression[Karni00]
  - Discrete differential forms[Desbrun99-05,Gu03]
  - Design of low-pass filters & Implicit mesh fairing[Taubin95,Desbrun99,Kim05,Pinkall93,Zhang03]
  - Mesh watermarking & Geometry compression[Obuchi01-02,Karni00]
  - Approximation and smoothing of scalar functions[Patanè13]
  - Surface deformation[Levy06,Sorkine04,Vallet08,Zhang07]
  - Local/global parameterisation[Floater,Patanè04-07,Zhang05]
  - Surface quadrangulation[Dong05]

# Heat Equation & Kernel

Consiglio Nazionale delle Ricerche

Shape Modelling Group CNR IMATI-GE

# Heat diffusion equation



Surface Volume

$$\begin{cases} (\partial_t + \Delta) F(\cdot, t) = 0 \\ F(\cdot, 0) = f \end{cases}$$

# Heat diffusion equation

- The solution to the heat equation can be expressed in terms of
  - the **Laplacian spectrum** $(\lambda_n, \phi_n)_{n=0}^{+\infty}$

$$F(\mathbf{p}, t) = \langle K_t(\mathbf{p}, \cdot), f \rangle_2, \quad K_t(\mathbf{p}, \mathbf{q}) = \sum_{n=0}^{+\infty} \exp(-\lambda_n t) \phi_n(\mathbf{p}) \phi_n(\mathbf{q})$$

diffusion kernel

  - the action of the **diffusion operator** $\Phi_t := \exp(-t\Delta)$

$$F(\cdot, t) = \exp(-t\Delta) f = \sum_{n=0}^{+\infty} \exp(-\lambda_n t) \langle f, \phi_n \rangle_2 \phi_n$$

# Heat diffusion equation

- On surfaces, the diffusion kernel encodes local geometric properties: ie.,
  - for an **isometry** between 2 manifolds[SOG09,Gri06]

$$\Phi : \mathcal{N} \to \mathcal{Q} \qquad K_t^{\mathcal{N}}(\mathbf{p}, \mathbf{q}) = K_t^{\mathcal{Q}}(\Phi(\mathbf{p}), \Phi(\mathbf{q}))$$

  - at small scales[SOG09, Var67], the auto-diffusive function encodes the **Gaussian** and **total** **curvature**

$$K_t(\mathbf{p}, \mathbf{p}) \approx \begin{cases} (4\pi t)^{-1}(1 + 1/3t\kappa(\mathbf{p})) + \mathcal{O}(t^2), \\ (4\pi t)^{3/2}(1 + 1/6s(\mathbf{p})), \end{cases} \quad t \to 0,$$

# Discrete heat kernel

- The solution to the discrete heat equation is

$$\mathbf{F}(t) = \mathbf{K}_t \mathbf{f}, \qquad \mathbf{K}_t \equiv \exp(-t\tilde{\mathbf{L}})$$

discrete heat kernel

- Considering the **spectral factorisation** of the Laplacian matrix (B area/volume-driven matrix)

$$\tilde{\mathbf{L}} = \mathbf{X} \Gamma \mathbf{X}^\top \mathbf{B} \qquad \begin{array}{ll} \Gamma := \mathrm{diag}(\lambda_i)_{i=1}^n & \text{Lapl. eig. val.} \\ \mathbf{X} := [\mathbf{x}_1, \ldots, \mathbf{x}_n] & \text{Lapl. eig. vec.} \end{array}$$

the resulting discrete heat kernel $\mathbf{K}_t$ admits the **spectral representation**

$$\mathbf{K}_t = \mathbf{X} \Gamma_t \mathbf{X}^\top \mathbf{B} \qquad \Gamma_t := \mathrm{diag}(\exp(-\lambda_i t))_{i=1}^n$$

# Previous work

- Previous work for the computation of the heat kernel and the corresponding diffusion distances can be classified as
  - **geometry-driven approach**
    - multi-resolution prolongation operator
  - **numerical approaches**
    - truncated spectral approximation
    - theta-method method
    - power method

    Spectrum-free
  - **numerical approaches - higher precision**
    - Padè-Chebyshev approximation
    - polynomial approximation
    - Krylov subspace projection

# Previous work

- **Truncated spectral approximation** of the heat kernel considers the contribution of the Laplacian eigenvectors related to the k smaller eigenvalues

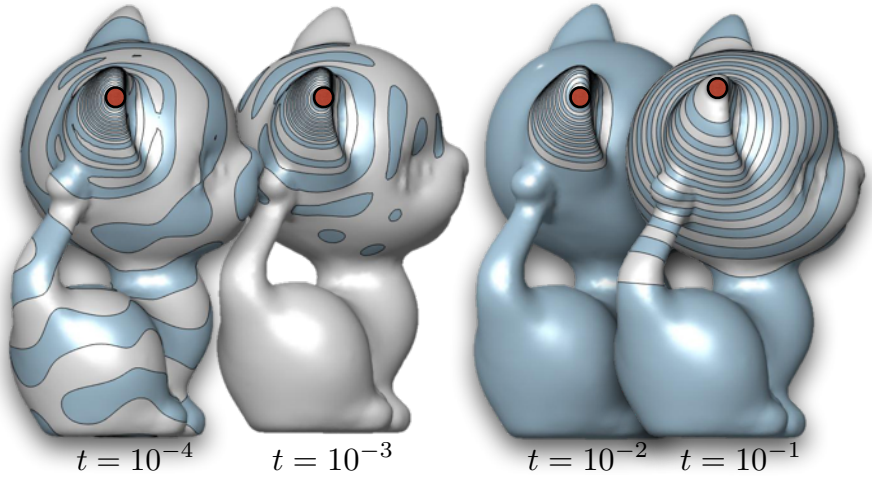
$$\mathbf{F}(t) = \sum_{i=1}^{k} \exp(-\lambda_i t)\langle \mathbf{f}, \mathbf{x}_i \rangle_{\mathbf{B}} \mathbf{x}_i$$

- **Main motivations**
  - exponential decay of the filter as the eigenvalues/time increase
  - the computation of the whole spectrum is not feasible for a large n
  - numerical instabilities due to close/multiple eigenvalues, associated with "high" dimensional eigenspaces (eg., *symmetric shapes*)

- **Remark:** for small scales, we must compute a large number of eigenpairs to achieve a good approximation accuracy

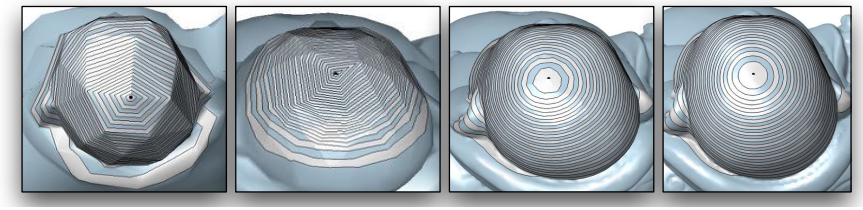
# Previous work

- Truncated spectral approximation (k=200)




# Previous work

- Approximate the heat kernel with **multi-resolution prolongation operators**[Vaxman10]
  - using k eigenpairs on a specific level of a multi-resolutive shape representation
  - selecting k according to time and the shape resolution in the hierarchy
  - prolongating the heat kernel from a given resolution to the input shape



# Previous work

- We discrete the temporal derivative as a finite difference and introduce a convex combination of the values of the solution at consecutive times: $\theta$**-method**

$$\begin{cases} (\partial_t + \Delta)F(\cdot,t) = g \\ F(\cdot,0) = f \end{cases}$$

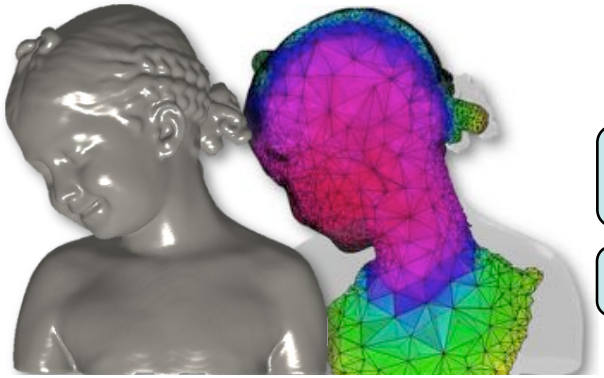
$$\begin{aligned} &\tfrac{1}{\delta t}\left[F(\mathbf{p},t_{k+1}) - F(\mathbf{p},t_k)\right] + \ldots \longrightarrow \partial_t \\ &\quad + \theta\Delta F(\mathbf{p},t_{k+1}) + (1-\theta)\Delta F(\mathbf{p},t_k) = \ldots \longrightarrow \Delta \\ &\quad = g(\mathbf{p},t_{k+1}). \longrightarrow g \end{aligned}$$

- Special cases: **Euler forward** (theta=0) & **backward method** (theta=1)
- In geometry processing: Euler backward method for fairing[Desbrun99]

# Previous work

- **Limitations**
  - Need to select/adapt parameters (eg., number of eigenpairs, iterations, resolution in the hierarchy) to shape/volume details and selected scales
  - No a-priori estimation of the approximation accuracy with respect to the selection of k Laplacian eigenpairs
- **Goal:** review of **numerical approaches with a higher approximation accuracy** achieved by applying a (rational) polynomial approximation to the exponential filter
  - No computation of the Laplacian spectrum
  - High approximation accuracy, adjusted through the selection of the polynomial degree
  - No selection of input parameters

# Spectrum-free approximation



3D Shape Volume

$$\begin{cases} (\partial_t + \Delta)\, F(\cdot,t) = 0 \\ F(\cdot,0) = f \end{cases}$$

$$F(\mathbf{p},t) = \langle K_t(\mathbf{p},\cdot), f\rangle_2$$

$$F(\cdot,t) = \exp(-t\Delta) f$$

# Chebyshev approximation

- **Idea - 1D case**[Golub89]
  - Compute the best (r,r)-degree rational approximation $c_{rr}(x)$ of $e^x$ with respect to the $l_\infty$-norm

$$\exp(x) \approx \alpha_0 - \sum_{i=1}^{r} \alpha_i (x + \theta_i)^{-1}$$

coeff poles

  - $l_\infty$ error between $e^x$ and its rational approximation is lower than $\sigma_{rr} \approx 10^{-r}$ (*unif. rational Cheb. constant*)

# Chebyshev approximation

- Apply the (r,r)-degree Padè-Chebyshev rational approximation to the exponential representation of the solution to the heat equation[Hammond11,Patane14]

$$F(\cdot,t) = \exp(-t\Delta)f$$
$$\approx \alpha_0 f - \sum_{i=1}^{r} \alpha_i \left(t\Delta + \theta_i \mathrm{id}\right)^{-1} f$$
$$= \alpha_0 f + \sum_{i=1}^{r} \alpha_i g_i,$$

Change of basis functions

$$\left(t\Delta + \theta_i \mathrm{id}\right) g_i = -f$$

- Convert the diffusion problem to a set of r differential equations that involve only the Laplace-Beltrami operator

# Chebyshev approximation

- The solution is approximated in a low dimensional space generated by (r+1) functions, which are induced by the input domain, the initial condition f, and the selected scale t.

$$F(\cdot,t) = \sum_{n=0}^{+\infty} \exp(-\lambda_n t)\langle f, \phi_n\rangle_2 \phi_n$$

$(\lambda_n, \phi_n)_{n=0}^{+\infty}$
$\Delta\phi_n = \lambda_n \phi_n$

- **Convergence.** The resulting Padè-Chebyshev approximation converges to the solution as the polynomial degree increases

$$\begin{aligned}\|F_r(\cdot,t) - F(\cdot,t)\|_2 &\le \|c_{rr}(\cdot,t) - \exp(\cdot,t)\|_\infty \|f\|_2 \\ &\le \sigma_{rr}\|f\|_2 \\ &\le 10^{-r}\|f\|_2 \to 0, \quad r \to +\infty\end{aligned}$$

# Chebyshev approximation

- Applying the Chebyshev approximation to $\mathbf{K}_t = \exp(-t\tilde{\mathbf{L}})$, we get the **spectrum-free computation** of the solution to the heat equation

$$\mathbf{K}_t\mathbf{f} \approx \alpha_0\mathbf{f} + \sum_{i=1}^{r} \alpha_i \mathbf{g}_i$$

that requires the solution to r sparse, symmetric linear systems

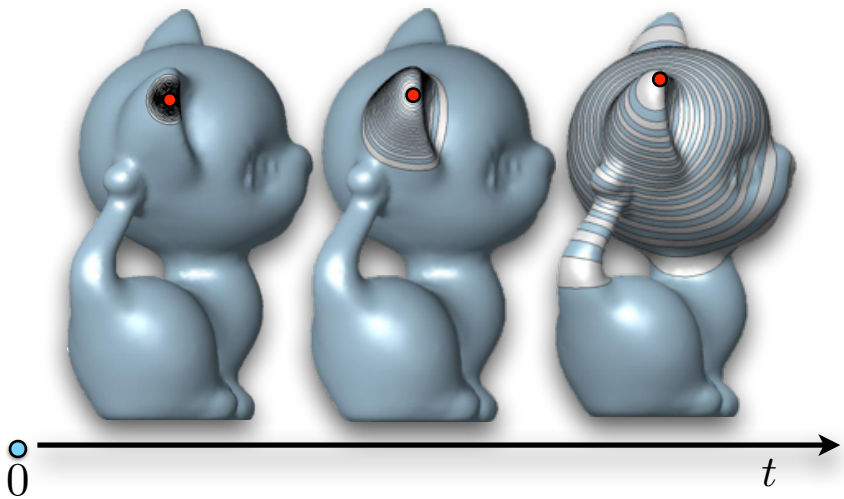
$$(t\mathbf{L} + \theta_i\mathbf{B})\mathbf{g}_i = -\mathbf{B}\mathbf{f}, \quad i = 1,\ldots,r$$

- No input parameters (*degree r is fixed*)
- **Numerical solver**
  - apply an iterative solver for linear systems (e.g., minres): O(rn)-O(rnlog(n)), according to the sparsity of (**L**,**B**)
  - pre-factorise the matrix **B** (if not diagonal); only for several values of t or several initial conditions **F**(·,0)=**f** (eg., diffusion distances)

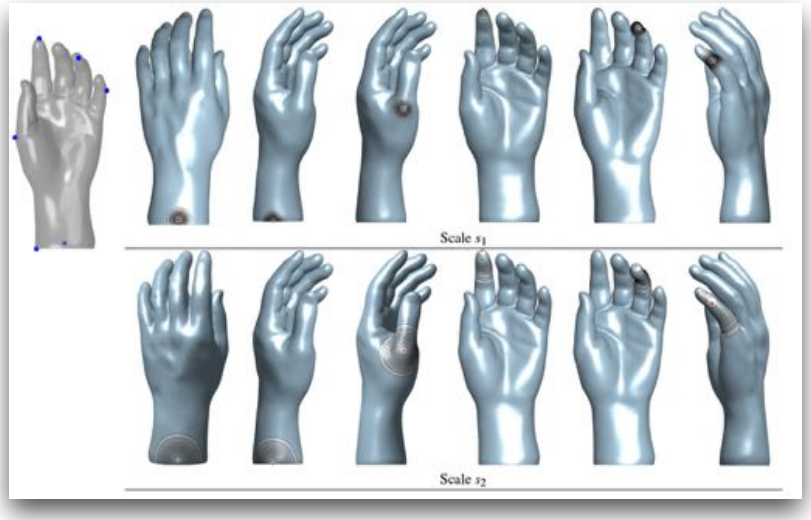
# Chebyshev approximation

seed point

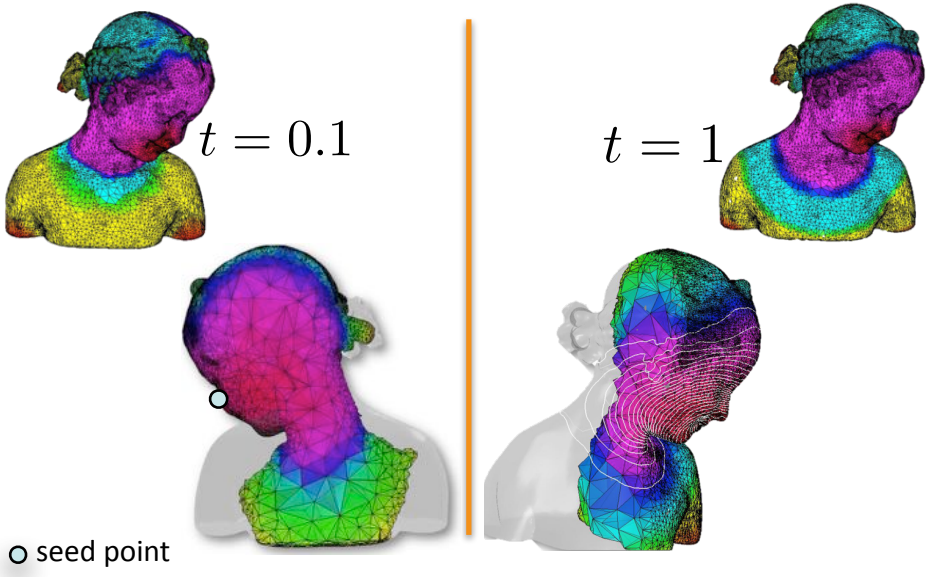
P.C. approx., $r = 7$



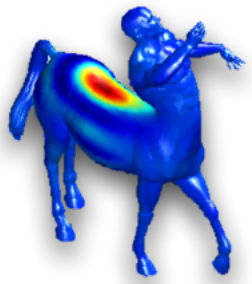
# Chebyshev approximation




# Chebyshev approximation

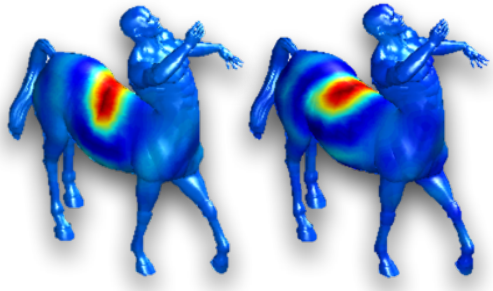



# Chebyshev approximation



**Isotropic diffusion**

Voronoi-cotg Laplacian weights[Desbrun99,Pinkiall99]

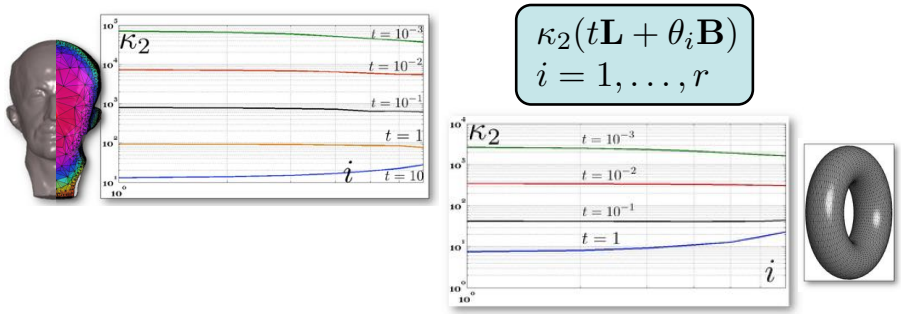


**Anisotropic diffusion**[Boscaini16]

Anisotropic Laplacian weights[Andreux14]

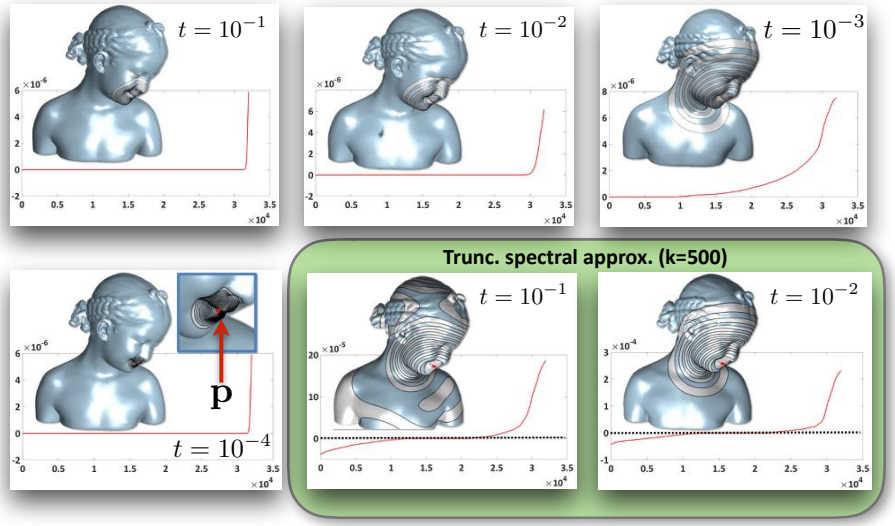
# Numerical stability

- The Cheb./polyn. approx. of exp(-t**C**) is unstable if $||t\mathbf{C}||_2$ is too large.
- From the upper bound $\|t\mathbf{B}^{-1}\mathbf{L}\|_2 \leq t\lambda_{\max}(\mathbf{L})\lambda_{\min}^{-1}(\mathbf{B})$ a well-conditioned matrix **B** guarantees that $||t\mathbf{B}^{-1}\mathbf{L}||_2$ is low.
- If the Laplacian matrix is ill-conditioned, then we can apply specialized Laplacian pre-conditioners[Krishnan13].

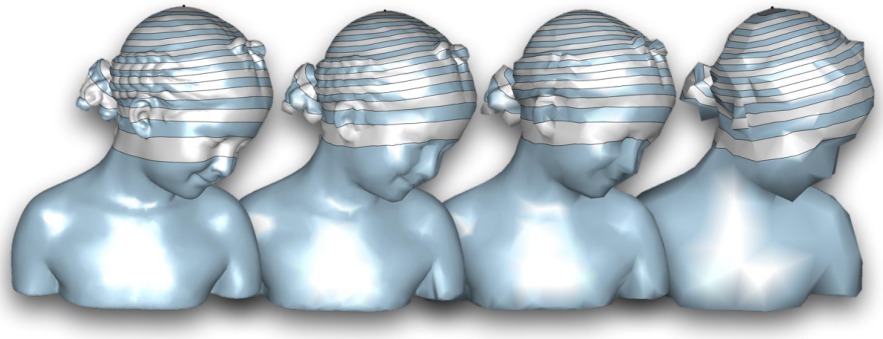


# Numerical stability - Gibbs phen.

$f(\cdot) := K_t(\mathbf{p}, \cdot) \geq 0$ Padè-Chebyshev approx. r:=7

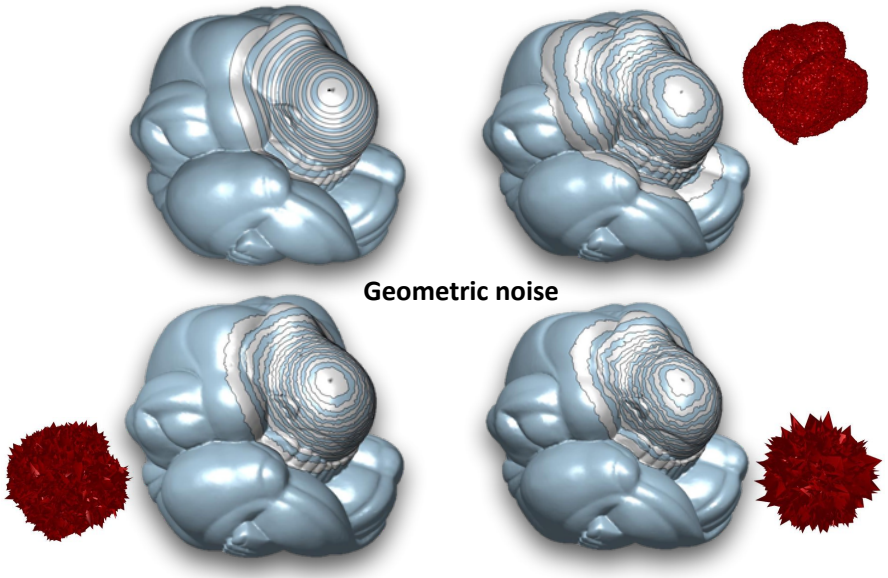



# Robustness



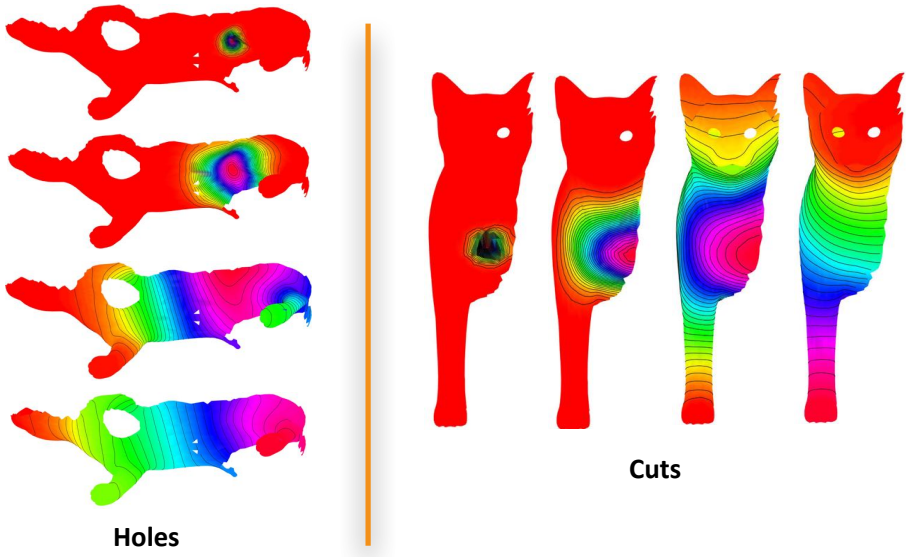
Sampling density

# Robustness



Geometric noise

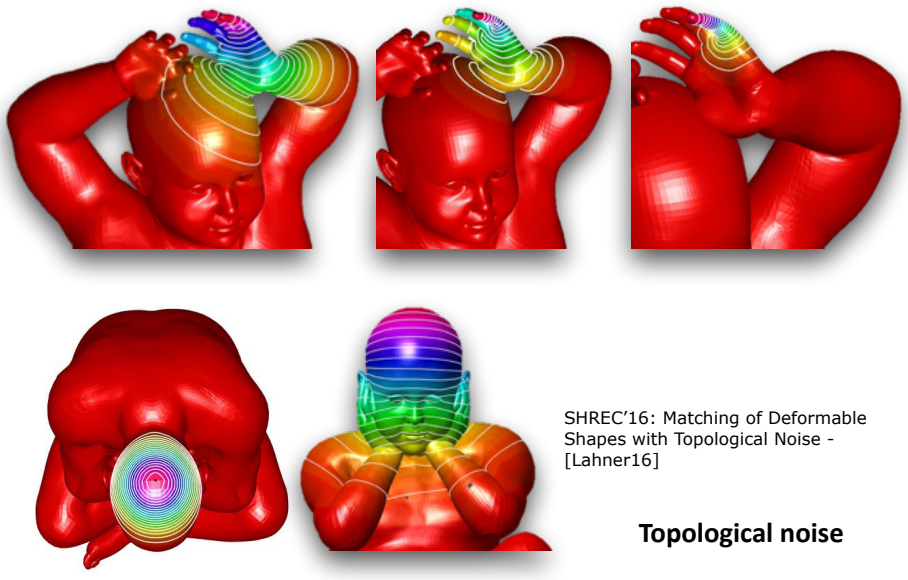
# Robustness



Holes

Cuts

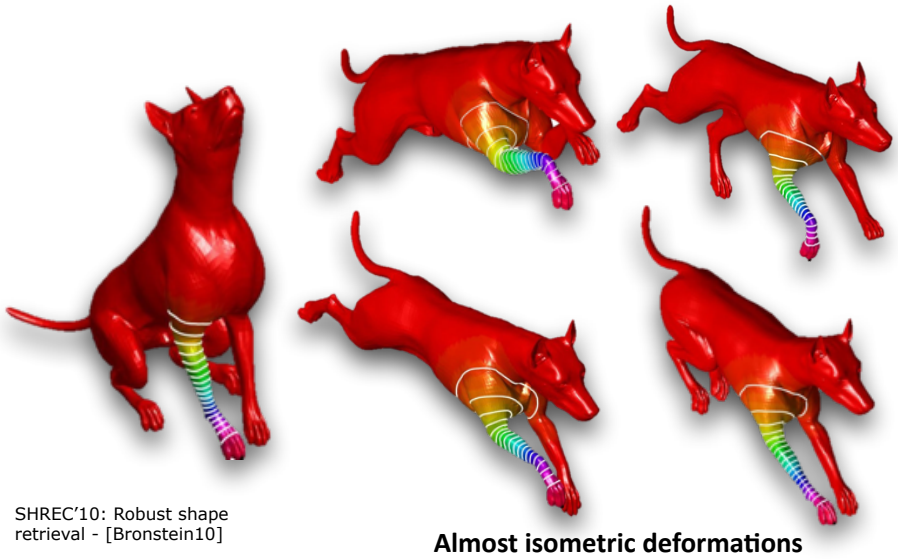
# Robustness



SHREC'16: Matching of Deformable Shapes with Topological Noise - [Lahner16]

**Topological noise**

# Robustness



SHREC'10: Robust shape retrieval - [Bronstein10]

**Almost isometric deformations**

# Polynomial approximation

- **Rational polynomial approximation**[Pusa11] of the exponential filter based on quadrature formulas derived from complex contour integrals.
- **Polynomial approximation**[Golub89]
  - applies the **Taylor power series** to the exponential matrix (*first r terms*)

$$\exp(-t\tilde{\mathbf{L}}) = \sum_{n=0}^{+\infty} \frac{(-t\tilde{\mathbf{L}})^n}{n!}$$

  - has an accuracy lower than the Padè-Chebyshev method (*point-wise instead of uniform convergence*)
  - generalises the 1st order Taylor approximation applied by the power method

# Polynomial approximation

- The discrete spectral kernel is approximated as

$$\mathbf{K}_t\mathbf{f} \approx \sum_{i=0}^{r} \alpha_i \tilde{\mathbf{L}}^i \mathbf{f}$$

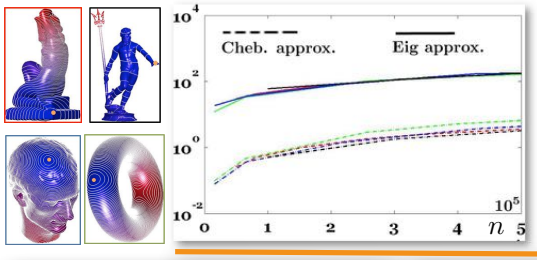
$$= \alpha_0\mathbf{f}_0 + \sum_{i=1}^{r} \alpha_i \mathbf{g}_i, \qquad \mathbf{g}_i := \tilde{\mathbf{L}}^i\mathbf{f} = (\mathbf{B}^{-1}\mathbf{L})^i\mathbf{f}$$

- If **B** is not diagonal (eg., linear/cubic FEM weights), then each vector $\mathbf{g}_i$ is computed through the recursive relation

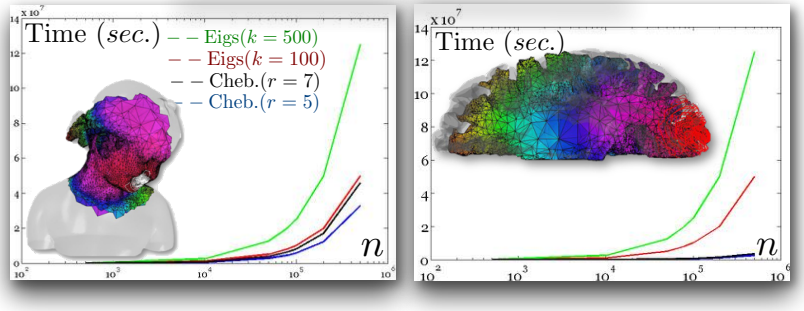
$$\mathbf{g}_i := \tilde{\mathbf{L}}^i\mathbf{f} = (\mathbf{B}^{-1}\mathbf{L})^i\mathbf{f} \qquad \begin{cases} \mathbf{B}\mathbf{g}_1 = \mathbf{L}\mathbf{f} \\ \mathbf{B}\mathbf{g}_i = \mathbf{L}\mathbf{g}_{i-1} \end{cases} \quad i = 2, \ldots, r$$

and we solve r sparse and symmetric linear systems.

# Computational cost




Computational cost for the evaluation of the heat kernel




# Summary

| Method | Numerical scheme | Scales | Comput. cost |
|---|---|---|---|
| **Linear approximation** | | | |
| Trunc. spec. approx. | $\mathbf{F}_k(t) = \sum_{i=1}^{k} \exp(-\lambda_i t)\langle \mathbf{f}, \mathbf{x}_i \rangle_{\mathbf{B}} \mathbf{x}_i$ | Any | $\mathcal{O}(kn)$ |
| Euler backw. approx. | $(t\tilde{\mathbf{L}} + \mathbf{I})\mathbf{F}_{k+1}(t) = \mathbf{F}_k(t)$ | Small | $\mathcal{O}(\tau(n))$ |
| I order Taylor approx. | $\mathbf{BF}(t) = (\mathbf{B} - t\mathbf{L})\mathbf{f}$ | Small | $\mathcal{O}(\tau(n))$ |
| Krylov/Schur approx. | Projection on $\{\mathbf{g}_i := (\mathbf{B}^{-1}\mathbf{L})^i \mathbf{f}\}_{i=1}^{m}$ | Any | $\mathcal{O}(m\tau(n))$, $\mathbf{B} \neq \mathbf{I}$<br>$\mathcal{O}(n)$, $\mathbf{B} = \mathbf{I}$ |
| **Polynomial approximation** | | | |
| Power approx. | $\mathbf{F}(t) = \sum_{i=0}^{m} \mathbf{g}_i / i!$<br>$\mathbf{g}_i := \tilde{\mathbf{L}}^i \mathbf{f}$ | Any | $\mathcal{O}(m\tau(n))$, $\mathbf{B} \neq \mathbf{I}$<br>$\mathcal{O}(n)$, $\mathbf{B} = \mathbf{I}$ |
| **Rational approximation** | | | |
| Padé-Cheb. approx. | $\mathbf{F}(t) = \alpha_0 \mathbf{f} + \sum_{i=1}^{r} \mathbf{g}_i$<br>$(t\mathbf{L} + \theta_i \mathbf{B})\mathbf{g}_i = -\alpha_i \mathbf{Bf}$ | Any | $\mathcal{O}(r\tau(n))$ |
| Contour integral approx. | $\mathbf{F}(t) = \sum_{i=1}^{r} \alpha_i \mathbf{g}_i$<br>$(\alpha_i)_{i=1}^{r}$ quadr. coeff. | Any | $\mathcal{O}(r\tau(n))$ |

# Applications
## Signal Smoothing

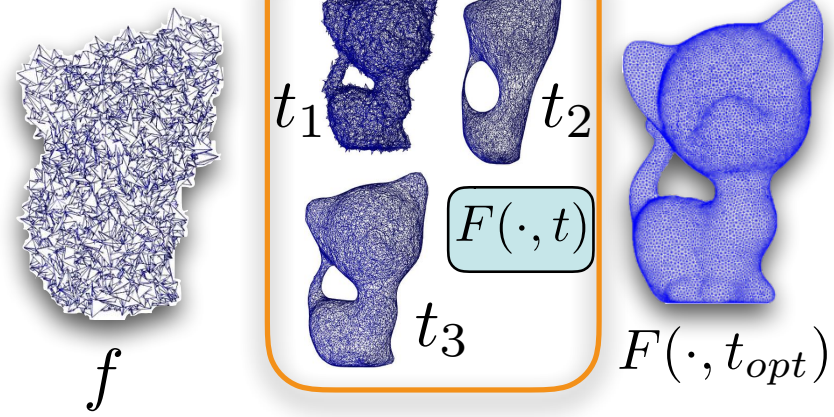
Consiglio Nazionale delle Ricerche

# Applications

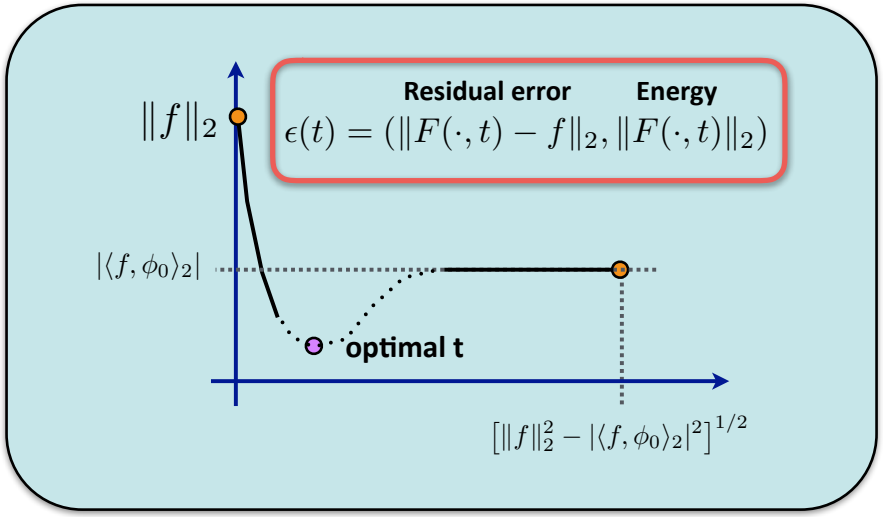
- Applications of the heat diffusion kernel and distance include
  - **multi-scale representations** of functions [Rosenberg97,Patanè10-13]
  - **shape comparison** with heat kernel shape signatures [Bronstein11,Gebal09,Memoli09,Ovsjanikov10,Sun09]
    - intrinsic to the input shape
    - isometry-invariant
    - multi-scale (local vs. global details)
  - **diffusion distances & descriptors** [Aubry11,Belkin03,Coifman06,Gine06,Litman14,Singer06,Smola03]
    - data matching [Lafon06]
    - gradient [Wang09], critical points computation [Luo09]
    - data representation and classification [Smola03]
  - **shape segmentation** [DeGoes08,Gebal08]
  - **dimensionality reduction** [Belkin03,Roweis00,Xiaoa10,Tenenbaum00]
  - **clustering** [Chapelle03]
  - ...

## Diffusive smoothing: optimal scale

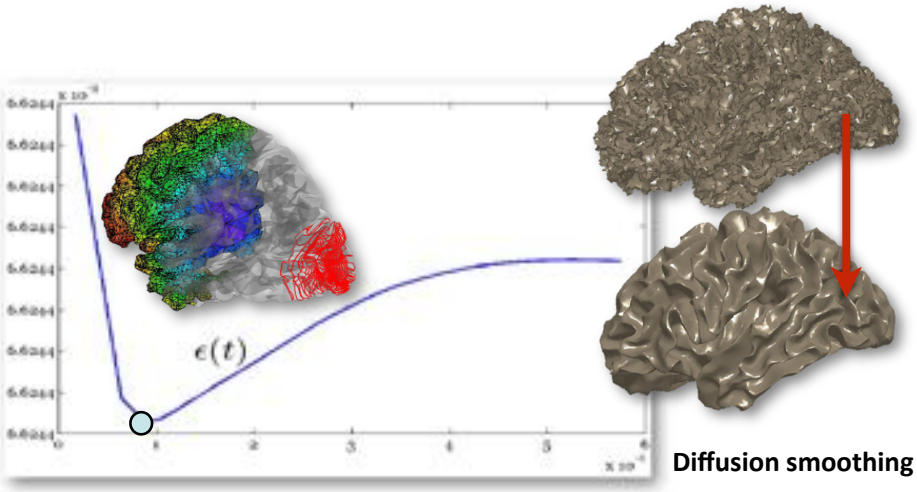
$$\begin{cases} (\partial_t + \Delta)F(\cdot,t) = 0 \\ F(\cdot,0) = f \end{cases}$$




## Diffusive smoothing: optimal scale




## Diffusive smoothing: optimal scale




Diffusion smoothing

**Approx. error:** P.C. <1%; trunc. specr. meth. 12% (k=100)-13% (k=1K)

## Applications

Diffusion distances
& descriptors

Consiglio Nazionale delle Ricerche

Shape Modelling Group CNR IMATI-Ge

# Geodesics approx. *via* heat kernel

- **Geodesic distances** can be expressed in terms of the heat kernel as

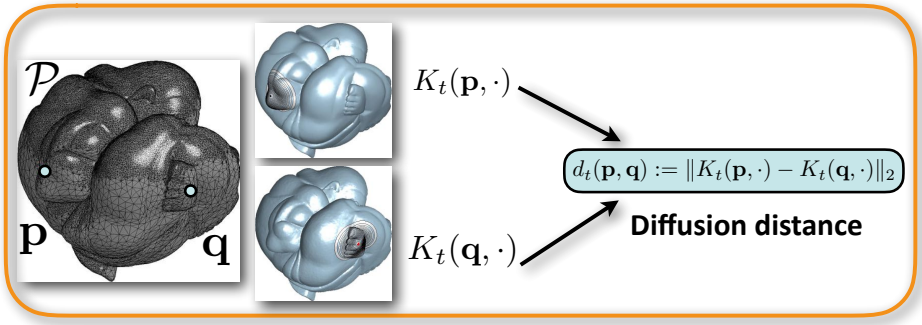
$$d_G(\mathbf{p},\mathbf{q}) = -\lim_{t\to 0}(4t\log K_t(\mathbf{p},\mathbf{q}))$$ **[Varadhan's formula[Varadhan67]]**

  – Otherwise[Crane13],
    - Integrate the heat flow $\partial_t F(\cdot,t) = \Delta F(\cdot,t)$ (*for a fixed t*)
    - Evaluate the vector field $\mathbf{X} := -\nabla F(\cdot,t)/\|\nabla F(\cdot,t)\|_2$
    - Solve the Poisson eq. $\Delta\phi = \mathrm{div}\mathbf{X}$
- **Optimal transportation distances** are approximated through the solution of two sparse linear systems that involve the heat kernel[Solomon15] .
- In both cases, we can apply the spectrum-free approach to guarantee an accurate approximation of the heat kernel.

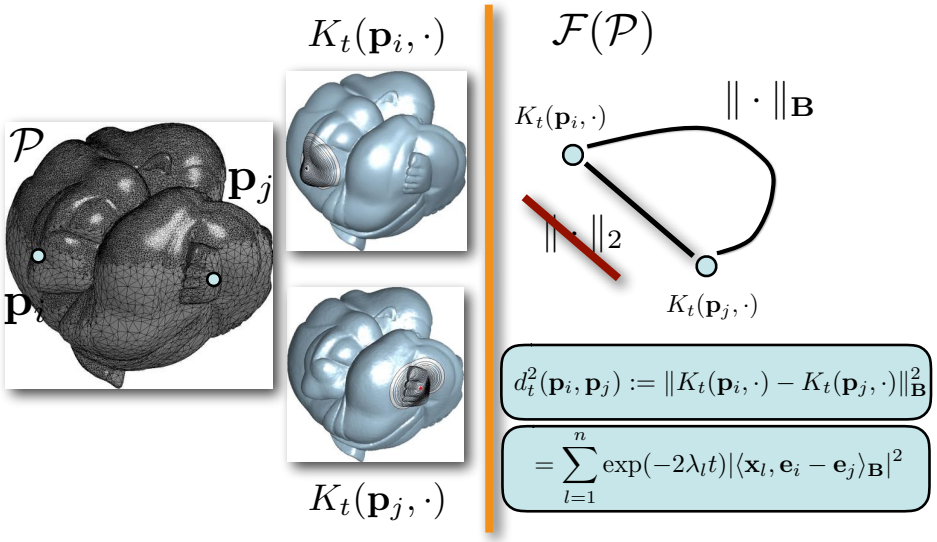
# Heat diffusion distance

- **Idea**: associate a shape M with the **functional space**
  – **F(M)**:={f:M—>R, f scalar function on M}
    - eg., Laplacian eigenproblem, heat equation, etc

  and define the **metric space (M,$d_M$)**, equipped with **diffusion distances derived from $K_t$ on F(M)** (*diffusion geometry*).




# Heat diffusion distance

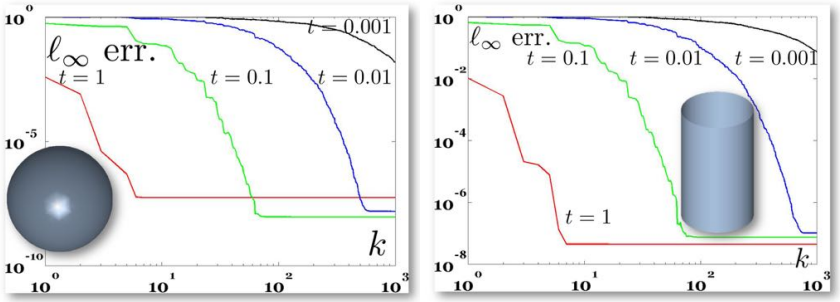



# Heat diffusion distance

- Apply the Padè-Chebyshev of the heat kernel to approximate diffusion distances.

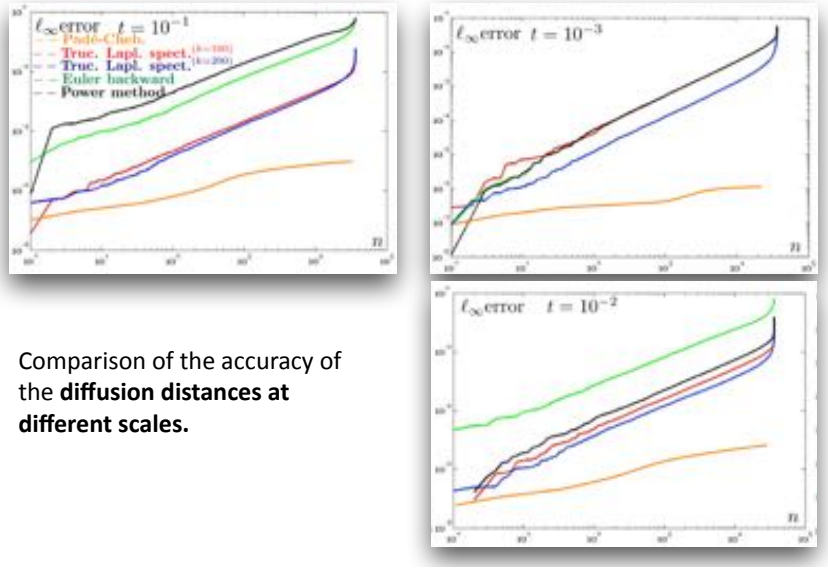
# Approximation accuracy

- Padè-Chebyshev approximation *versus* truncated spectral approximation of the diffusion distances (r=5)



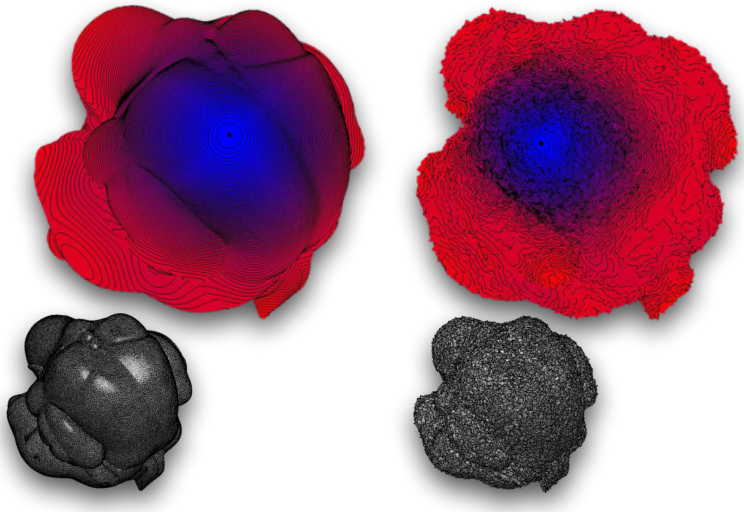

**Padè-Chebyshev approximation error:** for all t, lower than $8.9*10^{-6}$

# Approximation accuracy

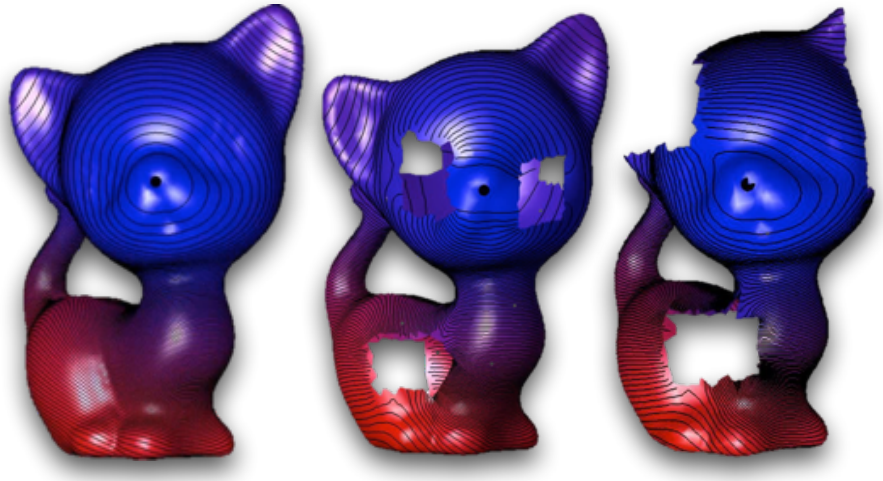



Comparison of the accuracy of the **diffusion distances at different scales.**

# Robustness



# Robustness



# Computational cost

- **Truncated spectral approximation:** $\mathcal{O}(kn)$
- **Padè-Chebyshev approximation:** $\mathcal{O}(r\tau(n))$
  - solution to the heat equation or evaluation of the diffusion distance between 2 points

$$\begin{cases} \mathcal{O}(\tau(n)) \text{ lin. syst. solver} \\ \tau(n) \approx n,\ n\log n \end{cases}$$

  - one-to-all distance (*no pre-factorisation*): $\mathcal{O}(rn\tau(n))$
  - one-to-all distance (*pre-factorisation of* **B**):$\mathcal{O}(n\log n + rn)$
    - if **B** is not diagonal

# Diffusion signatures & descriptors

- Heat kernel signature

$$HKS(\mathbf{p}) := \sum_{n=0}^{+\infty} \exp(-\lambda_n t)|\phi_n(\mathbf{p})|^2$$

- Diffusion embegging

$$DE(\mathbf{p}) := (\exp(-\lambda_n)\phi_n(\mathbf{p}))_{n=0}^{+\infty}$$

- Wave kernel signature

$$WKS(\mathbf{p}) = \sum_{n=0}^{+\infty} \exp(-i\lambda_n t)\phi_n^2(\mathbf{p})$$

# Diffusion signatures & descriptors

- The **heat kernel matrix $K_t$**[Bronstein10,Patanè14] is
  - **self-adjoint** with respect to the **B**-scalar product
  - **intrinsically scale-covariant** (ie., with no a-posteriori normalisation)

$$K_t(\alpha\mathcal{M}) = K_{\alpha^2 t}(\mathcal{M})$$

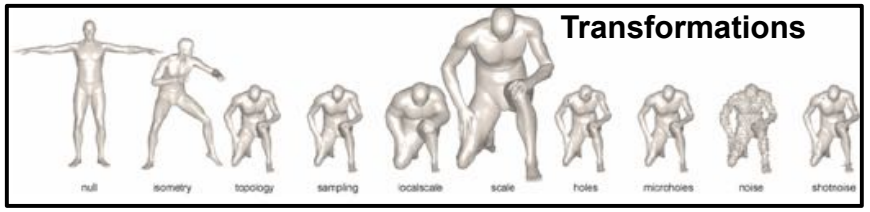
  - **scale-invariant** through a normalisation of the Laplacian eigenvalues

$$K_t(\alpha\mathcal{M}) = K_t(\mathcal{M})$$

  - **stable to noise and irregular sampling**, thus improving the robustness of matching based on heat kernel descriptors[SHREC10]

# Heat kernel & shape comparison

- Diffusion descriptors for shape comparison have been extensively analysed in SHREC contexts.





Bronstein, A. M., Bronstein, M. M., Castellani, U., Falcidieno, B., Fusiello, A., Godil, A., Guibas, L., Kokkinos, I., Lian, Z., Ovsjanikov, M., Patanè, G., Spagnuolo, M., and Toldo, R. 2010. Shrec 2010: robust large-scale shape retrieval benchmark. *Eurographics Workshop on 3D Object Retrieval.*

Bronstein, A. M., Bronstein, M. M., Bustos, B., Castellani, U., Crisani, M., Falcidieno, B., Guibas, L. J., I. Kokkinos, V. M., Isipiran, I., Ovsjanikov, M., Patanè, G., Spagnuolo, M., and Sun, J. 2010. Shrec 2010: robust large-scale shape retrieval benchmark. *Eurographics Workshop on 3D Object Retrieval.*

# Laplacian Spectral Kernels and Distances

Consiglio Nazionale delle Ricerche

# Distances on 3D shapes

- **Geometry-driven approaches:** define the distance on the input shape; eg., geodesics[Mitchell87,Surazhsky05,Kimmel98,Lipman10]
- **Functional approaches:** define the distance in the space of functions on the input surface
  - diffusion distances[Bronstein10-11,Coifman06,Gebal09,Lafon06,Luo09,Hammond11,Patanè10]
  - commute-time & bi-harmomic distances[Lipman10,Rustamov11]
  - wave kernel distances[Bronstein11,Aubry11]
  - random walks[Fouss05,Ramani13], Mexican hat wavelets & distances[Hou12]
- **Mixed approaches:** geodesic distances & optimal transportation distances are approximated in the geometric and function space
  - approximation through the heat kernel[Crane13]
  - multi-dimensional scaling[Bronstein06,Panozzo13]

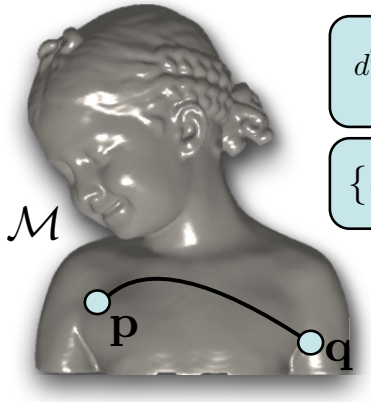
# Spectral distances

- **Aim:** review of previous work on the definition and computation of
  - the commute-time, bi-harmonic, diffusion, wave kernel distances
  - the corresponding embeddings and shape descriptors

  in a unified way by
  - introducing the spectral distances, which are defined by filtering the Laplacian spectrum
  - interpreting the main properties of the spectral distances in terms of the properties of the corresponding filter function

# Spectral distances

- **Idea:** define **spectral distances**[Bronstein11,Patane14-16] by filtering the Laplacian spectrum

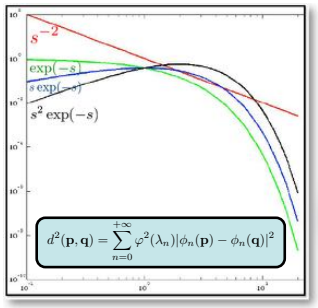


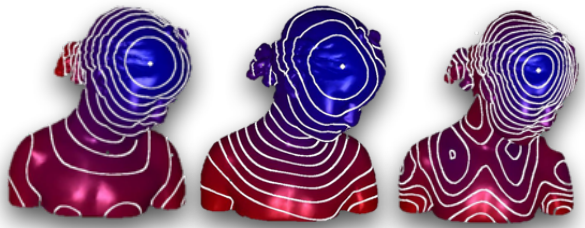

$$d^2(\mathbf{p},\mathbf{q}) = \sum_{n=0}^{+\infty} \varphi^2(\lambda_n)|\phi_n(\mathbf{p}) - \phi_n(\mathbf{q})|^2$$

$$\{(\phi_n, \lambda_n)\}_{n=0}^{+\infty} : \quad \Delta\phi_n = \lambda_n\phi_n$$

$$\varphi : \mathbb{R}^+ \to \mathbb{R} \quad \textbf{filter function}$$

# Spectral distances



**Bi-harmonic dist.** $\varphi^2(s) = s^{-2}$

**Diffusion dist.** $\varphi_t^2(s) = \exp(-st)$

**Mexican hat dist.** $\varphi_t^2(s) = s^{-1}\exp(-st)$

| $\varphi^2(s)$ | Distance |
|---|---|
| $s^{-1}$ | Comm.-time |
| $s^{-2}$ | Biharm. |
| $s^{-k}$ | Poly-harm. |
| $e^{-st}$ | Heat diff. |
| $e^{-ist}$ | Wave ker. |

- Diffusion distances[Bronstein10-11,Coifman06,Gebal09,Lafon06,Luo09,Hammond11,Patanè10]
- Commute-time & bi-harmomic distances[Lipman10,Rustamov11]
- Wave kernel distances[Bronstein11,Aubry11]
- Random walks[Fouss05,Ramani13], Mexican hat wavelets & distances[Hou12]

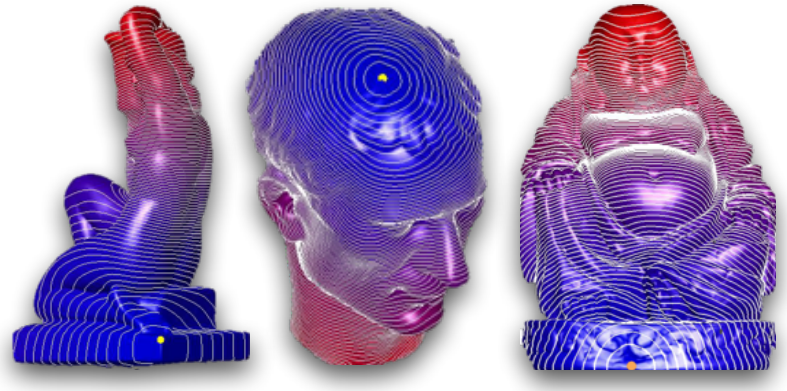
# Spectral distances

- **Commute-time distance**[Bronstein11] are defined the integral of the diffusion distance with respect to scale $\varphi(s) := s^{-1/2}$

$$d^2(\mathbf{p},\mathbf{q}) = \frac{1}{2}\int_0^{+\infty} d_t^2(\mathbf{p},\mathbf{q})\mathrm{d}t = \sum_{n=0}^{+\infty} \lambda_n^{-1}|\phi_n(\mathbf{p}) - \phi_n(\mathbf{q})|^2$$

- **Bi-harmonic distances**[Ovsjanikov12,Lipman10,Rustamov11] $\varphi(s) := s^{-1}$
  - for small distances, they have a nearly geodesic behavior
  - for large distances, they encode global shape properties

# Spectral distances



**Bi-harmonic distances**

# Spectral distances

- The **filters** are defined
  - **analytically** and analogously to Laplacian signal smoothing[Desbrun99,Kim05,Taubin95-96,Zhang03]
  - by **applying supervised learning**[Aflalo11,Litman14] on a data set of 3D shapes
    - optimal spectral signature[Litman14]: linear combination of B-splines by minimising a task-specific loss function
  - by **controlling their behaviour**
    - decay to zero, periodicity
    - normalisation with respect to geometric properties of the domain

  in such a way that the corresponding distances are
    - multi-scale and/or invariant to isometric transformations
    - smooth and/or localised in both time and frequency[Hammond11].

# Spectral distances

- The **smoothness**, **locality**, and **encoding of local/global shape properties** depend on the convergence of the filtered Laplacian eigenvalues to zero
  - increasing the filter decay to zero
    - global shape properties are encoded by the spectral distances, by reducing the influence of eigenfunctions associated with small eigenvalues in the spectral distances
  - reducing the filter decay to zero
    - local shape properties are encoded by the spectral distances.

# Spectral distances

- Given a strictly positive, square-integrable filter that admits the power series' representation

$$\varphi(s) = \sum_{n=0}^{+\infty} \alpha_n s^n$$

we define the **spectral operator**

$$\Phi(f) := \varphi(\Delta) f = \sum_{n=0}^{+\infty} \varphi(\lambda_n) \langle f, \phi_n \rangle_2 \phi_n$$

which is well-defined, linear, continuous, and

$$\Phi(f) = \langle K_\varphi, f \rangle_2 \qquad K_\varphi(\mathbf{p}, \mathbf{q}) = \sum_{n=0}^{+\infty} \varphi(\lambda_n) \phi_n(\mathbf{p}) \phi_n(\mathbf{q})$$

where $K_\varphi$ is **the spectral kernel**.

# Spectral distances

- Analogously to the diffusion distances, previous work has defined the **equivalent representations of the spectral distances**

$$d^2(\mathbf{p}, \mathbf{q}) = \|\Phi(\delta_\mathbf{p}) - \Phi(\delta_\mathbf{q})\|_2^2 \quad \text{Spectral operator}$$

$$= \sum_{n=0}^{+\infty} \varphi^2(\lambda_n) |\phi_n(\mathbf{p}) - \phi_n(\mathbf{q})|^2 \quad \text{Lapl. spectrum}$$

$$= \|K_\varphi(\mathbf{p}, \cdot) - K_\varphi(\mathbf{q}, \cdot)\|_2^2 \quad \text{Spectral kernel}$$

$$= \|\phi(\mathbf{p}) - \phi(\mathbf{q})\|_2^2 \quad \text{Spectral embed.}$$

$$\phi : \mathcal{M} \to \ell_2, \; \phi(\mathbf{p}) = (\varphi(\lambda_n)\phi_n(\mathbf{p}))_{n=0}^{+\infty}$$

# Discrete spectral distances

- Applying the generalised eigendecomposition of the Laplacian matrix, the **discrete spectral kernel** is

$$\tilde{\mathbf{L}} = \mathbf{X} \Gamma \mathbf{X}^\top \mathbf{B} \qquad \mathbf{K}_\varphi = \varphi(\tilde{\mathbf{L}}) = \mathbf{X} \varphi(\Gamma) \mathbf{X}^\top \mathbf{B}$$

and the resulting **discrete spectral distance** is

$$d^2(\mathbf{p}_i, \mathbf{p}_j) = \|\mathbf{K}_\varphi(\mathbf{e}_i - \mathbf{e}_j)\|_\mathbf{B}^2 \quad \text{[spectrum-free approx.]}$$

$$= \sum_{l=1}^{n} \varphi^2(\lambda_l) |\langle \mathbf{x}_l, \mathbf{e}_i - \mathbf{e}_j \rangle_\mathbf{B}|^2 \quad \text{[truncated spectral approx.]}$$

# Discrete spectral distances

- We generalise previous work on the computation of the diffusion kernels/distances to the case of spectral distances
  - **spectrum-free approximation:** considers the representation of the distance in terms of the spectral kernel and apply the
    - polynomial approximation of the filter
    - Padè-Chebyshev approximation of the filter
    - Krylov sub-space projection
  - **truncated spectral approximation:** applies the representation of the distances in terms of the Laplacian spectrum.

# Spectrum-free computation

- Recalling that
  - the **spectral distances** are defined in terms of the spectral kernel as $d(\mathbf{p}_i, \mathbf{p}_j) = \|\mathbf{K}_\varphi(\mathbf{e}_i - \mathbf{e}_j)\|_\mathbf{B}$
  - the **spectral kernel** is achieved by applying filtering the Laplacian matrix as $\mathbf{K}_\varphi = \varphi(\tilde{\mathbf{L}})$

we compute and apply the **best r-degree polynomial approximation of the selected filter** to the Laplacian matrix

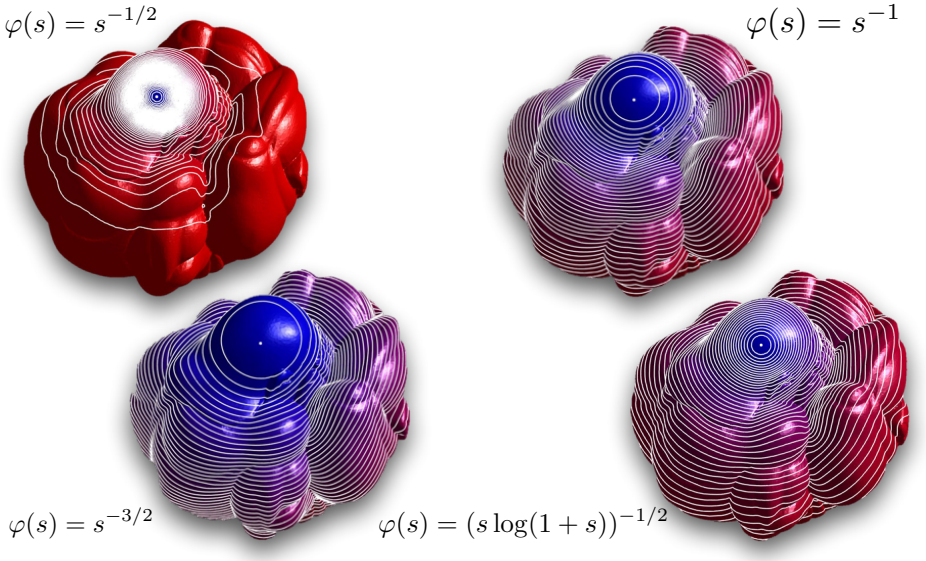
$$\mathbf{K}_\varphi = \varphi(\tilde{\mathbf{L}}) \approx p_r(\tilde{\mathbf{L}})$$

Filter map

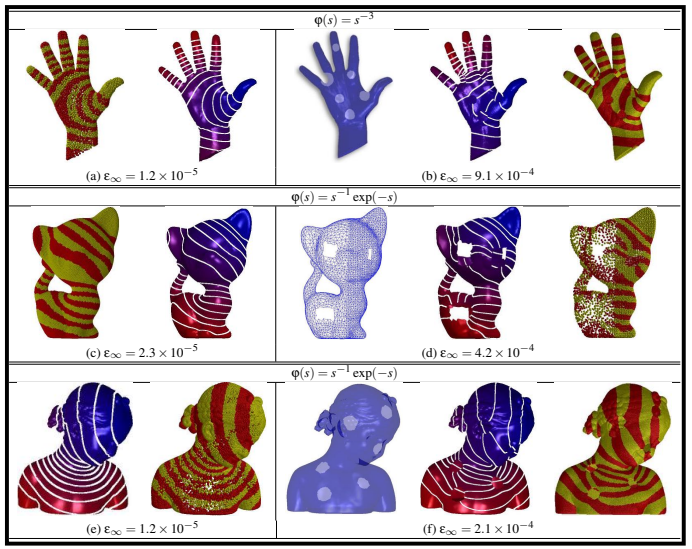
r-degree Taylor polynomial

$$p_r(s) := \sum_{i=0}^{r} \alpha_i s^i$$

# Spectrum-free computation




# Spectrum-free computation




Linear FEM & Voronoi-exp Laplacian matrix

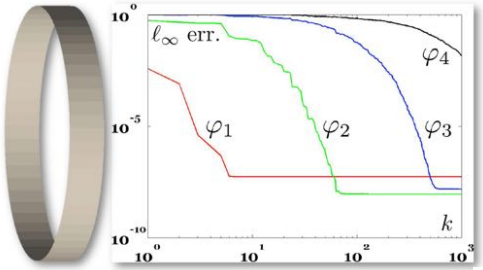
# Truncated spectral approximation

- The spectral distances can be approximated by considering the contribution of the Laplacian eigenvectors related to the smaller eigenvalues

$$d^2(\mathbf{p}_i, \mathbf{p}_j) \approx \sum_{l=1}^{k} \varphi^2(\lambda_l) \left| \langle \mathbf{x}_l, \mathbf{e}_i - \mathbf{e}_j \rangle_{\mathbf{B}} \right|^2$$

  - accurate approximation for filters with a fast decay (periodic filter: eg., wave kernel?)
  - the number of selected eigenpairs must be adapted to local shape details, target approximation accuracy, parameters (eg., time for wave kernel distances): *not a trivial task*

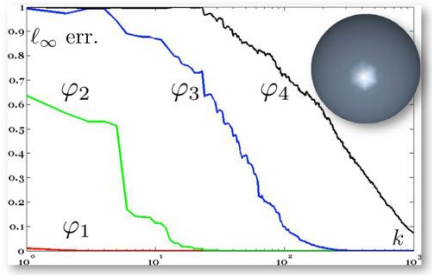
# Approximation accuracy




$$\varphi_1(s) := s^{-1} \exp(-ts)$$
$$\varphi_2(s) := s^{-1/2} \exp(-ts)$$
$$\varphi_3(s) := s^{-1}$$
$$\varphi_4(s) := s^{-1/2}$$

**Approximation of spectral distances**

- **Truncated spectral approximation**: $l_\infty$ error between the ground-truth spectral distances induced by different filters and their approximation with k Laplacian eigenpairs
- **Spectrum-free approximation:** r:=8 degree polynomial and $l_\infty$ error lower than $10^{-4}$




# Discrete spectral distances

- **Truncated spectral approximation**
  - computes k Laplacian eigenpairs in O(kn) time
  - uses the Laplacian eigenpairs to quickly evaluate distances induced by different filters on the same surface
  - generally has an accuracy lower than the spectrum-free approach.
- **Spectrum-free approximation**
  - **Distance evaluation between two points** is reduced to solve r sparse, symmetric, linear systems: $\mathcal{O}(r\tau(n))$
  - **Evaluation of the one-to-all distance**
    - no factorisation of **B**: $\mathcal{O}(rn\tau(n))$
    - with factorisation **B**: $\mathcal{O}(n \log n + rn)$

# Conclusions

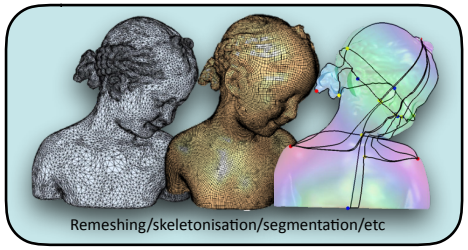
Consiglio Nazionale delle Ricerche

Shape Modelling Group CNR IMATI Ge

# Conclusions

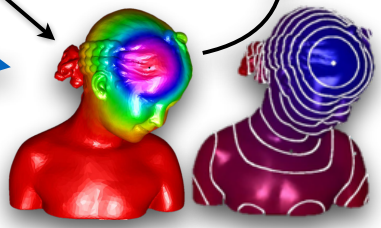
Laplacian spectral approaches $(\Delta, \varphi)$

Functional approaches



Remeshing/skeletonisation/segmentation/etc

Applications



Laplacian spectral functions, kernels & distances

**Main (target) properties**
- Intrinsic & multi-scale definition
- Invariance to uniform scaling/isometries
- Generalisation to n-D data
- Easy computation
- Approximation of geodesic & optimal transportation distances

# Conclusions

- Review of previous work on
  - the **Laplacian spectral functions, kernels, and distances**, defined by filtering the Laplacian spectrum and as a generalisation of the commute-time, bi-harmonic, diffusion, and wave kernel and distances
  - their **discretisation** according to a unified representation of the Laplace-Beltrami operator, which is "independent" of
    - the data dimensionality (surface, volume, nD data) and discretisation (mesh, point set) of the input domain
    - the selected Laplacian weights
  - the **computational aspects behind their evaluation**
    - approximation accuracy & stability
    - computational cost & storage overhead
    - use of input parameters & heuristics
  - their **main applications** to geometry processing and shape analysis

# Conclusions

- **Future work & possible collaborations**
  - Definition of **shape-aware functions for time-varying & multi-dimensional data** (eg., graphs, videos);
  - Analysis of the **constraints on the filter** in order to define "optimal" spectral kernels and distances for applications in geometry processing and shape analysis
  - Application/specialisation of the spectral basis functions to
    - **shape analysis**
    - **definition of shape-aware functional spaces** where we approximate signal or solve PDEs
- **Course material & Papers**
  - `http://pers.ge.imati.cnr.it/patane/SGP2019/Course.html`
  - `http://pers.ge.imati.cnr.it/patane/Home.html`

# References


# Acknowledgments


- **People**
  - SGP2019 Conference & Graduate School Chairs: Marcel Campen & Sylvain Lefebvre
  - Shape and Semantics Modelling Group, CNR-IMATI, Italy
- **Projects**
  - H2020 ERC-AdG CHANGE
  - IMAGE-FUSION, Biannual Project funded by Regione Liguria & EU FESR
- **Shapes**
  - AIM@SHAPE Repository
  - SHREC2010/2016 data sets
- **Contact:** `patane@ge.imati.cnr.it`