Analysis of 3D Shapes (IN2238)

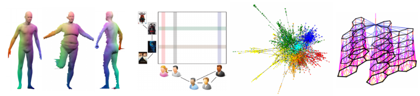
Frank R. Schmidt
Matthias Vestner

Summer Semester 2017

Short Course

Geometric deep learning on graphs and manifolds

Going beyond Euclidean data



June 30th, 14-17
July 7th, 14-16:30
MI HS 3

Michael Bronstein

- Associate Professor USI Lugano
- Associate Professor Tel Aviv University
- Principal Engineer at the Intel Perceptual Computing Group
- authored the first book on non-rigid shape analysis
- www.inf.usi.ch/bronstein/

---

10. Isometries, Rigid Alignment

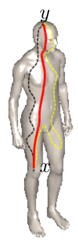
Isometries

Geodesic distance

Let $\mathcal{M}$ be a manifold. We define the geodesic distance between two points $x, y \in \mathcal{M}$ as

$$d_{\mathcal{M}}(x,y) = \inf_c \{\text{length}(c) | c : [0,1] \to \mathcal{M},\, c(0) = x,\, c(1) = y\}.$$

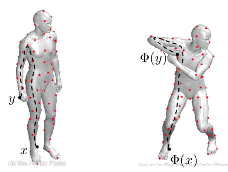
- For the manifolds we consider (compact) there exists a minimizer (not nec. unique)
- Using the first fundamental form the length of curves can be measured in the parameter domain
- every submanifold comes with a natural metric induced by the first fundamental form
- we ommit the proof that d is actually a metric



Isometries

A mapping $\Phi : \mathcal{M} \to \mathcal{N}$ between two shapes (manifolds) is an isometry if

$$d_{\mathcal{M}}(x,y) = d_{\mathcal{N}}(\Phi(x), \Phi(y)) \quad \text{for all points } x, y \in \mathcal{M}.$$

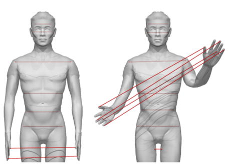


If such a mapping exists $\mathcal{M}$ and $\mathcal{N}$ are called isometric.
Many shape matching approaches assume that the shapes to be matched are (nearly) isometric. The task then becomes to find the (almost-)isometry Φ.

Intrinsic symmetry

Most of the shapes we consider come with an intrinsic symmetry $S : \mathcal{M} \to \mathcal{M}$, such that

$$d_{\mathcal{M}}(x,y) = d_{\mathcal{M}}(S(x), S(y)), \quad \forall x, y \in \mathcal{M}, S \neq \text{id}\,.$$

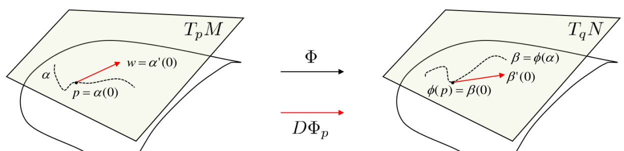


As a consequence isometries are not unique. Let $\Phi : \mathcal{M} \to \mathcal{N}$ be an isometry and $S : \mathcal{M} \to \mathcal{M}$ be an intrinsic symmetry. Then $\Phi \circ S^{-1}$ is also an isometry:

$$d_{\mathcal{M}}(x,y) = d_{\mathcal{M}}(S^{-1}(x), S^{-1}(y)) = d_{\mathcal{N}}(\Phi \circ S^{-1}(x), \Phi \circ S^{-1}(y))$$

Push Forward

We can define the differential of a map between manifolds as we did with coordinate maps. Given a map $\Phi : \mathcal{M} \to \mathcal{N}$ the differential is a linear map $D\Phi_p : T_p\mathcal{M} \to T_q\mathcal{N}$ which maps tangent vectors at $p \in \mathcal{M}$ to tangent vectors at $q = \Phi(p) \in \mathcal{N}$.



$$D\Phi_p(w) = \beta'(0)$$

also called **push-forward**

Equivalent definition

A diffeomorphism $\Phi : \mathcal{M} \to \mathcal{N}$ is an isometry iff it preserves angles:

$$\langle v, w \rangle_{T_p\mathcal{M}} = \langle D\Phi_p v, D\Phi_p w \rangle_{T_q\mathcal{N}}$$

Proof (only one direction): Let $c : [0,1] \to \mathcal{M}$ be a shortest curve connecting $p \in \mathcal{M}$ and $q \in \mathcal{M}$: $d_{\mathcal{M}}(p,q) = L(c) = \int_0^1 \|\dot{c}(t)\|\, dt$. Then the curve $d = \Phi \circ c : [0,1] \to \mathcal{N}$ has length

$$L(d) = \int_0^1 \left\| \frac{d}{dt}(\Phi \circ c(t)) \right\| dt = \int_0^1 \|D\Phi_{c(t)}\dot{c}(t))\|\, dt = \int_0^1 \|\dot{c}(t)\|\, dt = L(c)$$

Since there is no shorter curve connecting $\Phi(p)$ and $\Phi(q)$ (why?), it follows $d_{\mathcal{N}}(p,q) = d_{\mathcal{N}}(\Phi(p), \Phi(q))$.
Reason: the length of *all* (not only shortest) curves are preserved.

Preserving intrinsics

Let $\Phi : \mathcal{M} \to \mathcal{N}$ be an isometry, $p \in \mathcal{M}$ and $x_{\mathcal{M}} : \mathbb{R}^2 \supset U \to \mathcal{M}$ be a coordinate map of a neighborhood V of p. Then $x_{\mathcal{N}} := \Phi \circ x_{\mathcal{M}} : U \to \mathcal{N}$ is a coordinate map of the neighborhood $\Phi(V) \subset \mathcal{N}$ of $\Phi(p)$. For the first fundamental forms $g_{\mathcal{M}} : U \to \mathbb{R}^{2\times 2}$, $g_{\mathcal{N}} : U \to \mathbb{R}^{2\times 2}$ we observe:

$$\begin{aligned} g_{\mathcal{N}}(u) &= \langle Dx_{\mathcal{N}}(u), Dx_{\mathcal{N}}(u) \rangle \\ &= \langle D\Phi_{x_{\mathcal{M}}(u)} Dx_{\mathcal{M}}(u), D\Phi_{x_{\mathcal{M}}(u)} Dx_{\mathcal{M}}(u) \rangle = g_{\mathcal{M}}(u) \end{aligned}$$

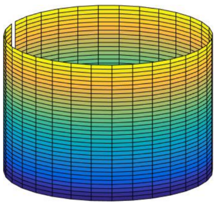
Thus **intrinsic properties of the shapes are preserved under isometries**.

Example (Cylinder)

By choosing $U = (0, \pi) \times \mathbb{R}$, $x_{\mathcal{M}}(\alpha, r) = (\alpha, r)$ and $x_{\mathcal{N}}(\alpha, r) = \begin{pmatrix} \cos(\alpha) \\ \sin(\alpha) \\ r \end{pmatrix}$ and observing

$$g_{\mathcal{M}}(\alpha, r) = g_{\mathcal{N}}(\alpha, r) \qquad \text{for all } (\alpha, r) \in U$$

we know that the stripe U of $\mathbb{R}^2$ and the (sliced!) cylinder are isometric.



Eigendecompositions

Eigenvalues and -vectors

In the next weeks (starting today) we will make a lot use of the concept of eigenvalues and eigenvectors. Let us briefly revise the definitions and fundamental properties of eigenvalues and -vectors.

Definition. Given a matrix $\mathbf{A} \in \mathbb{C}^{n\times n}$. If a pair $\lambda \in \mathbb{C}, 0 \neq \mathbf{v} \in \mathbb{C}^n$ satisfies

$$\mathbf{A}\mathbf{v} = \lambda\mathbf{v}$$

we call $\mathbf{v}$ an **eigenvector** of $\mathbf{A}$ and λ its corresponding **eigenvalue**.

The n Eigenvalues $\{\lambda_1, \ldots \lambda_n\}$ are the roots of $\mathbf{A}$'s **characteristic polynomial**

$$p_{\mathbf{A}}(\lambda) = \det(\mathbf{A} - \lambda\mathbf{I}) = \Pi_{i=1}^n(\lambda - \lambda_i)$$

The fundamental theorem of algebra guarantees that this polynomial has exactly n roots (counted by multiplicity).

Ugly facts about eigenvectors

Even for real matrices $\mathbf{A} \in \mathbb{R}^{n\times n}$ the **spectrum (set of eigenvalues)** can contain complex values:

$$\mathbf{A} = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}$$

$$p_{\mathbf{A}}(\lambda) = (\lambda^2 + 1) = (\lambda - i)(\lambda + i)$$

If $\mathbf{v}$ is a corresponding eigenvector of λ, then also every vector $\mathbf{w} = \alpha\mathbf{v}$ ($\alpha \neq 0$) is an eigenvector to λ:

$$\mathbf{A}\mathbf{w} = \mathbf{A}\alpha\mathbf{v} = \alpha\mathbf{A}\mathbf{v} = \alpha\lambda\mathbf{v} = \lambda\alpha\mathbf{v} = \lambda\mathbf{w}$$

If $\mathbf{v}_i, \mathbf{v}_j$ are eigenvectors to $\lambda_i = \lambda_j$ then every linear combination $\mathbf{w} = \alpha_i\mathbf{v}_i + \alpha_j\mathbf{v}_j$ of them is also an eigenvector (same holds for eigenvalues with higher multiplicity):

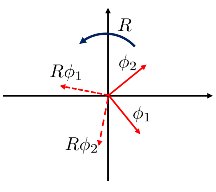
$$\mathbf{A}\mathbf{w} = \mathbf{A}(\alpha_i\mathbf{v}_i + \alpha_j\mathbf{v}_j) = \alpha_i\mathbf{A}\mathbf{v}_i + \alpha_j\mathbf{A}\mathbf{v}_j = \alpha_i\lambda_i\mathbf{v}_i + \alpha_j\lambda_j\mathbf{v}_j = \lambda_i\mathbf{w}$$

Symmetric matrices

Theorem. *Given a symmetric matrix $\mathbf{A} \in \mathbb{R}^{n\times n}$ all its eigenvalues are real and the corresponding eigenvectors can be chosen, such that*

$$\langle \mathbf{v}_i, \mathbf{v}_j \rangle = \begin{cases} 0 & i \neq j \\ 1 & i = j \end{cases}$$

Notice that even under the assumption of **orthonormality**, the choice of eigenfunctions is not unique.



Eigendecomposition 1

If the eigenvectors $\mathbf{v}_1, \ldots, \mathbf{v}_n$ of A span $\mathbb{R}^n$, we can **decompose** $\mathbf{A}$ as

$$\mathbf{A} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^{-1}$$

with

$$\mathbf{V} = \begin{pmatrix} | & & | \\ v_1 & \cdots & v_n \\ | & & | \end{pmatrix} \qquad \mathbf{\Lambda} = \begin{pmatrix} \lambda_1 & \ldots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \ldots & \lambda_n \end{pmatrix}$$

Let $\mathbf{x} = \sum \alpha_i \mathbf{v}_i = \mathbf{V}\alpha$ be an arbitrary vector. Then

$$\mathbf{V}\mathbf{\Lambda}\mathbf{V}^{-1}\mathbf{x} = \mathbf{V}\mathbf{\Lambda}\begin{pmatrix} \alpha_1 \\ \vdots \\ \alpha_n \end{pmatrix} = \mathbf{V}\begin{pmatrix} \lambda_1\alpha_1 \\ \vdots \\ \lambda_n\alpha_n \end{pmatrix} = \sum_i \alpha_i\lambda_i\mathbf{v}_i = \sum_i \alpha_i\mathbf{A}\mathbf{v}_i = \mathbf{A}\mathbf{x}$$

If the eigenvectors are orthonormal, we have $\mathbf{V}^{-1} = \mathbf{V}^T$.

SPD matrices

A symmetric matrix $\mathbf{A}$ is positive definit iff all its eigenvalues are positive.
Let $\{v_i\}$ be orthonormal eigenvectors of $\mathbf{A}$ and $0 \neq x = \sum_i \alpha_i \mathbf{v}_i$ an arbitrary vector.
Then

$$\begin{aligned} x^T\mathbf{A}x &= (\sum_i \alpha_i \mathbf{v}_i)^T \mathbf{A} (\sum_j \alpha_j \mathbf{v}_j) \\ &= (\sum_i \alpha_i \mathbf{v}_i)^T (\sum_j \alpha_j \lambda_j \mathbf{v}_j) \\ &= (\sum_i \sum_j \alpha_i \alpha_j \lambda_j \mathbf{v}_i^T \mathbf{v}_j) \\ &= \sum_j \alpha_j^2 \lambda_j \end{aligned}$$

which is positive iff all λ_j are positive.

Eigenvectors and optimization

There is a fundamental relation between the eigenvectors of a spd matrix $\mathbf{A}$ and the optimization problem

$$\max \quad \mathbf{x}^T\mathbf{A}\mathbf{x} \qquad \text{s.t.} \quad \langle \mathbf{x}, \mathbf{x} \rangle = 1$$

Let $\{\mathbf{v}_i\}$ be orthonormal eigenvectors of $\mathbf{A}$ (ordered from big to small) and $\mathbf{x} = \sum_i \alpha_i \mathbf{v}_i$, satisfying $\langle \mathbf{x}, \mathbf{x} \rangle = 1$. First we observe that

$$\langle \mathbf{x}, \mathbf{x} \rangle = \langle \sum_i \alpha_i \mathbf{v}_i, \sum_j \alpha_j \mathbf{v}_j \rangle = \sum_i \alpha_i^2 = 1$$

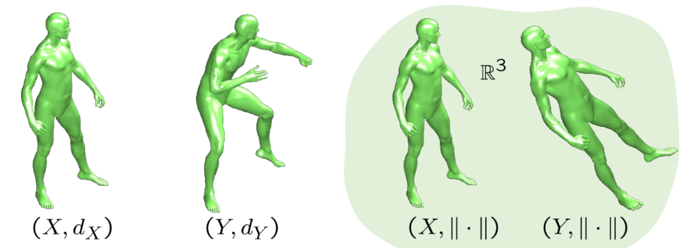
For the objective we get:

$$\mathbf{x}^T\mathbf{A}\mathbf{x} = (\sum_i \alpha_i \mathbf{v}_i)^T A \sum_j \alpha_j \mathbf{v}_j = \sum_i \alpha_i^2 \lambda_i \leqslant \lambda_1 = \mathbf{v}_1^T \mathbf{A} \mathbf{v}_1$$

Thus maximizing the quadratic function is equivalent to finding the **principal eigenvector** $\mathbf{v}_1$.

Rigid Alignment

Euclidean isometry



Intrinsic isometry **Euclidean isometry**

Two different metric spaces **Part of the same metric space**

Iterative closest point

Given two shapes X and Y find the degree of their **incongruence**. Compare X and Y as subsets of the Euclidean space $\mathbb{R}^3$. Find the best rigid motion $(\mathbf{R}, \mathbf{t})$ - *i.e.* $\mathbf{R} \in \mathbb{R}^{3\times3} : \mathbf{R}^T\mathbf{R} = I, \mathbf{t} \in \mathbb{R}^3$ - bringing $Y' = \mathbf{R}Y + t$ **as close as possible** to X:

$$d_{\text{ICP}}(X, Y) = \min_{\mathbf{R},\mathbf{t}} d(\mathbf{R}Y + \mathbf{t}, X)$$

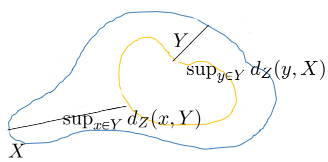
Minimum: extrinsic similarity of X and Y
Minimizer: best rigid alignment between X and Y
ICP is a **family of algorithms** differing in

- the choice of the shape-to-shape distance d
- the choice of the numerical minimization algorithm

Shape-to-shape distance

The Hausdorff distance d_H between two subsets $X, Y \subset Z$ of a metric space (Z, d_Z) is defined as

$$d_H(X, Y) = \max\{\sup_{y\in Y} d_Z(y, X), \sup_{x\in X} d_Z(x, Y)\}$$



Non-symmetric version: $\sup_{y\in Y} d_Z(y, X)$
The "maximum"-version is sensitive to outliers. A variant is to use some kind of average: $d(X, Y) = \int_Y d_Z^2(y, X)dn$.

ICP algorithm

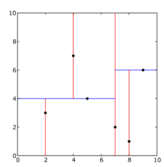
Given are a point cloud $X = \{x_i\}$, and an either discrete or continuous Y.

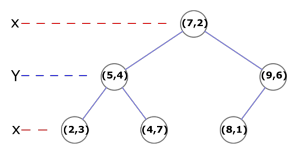
- Initialize Y
- Until convergence
  - Findest the best point-to-point correspondence $y_i = \text{argmin}_{y\in Y} \|x_i - y\|$
  - Minimize the misalignment between corresponding points: $(\mathbf{R}, \mathbf{t}) = \text{argmin}_{\mathbf{R},\mathbf{t}} \sum_i \|(Rx_i + t) - y_i\|^2$
  - Update $Y = \mathbf{R}Y + \mathbf{t}$

KD Tree

To perform the nearest neighbor search it is beneficial to make use of efficient datastructures such as k-d trees.

- binary tree
- each non-leaf node encodes a hyperplane
- Construction in $O(n \log n)$
- Average query time $O(\log n)$
- Course of dimensionality: in efficient for $n < 2^k$





Optimal Rigid alignment 1

For simplicity we assume that X and Y are centered at the origin: $\sum_i x_i = \sum y_i = 0$. Thus the second term in

$$\sum_i \|\mathbf{R}x_i - t - y_i\|^2 = \sum_i \|\mathbf{R}x_i - y_i\|^2 - 2\langle t, \sum_i (\mathbf{R}x_i - y_i)\rangle + n\,\|t\|^2$$

vanishes and it follows $t = 0$ (or in general $t = \sum_i x_i - \sum_i y_i$).
It remains to find the orthogonal matrix $\mathbf{R}$ minmimizing

$$\sum_i \|\mathbf{R}x_i\| - 2\sum_i y_i^T \mathbf{R}x_i + \sum_i \|y_i\|^2 = \sum_i \|x_i\| - 2\sum_i y_i^T \mathbf{R}x_i + \sum_i \|y_i\|^2$$

The first and the last term are independent of $\mathbf{R}$, we thus have to **maximize** $\sum_i y_i^T \mathbf{R} x_i$.

Optimal Rigid alignment 2

We want to maximize

$$\sum_i y_i^T \mathbf{R} x_i = \sum_i \mathrm{tr}(\mathbf{R}x_i y_i^T) = \mathrm{tr}(\mathbf{R}^T \sum_i y_i x_i^T) = \mathrm{tr}(\mathbf{R}^T\mathbf{M})$$

If $\mathbf{M}$ has full rank, we can construct

$$S = \sqrt{\mathbf{M}^T\mathbf{M}} \quad \mathbf{U} = \mathbf{M}\mathbf{S}^{-1}$$

where the square root of a symmetric positive definit (spd) matrix $\mathbf{A} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^T$ is defined as

$$\sqrt{\mathbf{A}} = \mathbf{V}\sqrt{\mathbf{\Lambda}}\mathbf{V}^T$$

such that it holds $\sqrt{\mathbf{A}}^T\sqrt{\mathbf{A}} = \mathbf{A}$.
One can show that the optimal choice is $\mathbf{R} = \mathbf{U}$.

Optimal Rigid alignment 3

We want to maximize

$$\sum_i y_i^T \mathbf{R} x_i = \sum_i \mathrm{tr}(\mathbf{R}x_i y_i^T) = \mathrm{tr}(\mathbf{R}^T \sum_i y_i x_i^T) = \mathrm{tr}(\mathbf{R}^T\mathbf{M})$$

If $\mathbf{M}$ has full rank, we can construct

$$S = \sqrt{\mathbf{M}^T\mathbf{M}} \quad \mathbf{U} = \mathbf{M}\mathbf{S}^{-1}$$

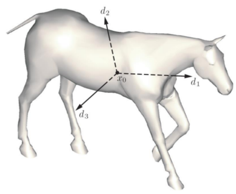
such that $\mathbf{M} = \mathbf{U}\mathbf{S}$ is a decomposition of $\mathbf{M}$ in an orthogonal matrix $\mathbf{U}$ and a positive definit matrix $\mathbf{S}$, thus $\mathrm{tr}(\mathbf{R}^T\mathbf{M}) = \mathrm{tr}(\mathbf{R}^T\mathbf{U}\mathbf{S})$. The orthognal matrix $\mathbf{R}$ maximizing this term equals $\mathbf{U}$.
proof: Let $\mathbf{S} = \sum_i \lambda_i v_i v_i^T$ (eigenvalues and eigenvectors). Then

$$\mathrm{tr}(\mathbf{R}^T\mathbf{U}\mathbf{S}) = \mathrm{tr}(\mathbf{R}^T\mathbf{U}\sum_i \lambda_i v_i v_i^T) = \mathrm{tr}(\sum_i \lambda_i (\mathbf{R}^T\mathbf{U}v_i)^T v_i) \leqslant \mathrm{tr}(\sum_i \lambda_i v_i v_i^T)$$

Drawbacks of ICP

Although a very simpe algorithm, ICP relies on a good initialization.

As an alternative to ICP and/or an initializier one could bring the shapes into a "canonical" pose. This canonical pose can be found using **principal component analysis (PCA)**.



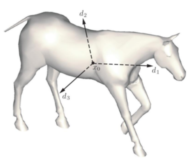
Principal component analysis 1

Given a pointset $X = \{x_i\}_{i=1}^n$ we want to align it with a rigid motion $X \to \mathbf{R}(X - \mathbf{t})$ such that:

- the center of mass lies at the origin
- the direction in which the pointset expands the most should be the x_1-axis and so forth

By translating the center of mass of the point set to the origin, the first goal is easily achieved: $\mathbf{t} = \sum_i x_i$

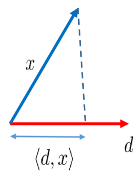
Now that the pointset is centered at the origin it remains to find the orthogonal matrix $\mathbf{R}$ aligning it with the axis. Assume we know the three **principal components** d_1, d_2, d_3 with which it is aligned before the rotation, then choosing $\mathbf{R} = \begin{pmatrix} | & | & | \\ d_1 & d_2 & d_3 \\ | & | & | \end{pmatrix}^T$ aligns it with the axis.



Principal component analysis 2

We are looking for a direction $\mathbf{d}$ ($\|\mathbf{d}\| = 1$) maximizing

$$\sum_{i=1}^n \langle \mathbf{d}, \mathbf{x}_i\rangle^2 = \sum_{i=1}^n \mathbf{d}^T\mathbf{x}_i\mathbf{x}_i^T\mathbf{d}$$



The **covariance matrix** $\mathbf{\Sigma}_X$ is defined as $\mathbf{\Sigma}_X = \sum_{i=1}^n \mathbf{x}_i\mathbf{x}_i^T$ and is spd. So we can rewrite the objective as

$$\begin{aligned} \max \quad & \mathbf{d}^T\mathbf{\Sigma}_X\mathbf{d} \\ \text{s.t.} \quad & \langle \mathbf{d}, \mathbf{d}\rangle = 1 \end{aligned}$$

We have seen that this objective is maximized by the principal eigenvector of $\mathbf{\Sigma}_X$. d_2 and d_3 are then the eigenvectors corresponding to second and third eigenvalue (when ordered by magnitude).