Riemannian Geometry for Computational Anatomy

Introducing the LDDMM framework.

Jean Feydy

October 10, 2017

Écoles Normales Supérieures de Paris et Paris-Saclay

Some information

Jean Feydy (2016-2019) :

- PhD student under the supervision of Alain Trouvé.
- Caïman at the ENS.

Some information

Jean Feydy (2016-2019) :

- PhD student under the supervision of Alain Trouvé.
- Caïman at the ENS.

Notes for this talk are available online (in French) :

`www.math.ens.fr/~feydy/Teaching/`

- *Culture Mathématique*, chap. 4-6.
- *Introduction à la Géométrie Riemannienne par l'Étude des Espaces de Formes.*

Introduction

How do we decompose variability ?

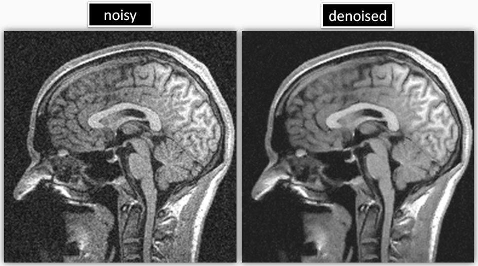


Figure 1: Image denoising, from [2]. .

Research in **Image Processing** :

- Signal analysis.

How do we decompose variability ?

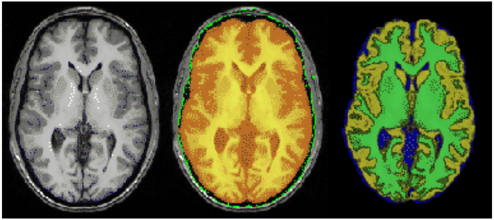


Figure 1: Brain segmentation, from [7]. .

Research in **Image Processing** :

- Signal analysis.
- Segmentation.

How do we decompose variability ?

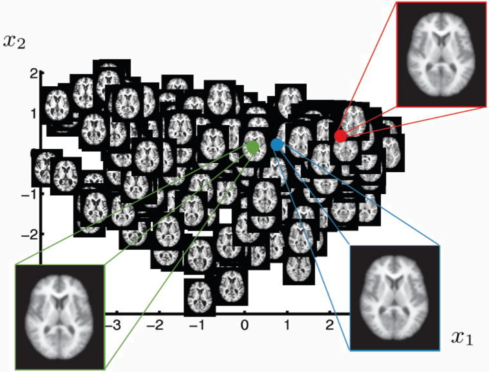


Figure 1: Brain database, from [4].

Research in **Image Processing** :

- Signal analysis.
- Segmentation.
- **Population Analysis.**

How do we decompose variability ?

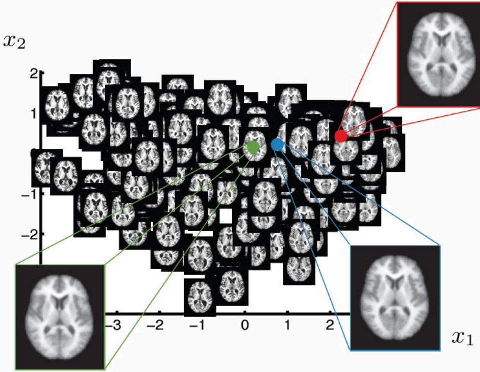


Figure 1: Brain database, from [4].

Research in **Image Processing** :

- Signal analysis.
- Segmentation.
- **Population Analysis.**

We need appropriate representations.

Orthonormal image transforms

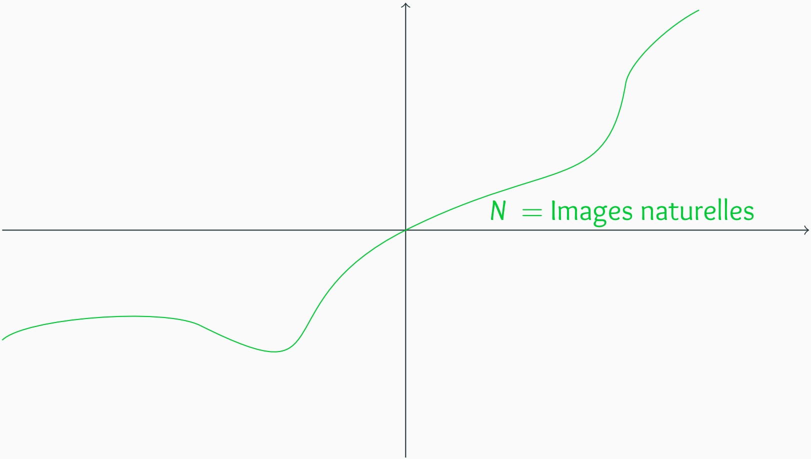


Figure 2: A well-chosen orthonormal basis (aka. transform) of $\mathbb{R}^{W \times H}$ can help us to formulate efficient signal processing algorithms.

Orthonormal image transforms

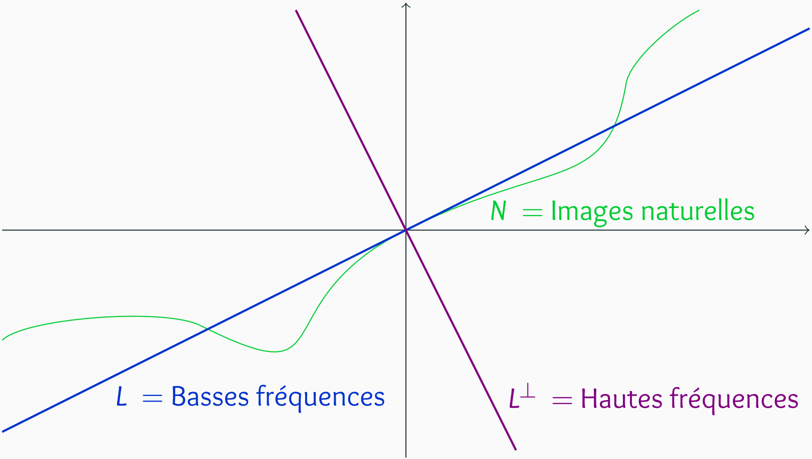


Figure 2: A well-chosen orthonormal basis (aka. transform) of $\mathbb{R}^{W \times H}$ can help us to formulate efficient signal processing algorithms.

Orthonormal image transforms

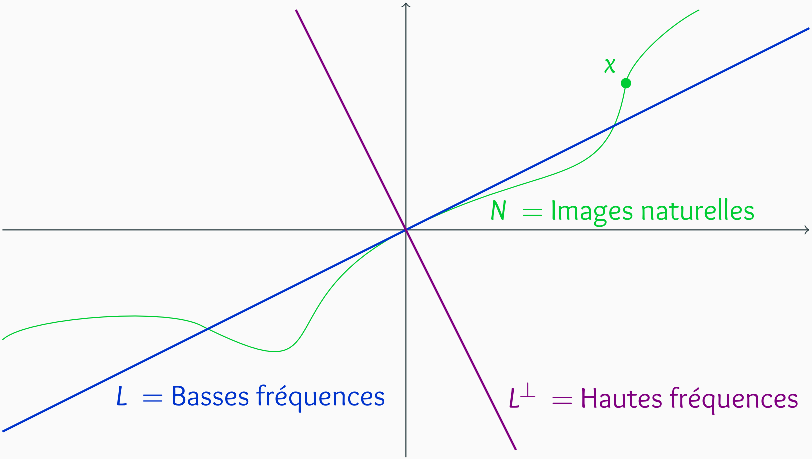


Figure 2: A well-chosen orthonormal basis (aka. transform) of $\mathbb{R}^{W \times H}$ can help us to formulate efficient signal processing algorithms.

Orthonormal image transforms

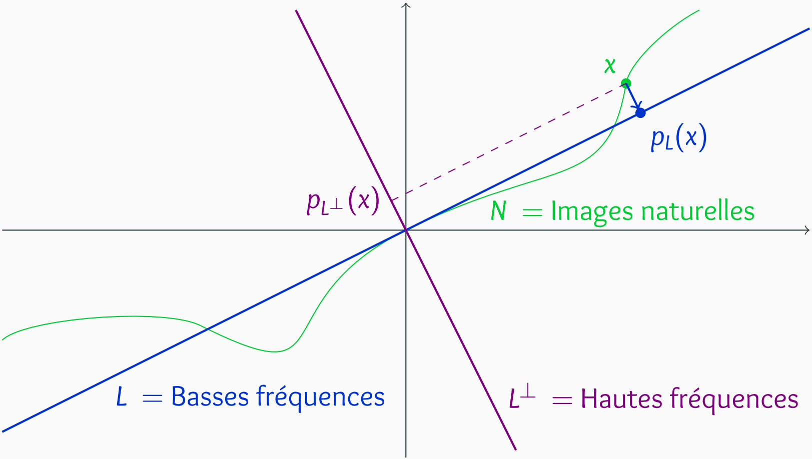
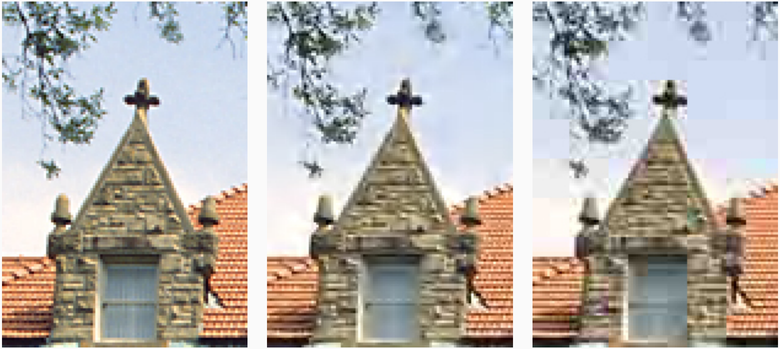


Figure 2: A well-chosen orthonormal basis (aka. transform) of $\mathbb{R}^{W\times H}$ can help us to formulate efficient signal processing algorithms.

JPEG2000, JPEG : Wavelets, Blockwise (high + low) frequencies



(a) Original image. **(b)** JPEG2000, 20 : 1. **(c)** JPEG, 20 : 1.

Figure 3: Taken from `www.photozone.de`.

Convolutional Neural Networks : Texture + Structure

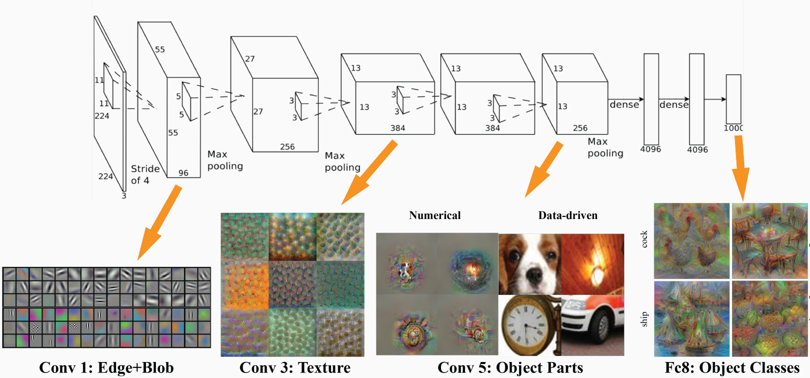


Figure 4: CNN visualization, from `vision03.csail.mit.edu/cnn_art/`.

Figure 5: Reference image.

Figure 5: With a transferred texture component. [6]

Convolutional Neural Networks : Texture + Structure

Figure 5: With a transferred texture component. [6]

Convolutional Neural Networks : Texture + Structure

Figure 5: With a transferred texture component. [6]

Figure 5: With a transferred texture component. [6]

Figure 5: With a transferred texture component. [6]

Convolutional Neural Networks : Texture-invariant Classification

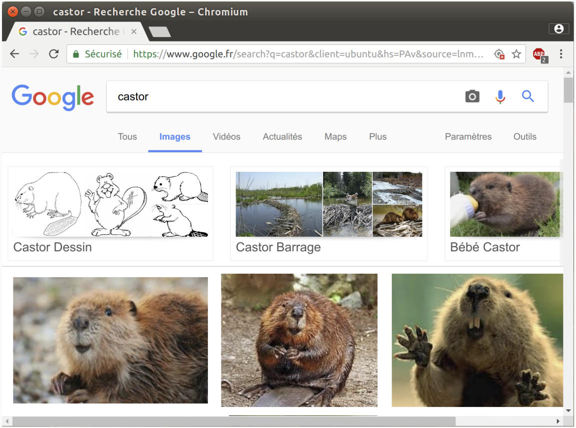


Figure 6: CNNs allow tech companies to group together photos and sketches of beavers.

How do we handle intra-class variability ?

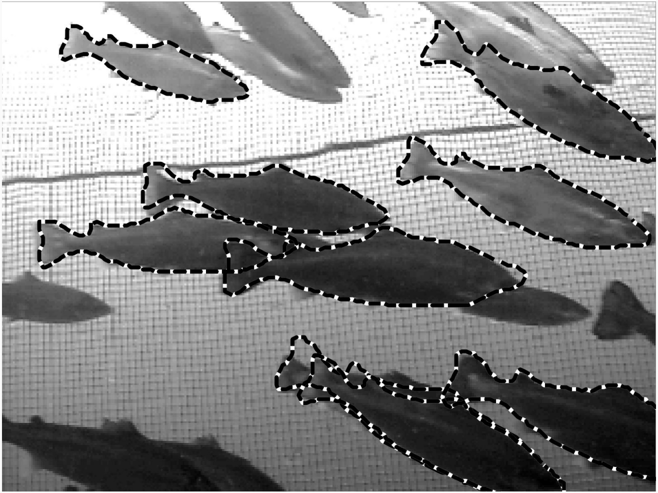


Figure 7: Silhouettes segmented from a fishing net. [3]

Rigid Body Analysis

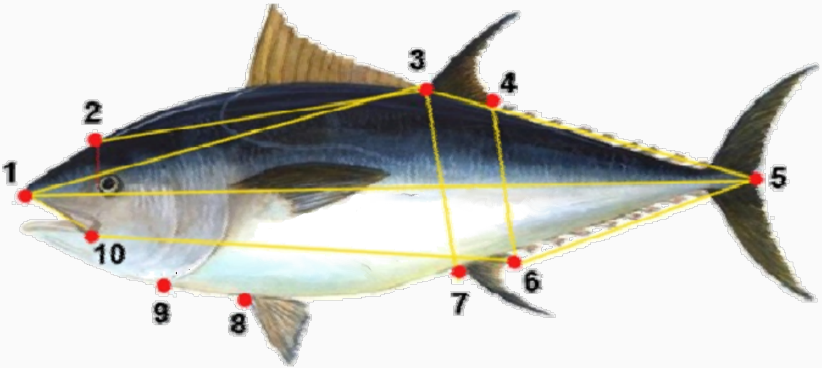


Figure 8: Anatomical landmarks on a tuna fish. [1]

Mathematical formulation

Let $X, Y \in \mathbb{R}^{M \times D}$ be two labeled point clouds.
Let $S_{\tau,\upsilon}$ denote the **rigid**-body transformation of parameters τ (translation) and υ (rotation + scaling).
Then, try to find

$$\tau_0, \upsilon_0 = \arg\min_{\tau,\upsilon} \| S_{\tau,\upsilon}(X) - Y \|_2^2 \tag{1}$$

$$= \arg\min_{\tau,\upsilon} \sum_{m=1}^{M} | \upsilon \cdot x^m + \tau - y^m |^2 . \tag{2}$$

Position, Scale and Orientation

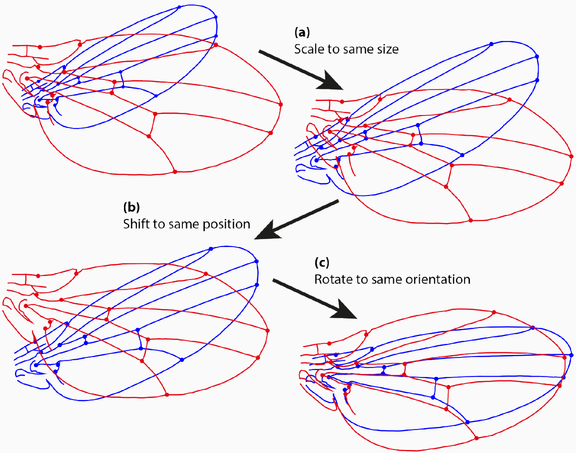


Figure 9: Matching the blue wing on the red one. (Wikipedia)

Pros and cons of Rigid body analysis

Pros :

- Simple and robust
- Parameters make sense
- Miracle results for populations of triangles (Kendall, 1984)

Pros and cons of Rigid body analysis

Pros :

- Simple and robust
- Parameters make sense
- Miracle results for populations of triangles (Kendall, 1984)

Cons :

- Max. number of 4 or 6 explicative parameters
- Unable to capture subtle shape deformations

Pros and cons of Rigid body analysis

Pros :

- Simple and robust
- Parameters make sense
- Miracle results for populations of triangles (Kendall, 1984)

Cons :

- Max. number of 4 or 6 explicative parameters
- Unable to capture subtle shape deformations

This model is a standard pre-processing tool.
However, it is too limited to allow in-detail analysis.

Optimal Transport

Dynamic formulation

Let : $(x^1, \ldots, x^M)$, $(y^1, \ldots, y^M)$ be two point clouds in $\mathbb{R}^D$.

Find a collection of paths $\gamma^m : t \in [0, 1] \mapsto \gamma_t^m$,
a permutation $\sigma : [\![1, M]\!] \to [\![1, M]\!]$ such that

$$\forall\, m, \quad \gamma_0^m = x^m \text{ and } \gamma_1^m = y^{\sigma(m)}, \tag{3}$$

minimizing

$$\ell^2(\gamma) = \sum_{m=1}^{M} \int_{t=0}^{1} \|\dot{\gamma}_t^m\|^2 \, \mathrm{d}t. \tag{4}$$

γ is the optimal transport path between the two shapes

$$X \xrightarrow{\gamma} Y. \tag{5}$$

Dynamic formulation

Let : $(x^1, \dots, x^M)$, $(y^1, \dots, y^M)$ be two point clouds in $\mathbb{R}^D$.

Find a collection of paths $\gamma^m : t \in [0, 1] \mapsto \gamma_t^m$,
a permutation $\sigma : [\![1, M]\!] \to [\![1, M]\!]$ such that

$$\forall\, m, \quad \gamma_0^m = x^m \text{ and } \gamma_1^m = y^{\sigma(m)}, \tag{3}$$

minimizing

$$\ell^2(\gamma) = \sum_{m=1}^{M} \int_{t=0}^{1} \|\dot{\gamma}_t^m\|^2 \,\mathrm{d}t. \tag{4}$$

γ is the optimal transport path between the two shapes

$$X \xrightarrow{\gamma} Y. \tag{5}$$

Dynamic formulation

Let : $(x^1, \ldots, x^M)$, $(y^1, \ldots, y^M)$ be two point clouds in $\mathbb{R}^D$.

Find a collection of paths $\gamma^m : t \in [0, 1] \mapsto \gamma^m_t$,
a permutation $\sigma : [\![1, M]\!] \to [\![1, M]\!]$ such that

$$\forall\, m, \qquad \gamma^m_0 = x^m \ \text{ and } \ \gamma^m_1 = y^{\sigma(m)}, \tag{3}$$

minimizing

$$\ell^2(\gamma) = \sum_{m=1}^{M} \int_{t=0}^{1} \|\dot{\gamma}^m_t\|^2 \, \mathrm{d}t. \tag{4}$$

γ is the optimal transport path between the two shapes

$$X \xrightarrow{\gamma} Y. \tag{5}$$

Dynamic formulation

Let : $(x^1, \ldots, x^M)$, $(y^1, \ldots, y^M)$ be two point clouds in $\mathbb{R}^D$.

Find a collection of paths $\gamma^m : t \in [0, 1] \mapsto \gamma_t^m$,
a permutation $\sigma : [\![1, M]\!] \to [\![1, M]\!]$ such that

$$\forall\, m, \quad \gamma_0^m = x^m \text{ and } \gamma_1^m = y^{\sigma(m)}, \tag{3}$$

minimizing

$$\ell^2(\gamma) = \sum_{m=1}^{M} \int_{t=0}^{1} \|\dot{\gamma}_t^m\|^2 \, \mathrm{d}t. \tag{4}$$

γ is the optimal transport path between the two shapes

$$X \xrightarrow{\gamma} Y. \tag{5}$$

If we relabel the unit masses $(x^1, \ldots, x^M)$ and $(y^1, \ldots, y^M)$, find a **permutation** $\sigma : [\![1, M]\!] \to [\![1, M]\!]$ minimizing

$$C^{X,Y}(\sigma) \;=\; \sum_{m=1}^{M} \left\| x^m - y^{\sigma(m)} \right\|^2 . \tag{6}$$

σ is an optimal labeling.

Image matching as a mass-carrying problem

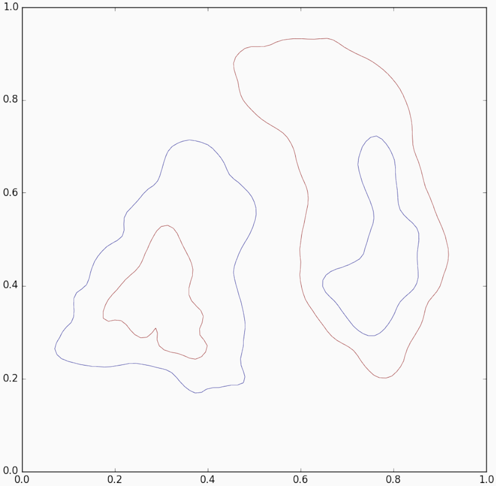


Figure 10: Optimal transport between two curves seen as mass distributions : from a déblai to a remblai.

Image matching as a mass-carrying problem

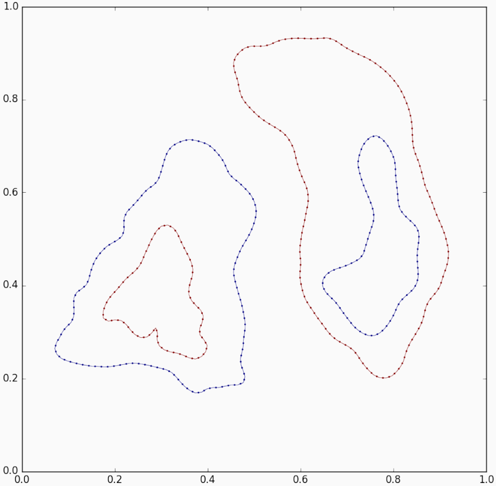


Figure 10: Optimal transport between two curves seen as mass distributions : from a déblai to a remblai.

Image matching as a mass-carrying problem

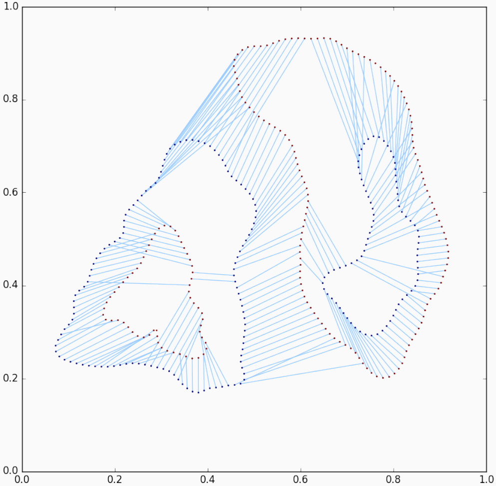


Figure 10: Optimal transport between two curves seen as mass distributions : from a déblai to a remblai.

Pros and cons of Optimal Transport

Pros :

- Well-posed, convex problem
- Global and precise matchings
- Light-speed numerical solvers at hand (Cuturi, 2013)

Cons :

- Discards topology : **tears** shapes apart

This model is mathematically and numerically appealing.
However, it does not provide any smoothness guarantee.

Pros and cons of Optimal Transport

Pros :

- Well-posed, convex problem
- Global and precise matchings
- Light-speed numerical solvers at hand (Cuturi, 2013)

Cons :

- Discards topology : **tears** shapes apart

This model is mathematically and numerically appealing.
However, it does not provide any smoothness guarantee.

Pros and cons of Optimal Transport

Pros :

- Well-posed, convex problem
- Global and precise matchings
- Light-speed numerical solvers at hand (Cuturi, 2013)

Cons :

- Discards topology : **tears** shapes apart

This model is mathematically and numerically appealing.
However, it does not provide any smoothness guarantee.

Can we build a rich and practical model
for smooth deformations ?

The LDDMM framework

Spoiler alert : yes indeed, but it won't be *convex* anymore

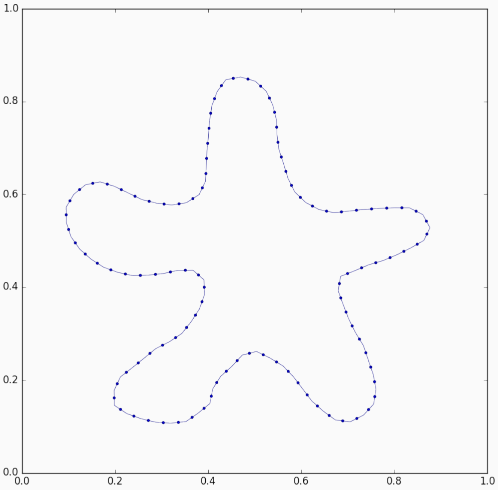


Figure 11: Source.

Spoiler alert : yes indeed, but it won't be *convex* anymore

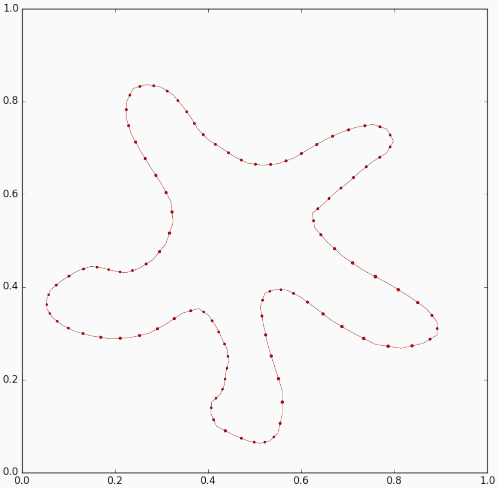


Figure 11: Target.

Spoiler alert : yes indeed, but it won't be *convex* anymore

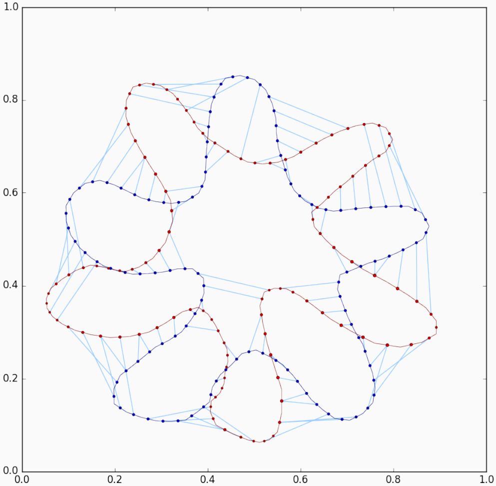


Figure 11: OT matching.

Spoiler alert : yes indeed, but it won't be *convex* anymore

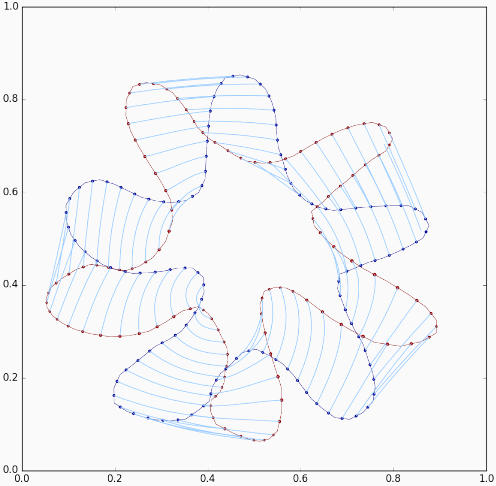


Figure 11: LDDMM matching.

The LDDMM framework

Regularized transport : a Riemannian problem

Static regularization : a first attempt

A naive way to regularize transport :

Find σ : $[\![1, M]\!] \to [\![1, M]\!]$ minimizing

$$C_k^{X,Y}(\sigma) = \underbrace{\sum_m \left\| x^m - y^{\sigma(m)} \right\|^2}_{\text{Displacement cost}} + \underbrace{\sum_{m,m'} k(x^m, x^{m'}) \cdot \left\| y^{\sigma(m)} - y^{\sigma(m')} \right\|^2}_{\text{Regularization cost}}, \tag{7}$$

with $k(x, y)$ a kernel neighborhood function.

Static regularization : symmetry without continuity

Find a permutation $\sigma : [\![1, M]\!] \to [\![1, M]\!]$ minimizing

$$C^{X,Y}_{k,\mathrm{sym}}(\sigma) = \underbrace{\sum_m \left\| x^m - y^{\sigma(m)} \right\|^2}_{\text{Displacement cost}} + \underbrace{\frac{1}{2} \sum_{m,m'} k(x^m, x^{m'}) \cdot \left\| y^{\sigma(m)} - y^{\sigma(m')} \right\|^2}_{X \to Y \text{ regularization cost}}$$
$$+ \underbrace{\frac{1}{2} \sum_{m,m'} k(y^m, y^{m'}) \cdot \left\| x^{\sigma^{-1}(m)} - x^{\sigma^{-1}(m')} \right\|^2}_{Y \to X \text{ regularization cost}} .$$

This cost is symmetric, but does not handle properly the shapes between X and Y.

Going back to the kinematic transportation

Find a collection of paths γ^m from X to Y minimizing

$$C_k(\gamma) = \int_0^1 \Bigg[\underbrace{\sum_m \|\dot{\gamma}_t^m\|^2}_{\text{Displacement cost}} + \underbrace{\sum_{m,m'} k(\gamma_t^m, \gamma_t^{m'}) \cdot \left\|\dot{\gamma}_t^m - \dot{\gamma}_t^{m'}\right\|^2}_{\text{Regularization cost}} \Bigg] \mathrm{d}t.$$

Particles will move optimally if they are :

- lazy
- gregarious wrt. their k-neighbors

With $\gamma_t = (\gamma_t^1, \dots, \gamma_t^M) \in \mathbb{R}^{M \times D}$, we can write

$$C_k(\gamma) \;=\; \int_0^1 \dot{\gamma_t}^\intercal g_{\gamma_t} \dot{\gamma_t} \mathrm{d}t. \tag{8}$$

Optimal deformations are geodesics on the space of landmarks $\mathbb{R}^{M \times D}$ endowed with a Riemannian metric g_q :

$$\begin{aligned} \frac{\big(\mathrm{d}_g(q \rightarrow q + v \cdot \mathrm{d}t)\big)^2}{\mathrm{d}t} &= \sum_m \|v^m\|^2 + \sum_{m,m'} k(q^m, q^{m'}) \cdot \left\| v^m - v^{m'} \right\|^2 \\ &= \; v^\intercal g_q v \;=\; \|v\|_{g_q}^2 \end{aligned} \tag{9}$$

With $\gamma_t = (\gamma_t^1, \ldots, \gamma_t^M) \in \mathbb{R}^{M \times D}$, we can write

$$C_k(\gamma) \;=\; \int_0^1 \dot{\gamma_t}^{\mathsf{T}} g_{\gamma_t} \dot{\gamma_t} \mathrm{d}t. \tag{8}$$

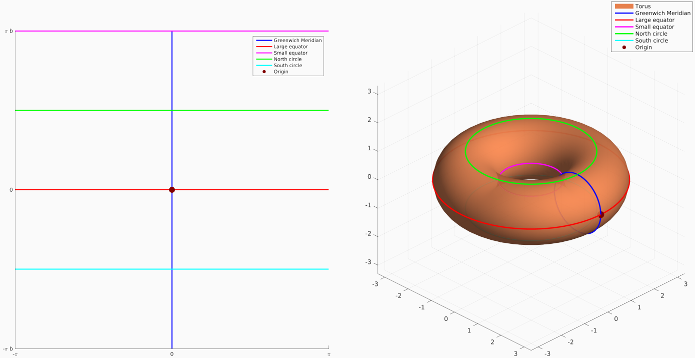
Optimal deformations are geodesics on the space of landmarks $\mathbb{R}^{M \times D}$ endowed with a Riemannian metric g_q :

$$\begin{aligned} \frac{\big(\mathrm{d}_g(q \to q + v \cdot \mathrm{d}t)\big)^2}{\mathrm{d}t} &= \sum_m \|v^m\|^2 + \sum_{m,m'} k(q^m, q^{m'}) \cdot \left\| v^m - v^{m'} \right\|^2 \\ &= \; v^{\mathsf{T}} g_q v \;=\; \|v\|_{g_q}^2 \end{aligned} \tag{9}$$

The LDDMM framework

Geodesic shooting on a Riemannian manifold

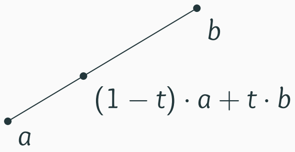
Riemann : conveniently working with arbitrary geometries



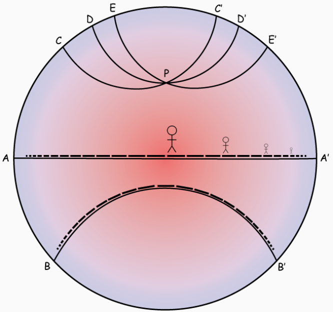
(a) As a deformed square. **(b)** Embedded in $\mathbb{R}^3$.

Figure 12: The donut-shaped torus.

Sometimes, we can compute geodesics explicitly...



(a) The Euclidean plane.



(b) The Poincaré disk.

Figure 13: Explicit geodesics on homogeneous manifolds.
(b) is adapted from `www.pitt.edu/~jdnorton/`.

But this is not the case in general

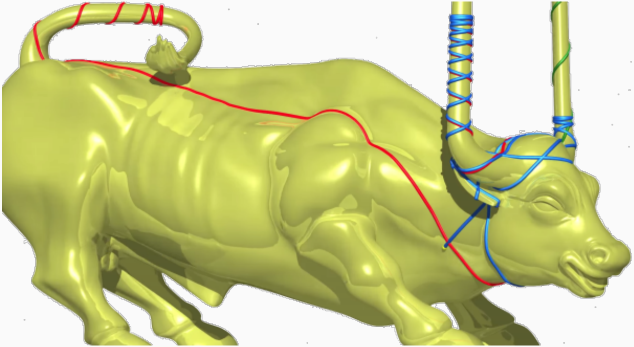


Figure 14: Geodesics on the Duhem's bull, embedded in $\mathbb{R}^3$.
Taken from `www.chaos-math.org`.

A first result : the geodesic equation

Geodesic $\Longrightarrow$ locally "straight" $\Longrightarrow$ second order ODE, the **geodesic equation** satisfied by $\gamma_t = (\gamma_t^1, \ldots, \gamma_t^D)$:

$$\forall\, d \in [\![1, D]\!]\,,\ \ \ddot{\gamma}_t^d = - \sum_{1 \leqslant i,j \leqslant D} \Gamma_{ij}^d(\gamma_t) \cdot \dot{\gamma}_t^i \, \dot{\gamma}_t^j, \tag{10}$$

where the Christoffel symbols $\Gamma_{ij}^d(\gamma_t)$ are given by :

$$\Gamma_{ij}^d(\gamma_t) = \frac{1}{2} \sum_{l=1}^{D} g^{dl}(q) \cdot \big(\partial_i g_{jl}(q) + \partial_j g_{il}(q) - \partial_l g_{ij}(q)\big)\,, \tag{11}$$

with g_{ij} the *metric* tensor and g^{dl} its inverse, the *cometric*.

A first result : the geodesic equation

Geodesic $\Longrightarrow$ locally "straight" $\Longrightarrow$ second order ODE,
the **geodesic equation** satisfied by $\gamma_t = (\gamma_t^1, \ldots, \gamma_t^D)$:

$$\forall\, d \in [\![1, D]\!]\,, \;\; \ddot{\gamma}_t^d = - \sum_{1 \leqslant i,j \leqslant D} \Gamma_{ij}^d(\gamma_t) \cdot \dot{\gamma}_t^i \, \dot{\gamma}_t^j, \tag{10}$$

where the Christoffel symbols $\Gamma_{ij}^d(\gamma_t)$ are given by :

$$\Gamma_{ij}^d(\gamma_t) = \frac{1}{2} \sum_{l=1}^{D} g^{dl}(q) \cdot \big(\partial_i g_{jl}(q) + \partial_j g_{il}(q) - \partial_l g_{ij}(q) \big)\,, \tag{11}$$

with g_{ij} the *metric* tensor and g^{dl} its inverse, the *cometric*.

From celerity to momentum

The “Christoffel” equation is an ODE on the tangent bundle :

$$(q_t, v_t) \;=\; (\gamma_t, \dot{\gamma}_t). \tag{12}$$

Hamilton : one should work on the cotangent bundle :

$$(q_t, p_t) \;=\; (q_t, g_{q_t} v_t). \tag{13}$$

We denote $K_q = g_q^{-1}$ and $H(q, p) = \frac{1}{2} p^\intercal K_q p$, so that

$$\frac{1}{2} v_t^\intercal g_{q_t} v_t \;=\; \underbrace{\frac{1}{2} \|\dot{\gamma}_t\|^2_{\gamma_t}}_{\text{Kinetic energy}} \;=\; \frac{1}{2} p_t^\intercal K_{q_t} p_t \;=\; H(q_t, p_t). \tag{14}$$

From celerity to momentum

The "Christoffel" equation is an ODE on the tangent bundle :

$$(q_t, v_t) \;=\; (\gamma_t, \dot{\gamma}_t). \qquad (12)$$

Hamilton : one should work on the cotangent bundle :

$$(q_t, p_t) \;=\; (q_t, g_{q_t} v_t). \qquad (13)$$

We denote $K_q = g_q^{-1}$ and $H(q,p) = \frac{1}{2} p^\top K_q p$, so that

$$\frac{1}{2} v_t^\top g_{q_t} v_t \;=\; \underbrace{\frac{1}{2} \|\dot{\gamma}_t\|_{\gamma_t}^2}_{\text{Kinetic energy}} \;=\; \frac{1}{2} p_t^\top K_{q_t} p_t \;=\; H(q_t, p_t). \qquad (14)$$

The "Christoffel" equation is an ODE on the tangent bundle :

$$(q_t, v_t) \;=\; (\gamma_t, \dot\gamma_t). \tag{12}$$

Hamilton : one should work on the cotangent bundle :

$$(q_t, p_t) \;=\; (q_t, g_{q_t} v_t). \tag{13}$$

We denote $K_q = g_q^{-1}$ and $H(q,p) = \frac{1}{2} p^\intercal K_q p$, so that

$$\frac{1}{2} v_t^\intercal g_{q_t} v_t \;=\; \underbrace{\frac{1}{2}\|\dot\gamma_t\|^2_{\gamma_t}}_{\text{Kinetic energy}} \;=\; \frac{1}{2} p_t^\intercal K_{q_t} p_t \;=\; H(q_t, p_t). \tag{14}$$

Hamilton geodesic equations

Hamilton, 1833

γ_t is a geodesic if and only if the lifted cotangent trajectory (q_t, p_t) follows the Hamiltonian equation :

$$\left\{ \begin{array}{lclcl} \dot{q}_t & = & +\frac{\partial H}{\partial p}(q_t, p_t) & = & +K_{q_t} p_t \\ \dot{p}_t & = & -\frac{\partial H}{\partial q}(q_t, p_t) & = & -\frac{1}{2}\partial_q (p_t, K_q p_t)(q_t) \end{array} \right. . \tag{15}$$

In the cotangent phase space, we flow along the symplectic gradient :

$$X(q,p) = \begin{pmatrix} +\frac{\partial H}{\partial p}(q,p) \\ -\frac{\partial H}{\partial q}(q,p) \end{pmatrix} = \text{“}\mathrm{R}_{-90^\circ}\text{”}\big(\nabla H(q,p)\big). \tag{16}$$

Hamiltonian geodesic equations

Hamilton, 1833

γ_t is a geodesic if and only if the lifted cotangent trajectory (q_t, p_t) follows the Hamiltonian equation :

$$\left\{ \begin{array}{rcrcl} \dot{q}_t & = & +\frac{\partial H}{\partial p}(q_t, p_t) & = & +K_{q_t} p_t \\ \dot{p}_t & = & -\frac{\partial H}{\partial q}(q_t, p_t) & = & -\frac{1}{2}\partial_q (p_t, K_q p_t)(q_t) \end{array} \right. . \tag{15}$$

In the cotangent phase space, we flow along the symplectic gradient :

$$X(q,p) = \begin{pmatrix} +\frac{\partial H}{\partial p}(q,p) \\ -\frac{\partial H}{\partial q}(q,p) \end{pmatrix} = \text{“}\mathrm{R}_{-90^\circ}\text{”}\big(\nabla H(q,p)\big). \tag{16}$$

Short physical “example”

Consider a free-falling particle of mass m :

$$q = z, \qquad v = \dot{z}, \tag{17}$$

$$\dot{q} = v, \qquad \dot{v} = -g. \tag{18}$$

Now, we can write $\mathbf{p} = \mathbf{m}\mathbf{v}$ so that

$$H(q,p) = \text{“}E_{\text{cin}}\text{”}(q,p) + \text{“}E_{\text{pp}}\text{”}(q,p) = \frac{1}{2}\frac{p^2}{m} + mgq. \tag{19}$$

We find :

$$\begin{cases} \dot{q} & = & +\frac{\partial H}{\partial p} & = & +p/m \\ \dot{p} & = & -\frac{\partial H}{\partial q} & = & -mg \end{cases}. \tag{20}$$

Short physical "example"

Consider a free-falling particle of mass m :

$$q = z, \qquad v = \dot{z}, \tag{17}$$

$$\dot{q} = v, \qquad \dot{v} = -g. \tag{18}$$

Now, we can write $\mathbf{p} = \mathbf{m}\mathbf{v}$ so that

$$H(q,p) = \text{"}E_{\text{cin}}\text{"}(q,p) + \text{"}E_{\text{pp}}\text{"}(q,p) = \frac{1}{2}\frac{p^2}{m} + mgq. \tag{19}$$

We find :

$$\begin{cases} \dot{q} = +\frac{\partial H}{\partial p} = +p/m \\ \dot{p} = -\frac{\partial H}{\partial q} = -mg \end{cases}. \tag{20}$$

Short physical "example"

Consider a free-falling particle of mass m :

$$q = z, \qquad v = \dot{z}, \tag{17}$$

$$\dot{q} = v, \qquad \dot{v} = -g. \tag{18}$$

Now, we can write $\mathbf{p} = \mathbf{m}\mathbf{v}$ so that

$$H(q,p) \;=\; \text{“}E_{\mathrm{cin}}\text{”}(q,p) + \text{“}E_{\mathrm{pp}}\text{”}(q,p) \;=\; \frac{1}{2}\frac{p^2}{m} + mgq. \tag{19}$$

We find :

$$\begin{cases} \dot{q} & = & +\frac{\partial H}{\partial p} & = & +p/m \\ \dot{p} & = & -\frac{\partial H}{\partial q} & = & -mg \end{cases}. \tag{20}$$

Short physical "example"

Consider a free-falling particle of mass m :

$$q = z, \qquad v = \dot{z}, \tag{17}$$

$$\dot{q} = v, \qquad \dot{v} = -g. \tag{18}$$

Now, we can write $\mathbf{p} = \mathbf{m}\mathbf{v}$ so that

$$H(q,p) \; = \; \text{“}E_{\text{cin}}\text{”}(q,p) + \text{“}E_{\text{pp}}\text{”}(q,p) \; = \; \frac{1}{2}\frac{p^2}{m} + mgq. \tag{19}$$

We find :

$$\begin{cases} \dot{q} & = & +\frac{\partial H}{\partial p} & = & +p/m \\ \dot{p} & = & -\frac{\partial H}{\partial q} & = & -mg \end{cases}. \tag{20}$$

The geodesic shooting algorithm

A geodesic path γ_t is characterized by (q_0, p_0).
To compute any geodesic starting from a source q_0, we simply need a shooting momentum p_0 and a simplistic Euler scheme :

$$\left\{ \begin{array}{lllll} q_{t+0.1} & = & q_t & + & 0.1 \cdot K_{q_t} p_t \\ p_{t+0.1} & = & p_t & - & 0.1 \cdot \partial_q (p_t, K_q p_t)(q_t) \end{array} \right. . \tag{21}$$

Exponential map :

$$\mathrm{Exp}_{q_0} : p_0 \in T^\star_{q_0} \mathcal{M} \mapsto q_1 \in \mathcal{M} \tag{22}$$

A geodesic path γ_t is characterized by (q_0, p_0).
To compute any geodesic starting from a source q_0, we simply need a shooting momentum p_0 and a simplistic Euler scheme :

$$\begin{cases} q_{t+0.1} & = & q_t & + & 0.1 \cdot K_{q_t} p_t \\ p_{t+0.1} & = & p_t & - & 0.1 \cdot \partial_q (p_t, K_q p_t)(q_t) \end{cases} . \tag{21}$$

Exponential map :

$$\mathrm{Exp}_{q_0} : p_0 \in T^\star_{q_0}\mathcal{M} \mapsto q_1 \in \mathcal{M} \tag{22}$$

Lessons taught by the Hamiltonian theory of geodesics

We are looking for :

- Tearing-adverse metrics on the space of landmarks
- Efficient ways to compute geodesics (deformations)

Hamilton has taught us that :

- Geodesics are "simple" iff the cometric $K_q = g_q^{-1}$ is simple
- The **Exponential map** can be computed efficiently

Lessons taught by the Hamiltonian theory of geodesics

We are looking for :

- Tearing-adverse metrics on the space of landmarks
- Efficient ways to compute geodesics (deformations)

Hamilton has taught us that :

- Geodesics are "simple" iff the cometric $K_q = g_q^{-1}$ is simple
- The **Exponential map** can be computed efficiently

The LDDMM framework

Kernel cometrics and Diffeomorphic trajectories

Parallelism is the way forward

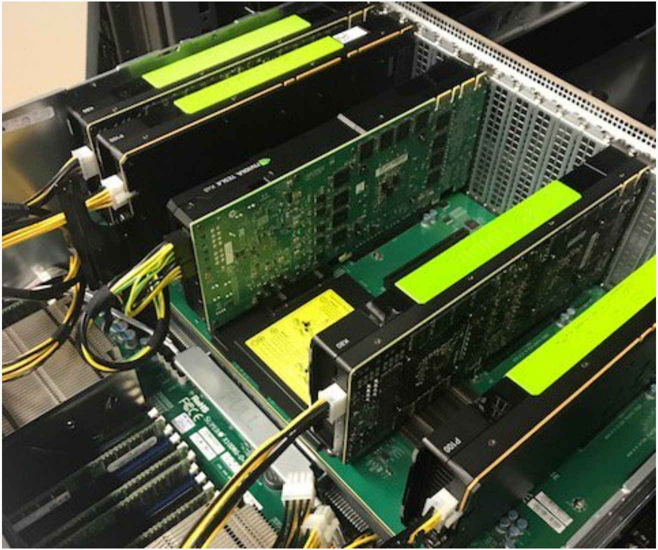


Figure 15: Highly-parallel MoKaMachine (Mokaplan Inria team).

GPUs in action

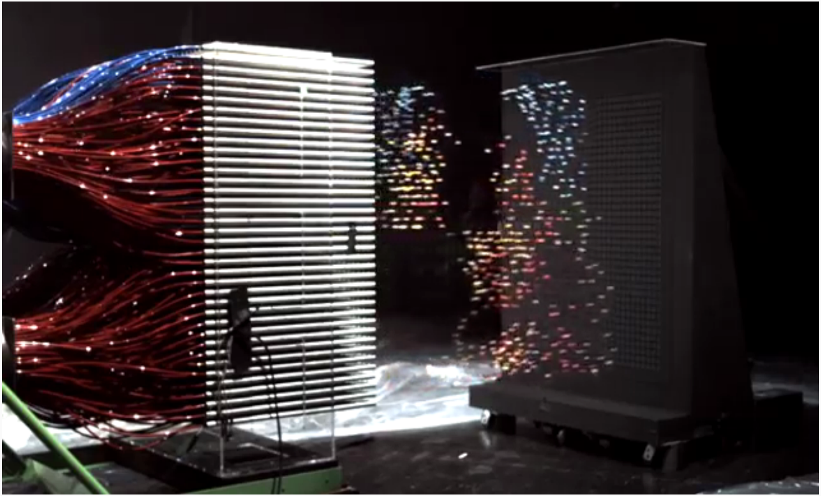


Figure 16: *Mythbusters Demo GPU versus CPU*, from the Nvidia YouTube channel.

Kernel cometrics, reduced tensor

Use a reduced kernel matrix

$$k_q = \begin{pmatrix} k(q^1,q^1) & k(q^1,q^2) & \cdots & k(q^1,q^M) \\ k(q^2,q^1) & k(q^2,q^2) & \cdots & k(q^2,q^M) \\ \vdots & \vdots & \ddots & \vdots \\ k(q^M,q^1) & k(q^M,q^2) & \cdots & k(q^M,q^M) \end{pmatrix} \qquad (23)$$

so that

$$H(q,p) \;=\; \frac{1}{2} p^\intercal K_q p \;=\; \frac{1}{2} \sum_{i,j=1}^{M} k(q^i,q^j) \cdot p^{i\intercal} p^j. \qquad (24)$$

In a computational sense, this is the simplest family of cometrics on the space of points clouds.

Kernel cometrics, reduced tensor

Use a reduced kernel matrix

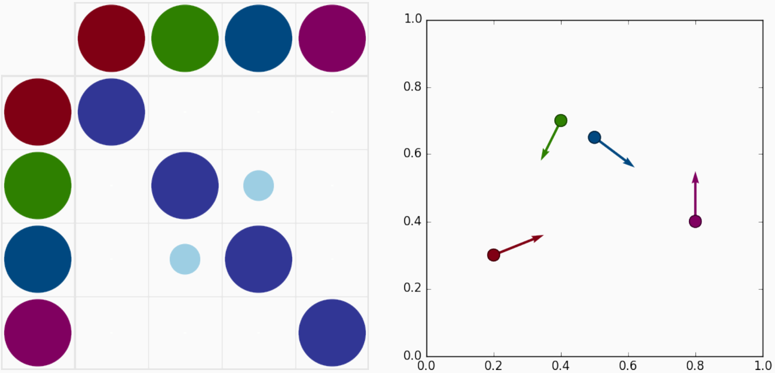
$$k_q = \begin{pmatrix} k(q^1, q^1) & k(q^1, q^2) & \cdots & k(q^1, q^M) \\ k(q^2, q^1) & k(q^2, q^2) & \cdots & k(q^2, q^M) \\ \vdots & \vdots & \ddots & \vdots \\ k(q^M, q^1) & k(q^M, q^2) & \cdots & k(q^M, q^M) \end{pmatrix} \tag{23}$$

so that

$$H(q, p) \;=\; \frac{1}{2} p^{\intercal} K_q p \;=\; \frac{1}{2} \sum_{i,j=1}^{M} k(q^i, q^j) \cdot p^{i\intercal} p^j. \tag{24}$$

In a computational sense, this is the simplest family of cometrics on the space of points clouds.

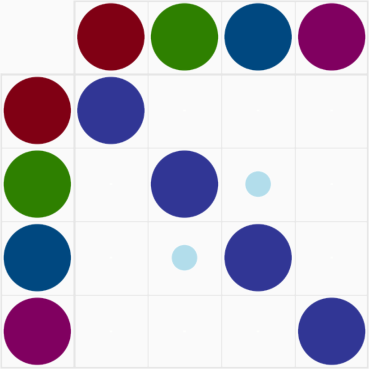
Influence of the kernel width, $\sigma = .25$



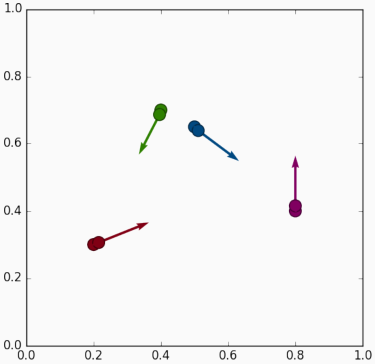
(a) Kernel matrix k_{q_t}. **(b)** Shooted cloud (q_t, p_t).

Figure 17: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2 / 2\sigma^2)$, $\sigma = .25$.

Influence of the kernel width, $\sigma = .25$



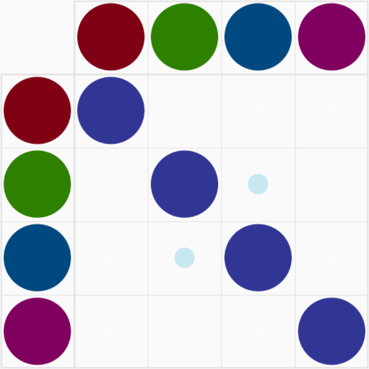
(a) Kernel matrix k_{q_t}.



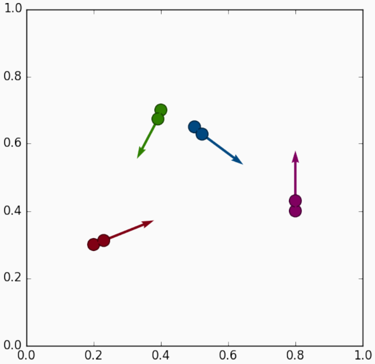
(b) Shooted cloud (q_t, p_t).

Figure 17: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2 / 2\sigma^2)$, $\sigma = .25$.

Influence of the kernel width, $\sigma = .25$



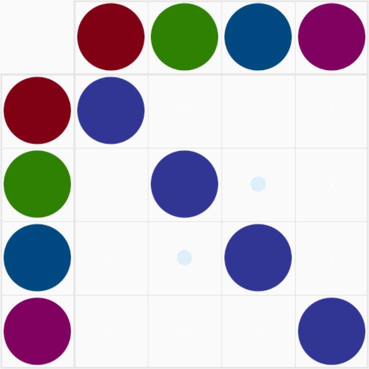
(a) Kernel matrix k_{q_t}.



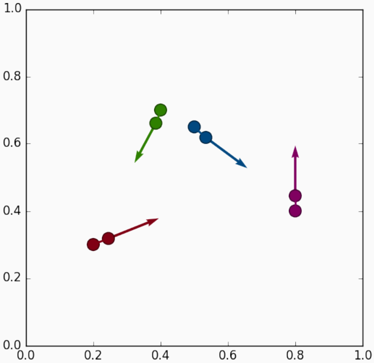
(b) Shooted cloud (q_t, p_t).

Figure 17: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2 / 2\sigma^2)$, $\sigma = .25$.

Influence of the kernel width, $\sigma = .25$



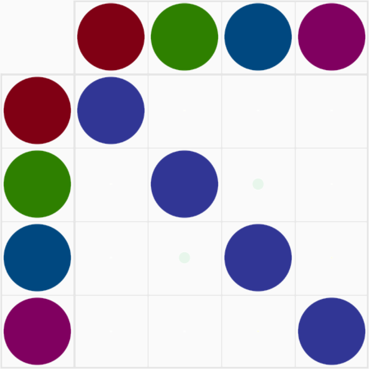
(a) Kernel matrix k_{q_t}.



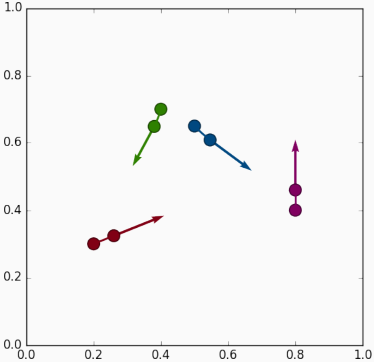
(b) Shooted cloud (q_t, p_t).

Figure 17: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2/2\sigma^2)$, $\sigma = .25$.

Influence of the kernel width, $\sigma = .25$



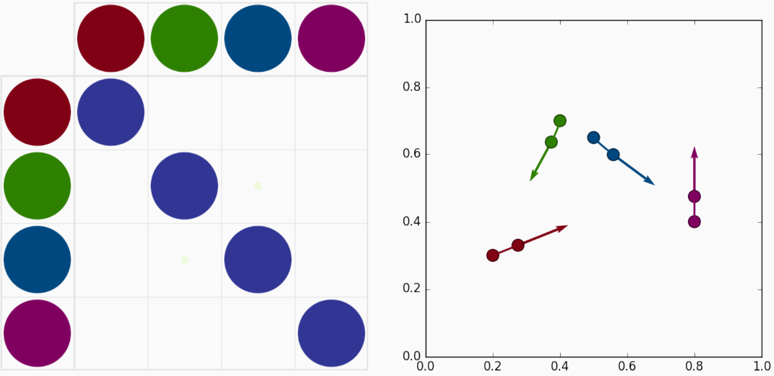
(a) Kernel matrix k_{q_t}.



(b) Shooted cloud (q_t, p_t).

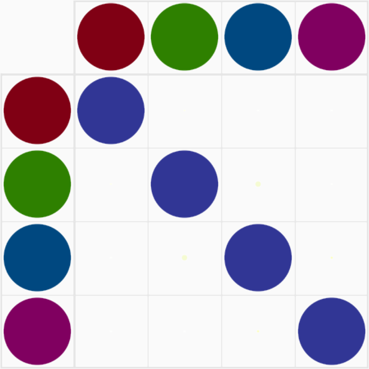
Figure 17: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2/2\sigma^2)$, $\sigma = .25$.

Influence of the kernel width, $\sigma = .25$

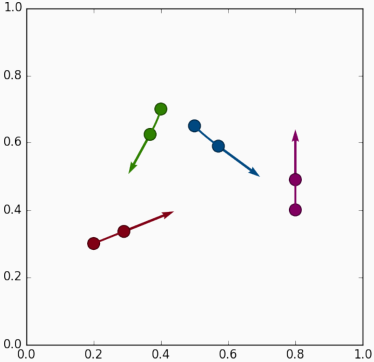


(a) Kernel matrix k_{q_t}. **(b)** Shooted cloud (q_t, p_t).

Figure 17: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2/2\sigma^2)$, $\sigma = .25$.

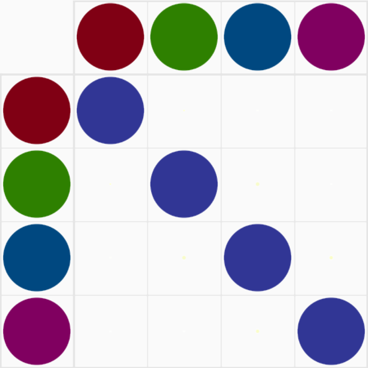


(a) Kernel matrix k_{q_t}.

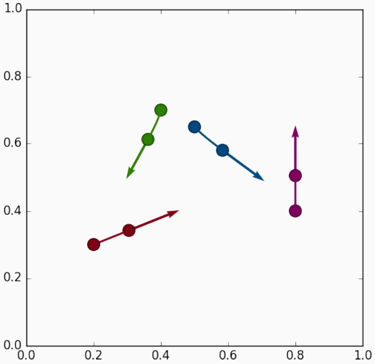


(b) Shooted cloud (q_t, p_t).

Figure 17: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2/2\sigma^2)$, $\sigma = .25$.



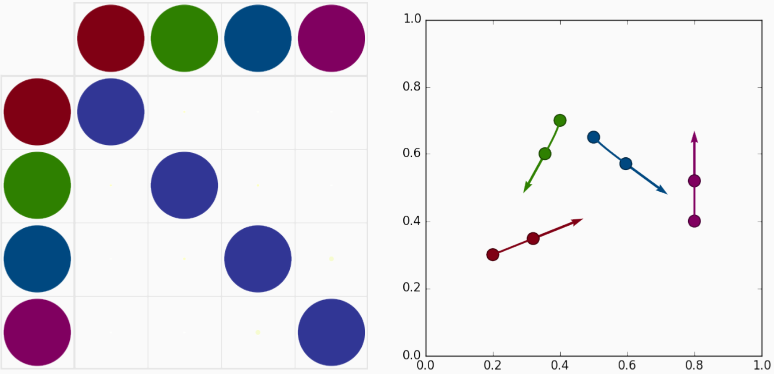
(a) Kernel matrix k_{q_t}.



(b) Shooted cloud (q_t, p_t).

Figure 17: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2 / 2\sigma^2)$, $\sigma = .25$.

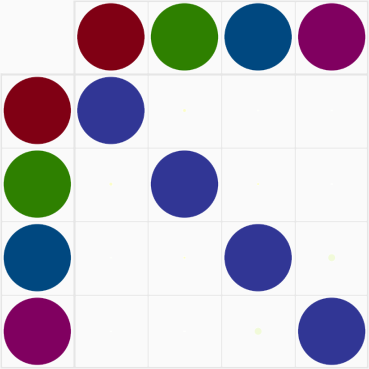
Influence of the kernel width, $\sigma = .25$



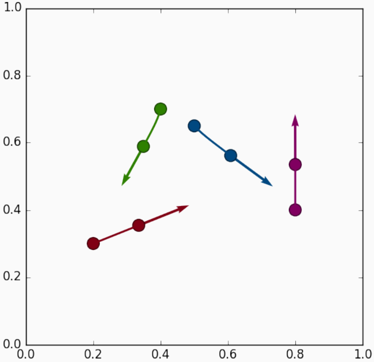
(a) Kernel matrix k_{q_t}. **(b)** Shooted cloud (q_t, p_t).

Figure 17: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2 / 2\sigma^2)$, $\sigma = .25$.

Influence of the kernel width, $\sigma = .25$

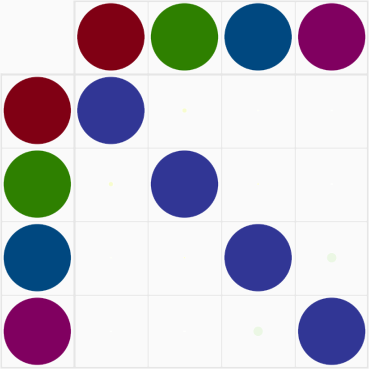


(a) Kernel matrix k_{q_t}.

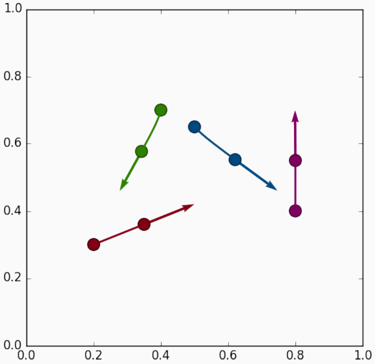


(b) Shooted cloud (q_t, p_t).

Figure 17: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2/2\sigma^2)$, $\sigma = .25$.

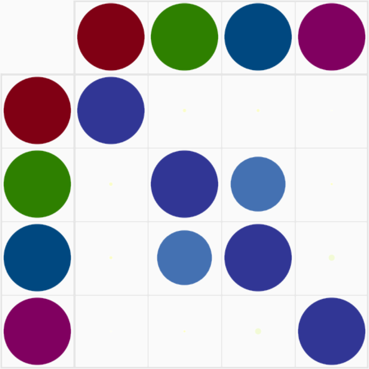


(a) Kernel matrix k_{q_t}.

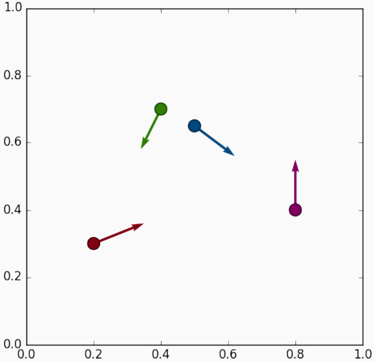


(b) Shooted cloud (q_t, p_t).

Figure 17: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2 / 2\sigma^2)$, $\sigma = .25$.



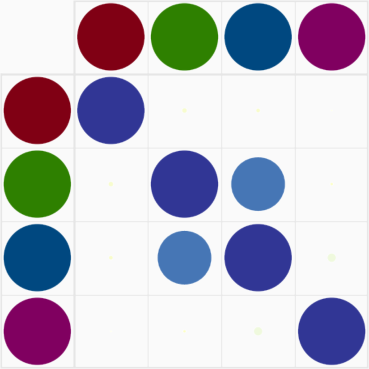
(a) Kernel matrix k_{q_t}.



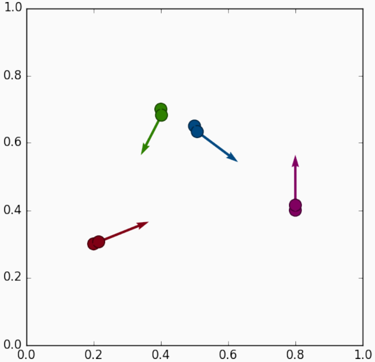
(b) Shooted cloud (q_t, p_t).

Figure 18: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2 / 2\sigma^2)$, $\sigma = .35$.

Influence of the kernel width, $\sigma = .35$



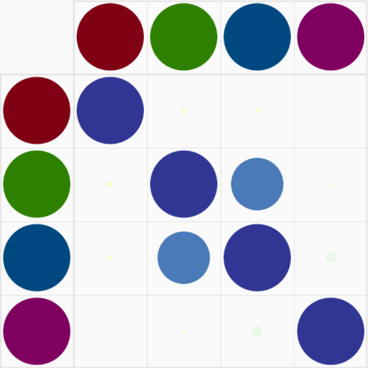
(a) Kernel matrix k_{q_t}.



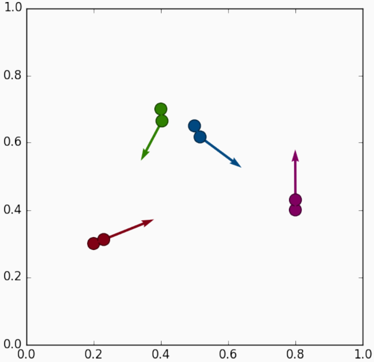
(b) Shooted cloud (q_t, p_t).

Figure 18: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2 / 2\sigma^2)$, $\sigma = .35$.

Influence of the kernel width, $\sigma = .35$

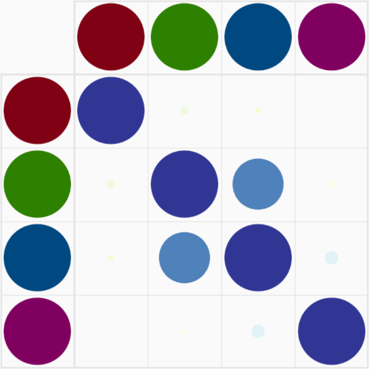


(a) Kernel matrix k_{q_t}.

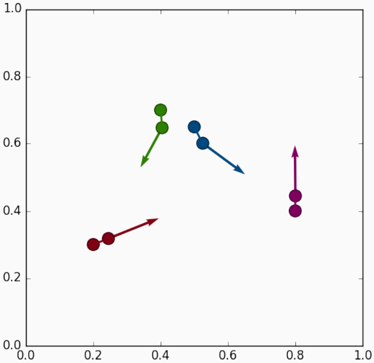


(b) Shooted cloud (q_t, p_t).

Figure 18: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2/2\sigma^2)$, $\sigma = .35$.



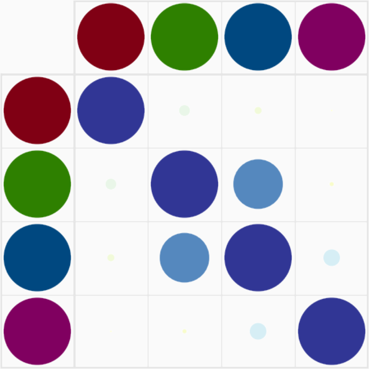
(a) Kernel matrix k_{q_t}.



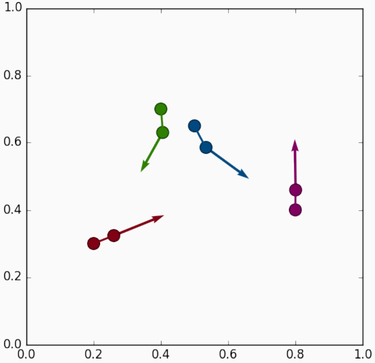
(b) Shooted cloud (q_t, p_t).

Figure 18: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2/2\sigma^2)$, $\sigma = .35$.

Influence of the kernel width, $\sigma = .35$



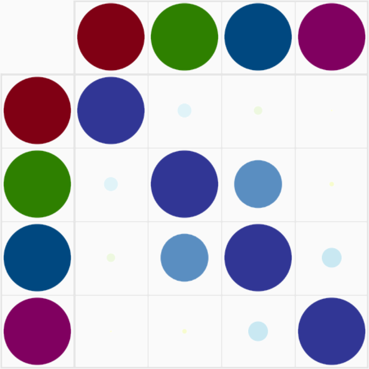
(a) Kernel matrix k_{q_t}.



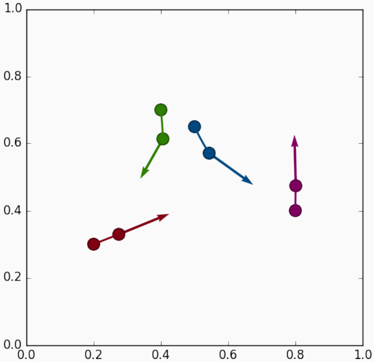
(b) Shooted cloud (q_t, p_t).

Figure 18: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2 / 2\sigma^2)$, $\sigma = .35$.

Influence of the kernel width, $\sigma = .35$



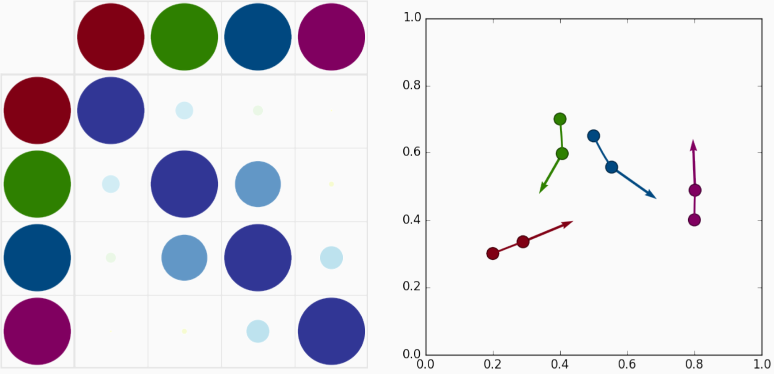
(a) Kernel matrix k_{q_t}.



(b) Shooted cloud (q_t, p_t).

Figure 18: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2 / 2\sigma^2)$, $\sigma = .35$.

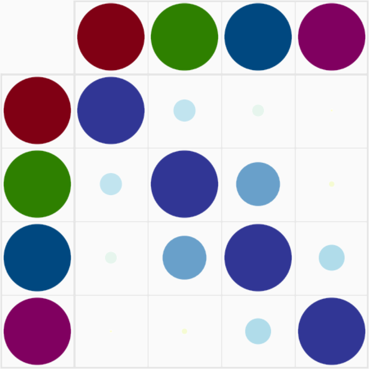
Influence of the kernel width, $\sigma = .35$



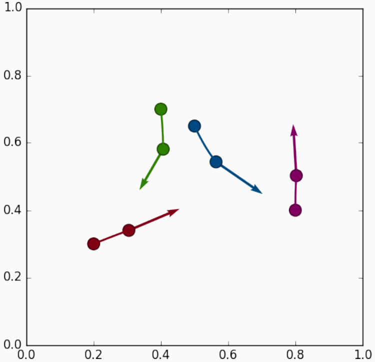
(a) Kernel matrix k_{q_t}. **(b)** Shooted cloud (q_t, p_t).

Figure 18: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2/2\sigma^2)$, $\sigma = .35$.

Influence of the kernel width, $\sigma = .35$



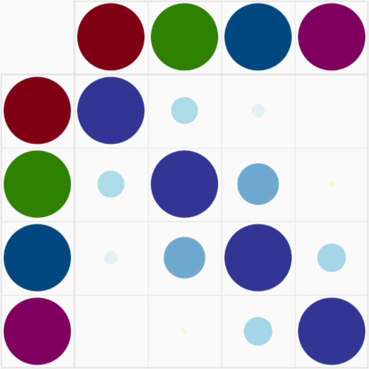
(a) Kernel matrix k_{q_t}.



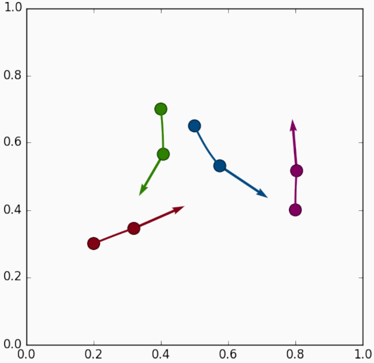
(b) Shooted cloud (q_t, p_t).

Figure 18: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2/2\sigma^2)$, $\sigma = .35$.

Influence of the kernel width, $\sigma = .35$



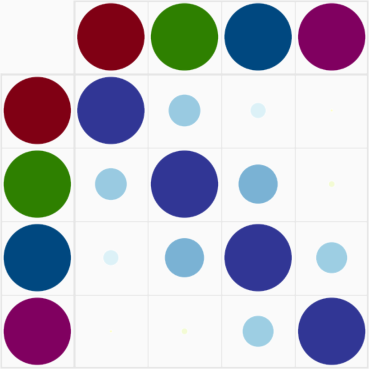
(a) Kernel matrix k_{q_t}.



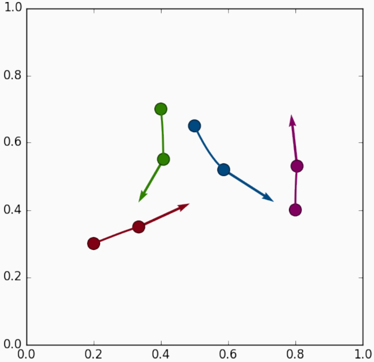
(b) Shooted cloud (q_t, p_t).

Figure 18: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2/2\sigma^2)$, $\sigma = .35$.

Influence of the kernel width, $\sigma = .35$



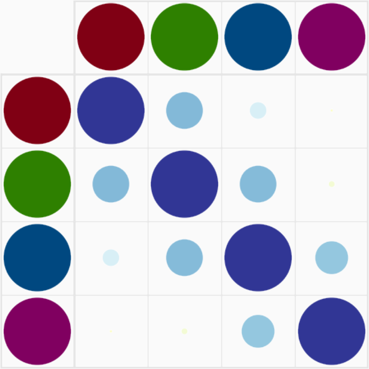
(a) Kernel matrix k_{q_t}.



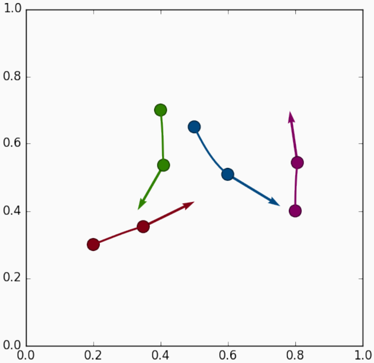
(b) Shooted cloud (q_t, p_t).

Figure 18: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2 / 2\sigma^2)$, $\sigma = .35$.

Influence of the kernel width, $\sigma = .35$



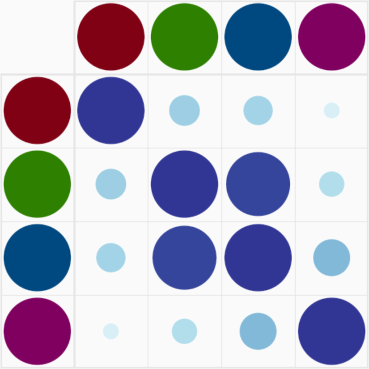
(a) Kernel matrix k_{q_t}.



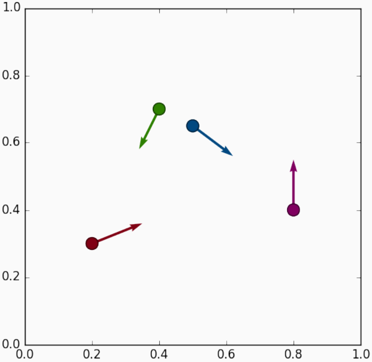
(b) Shooted cloud (q_t, p_t).

Figure 18: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2 / 2\sigma^2)$, $\sigma = .35$.

Influence of the kernel width, $\sigma = .50$



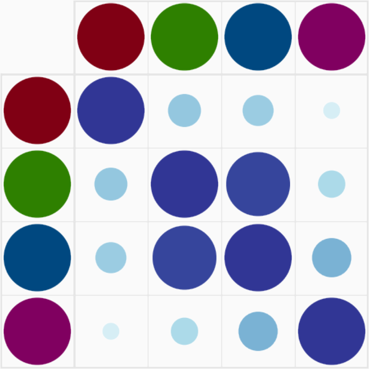
(a) Kernel matrix k_{q_t}.



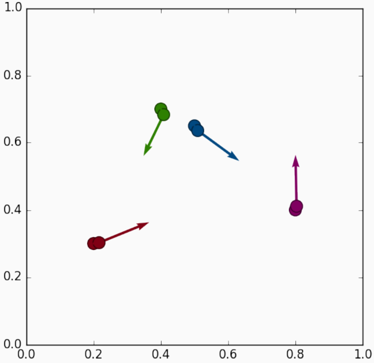
(b) Shooted cloud (q_t, p_t).

Figure 19: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2/2\sigma^2)$, $\sigma = .50$.

Influence of the kernel width, $\sigma = .50$



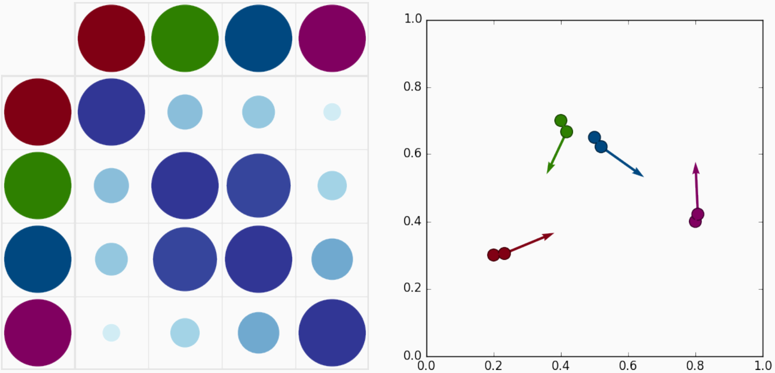
(a) Kernel matrix k_{q_t}.



(b) Shooted cloud (q_t, p_t).

Figure 19: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2/2\sigma^2)$, $\sigma = .50$.

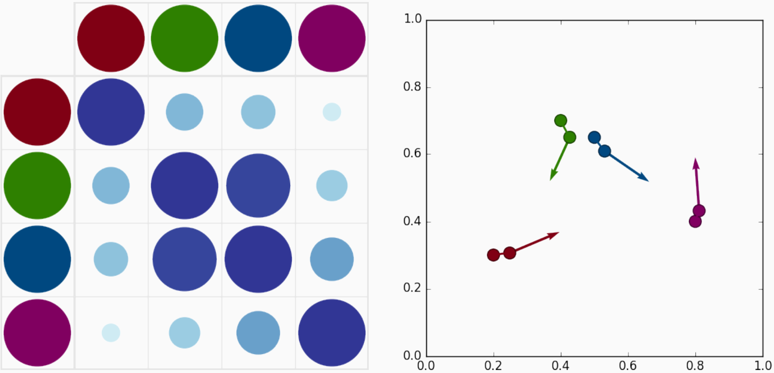
Influence of the kernel width, $\sigma = .50$



(a) Kernel matrix k_{q_t}. **(b)** Shooted cloud (q_t, p_t).

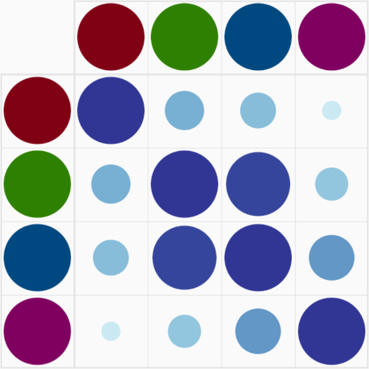
Figure 19: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2/2\sigma^2)$, $\sigma = .50$.

Influence of the kernel width, $\sigma = .50$

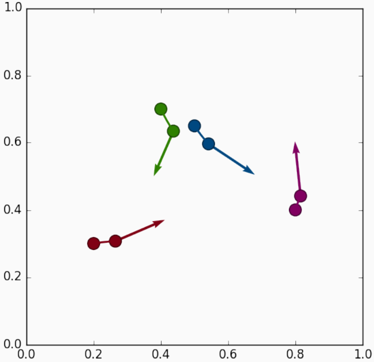


(a) Kernel matrix k_{q_t}.   **(b)** Shooted cloud (q_t, p_t).

Figure 19: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2 / 2\sigma^2)$, $\sigma = .50$.



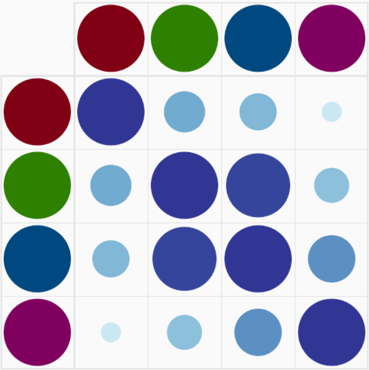
(a) Kernel matrix k_{q_t}.



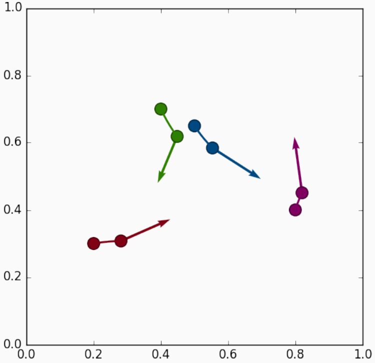
(b) Shooted cloud (q_t, p_t).

Figure 19: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2 / 2\sigma^2)$, $\sigma = .50$.

Influence of the kernel width, $\sigma = .50$



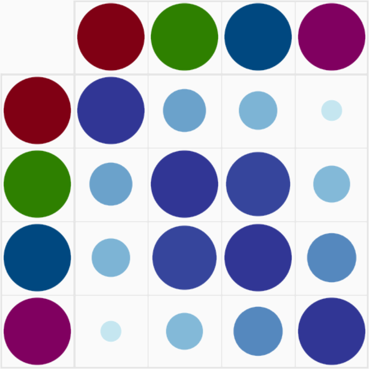
(a) Kernel matrix k_{q_t}.



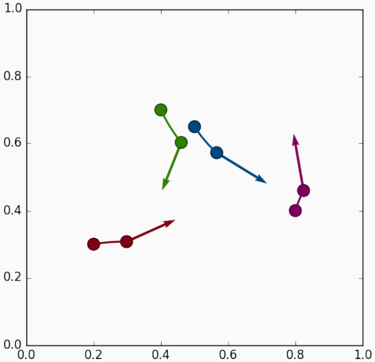
(b) Shooted cloud (q_t, p_t).

Figure 19: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2 / 2\sigma^2)$, $\sigma = .50$.

Influence of the kernel width, $\sigma = .50$



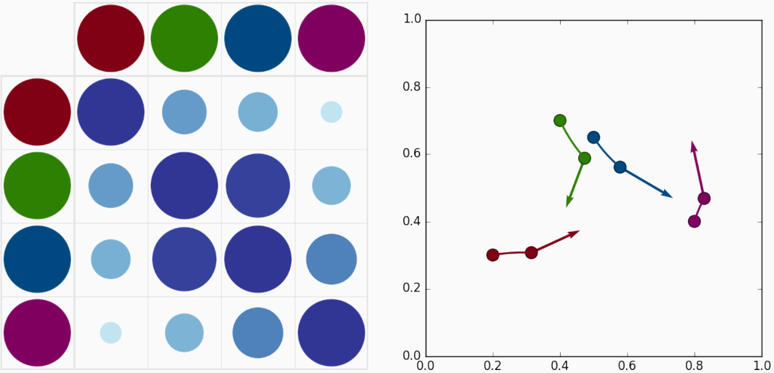
(a) Kernel matrix k_{q_t}.



(b) Shooted cloud (q_t, p_t).

Figure 19: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2 / 2\sigma^2)$, $\sigma = .50$.

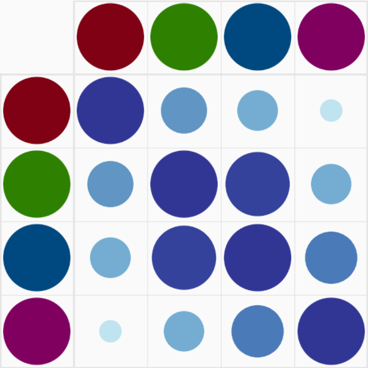
Influence of the kernel width, $\sigma = .50$



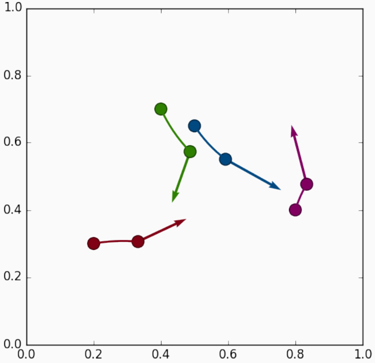
(a) Kernel matrix k_{q_t}. **(b)** Shooted cloud (q_t, p_t).

Figure 19: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2/2\sigma^2)$, $\sigma = .50$.

Influence of the kernel width, $\sigma = .50$



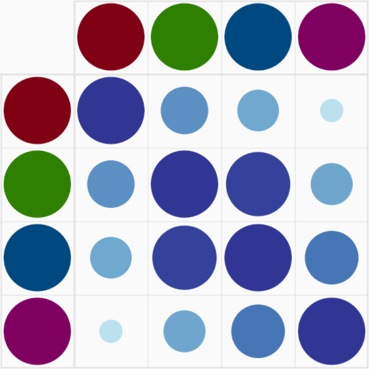
(a) Kernel matrix k_{q_t}.



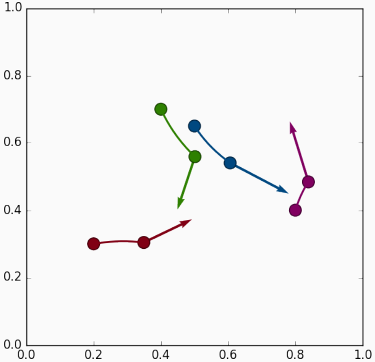
(b) Shooted cloud (q_t, p_t).

Figure 19: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2 / 2\sigma^2)$, $\sigma = .50$.

Influence of the kernel width, $\sigma = .50$



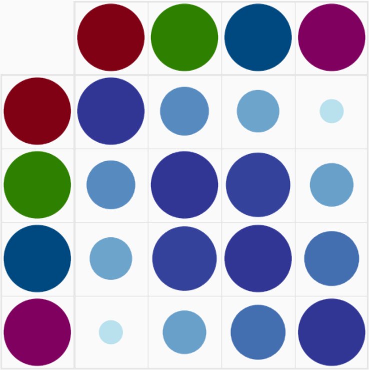
(a) Kernel matrix k_{q_t}.



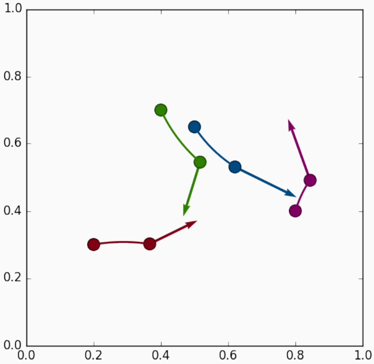
(b) Shooted cloud (q_t, p_t).

Figure 19: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2 / 2\sigma^2)$, $\sigma = .50$.

Influence of the kernel width, $\sigma = .50$



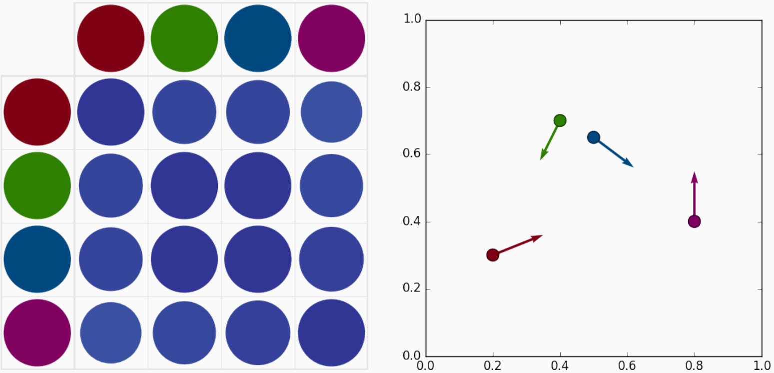
(a) Kernel matrix k_{q_t}.



(b) Shooted cloud (q_t, p_t).

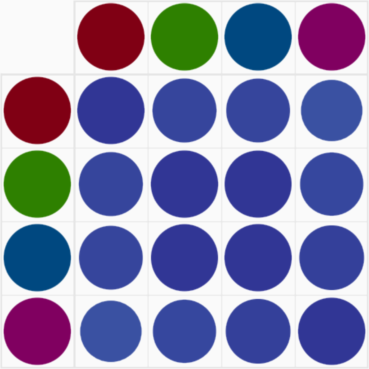
Figure 19: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2/2\sigma^2)$, $\sigma = .50$.

Influence of the kernel width, $\sigma = 1.$

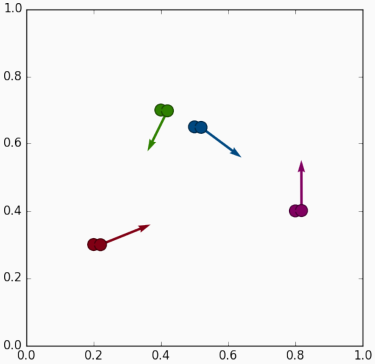


(a) Kernel matrix k_{q_t}. **(b)** Shooted cloud (q_t, p_t).

Figure 20: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2 / 2\sigma^2)$, $\sigma = 1.$.



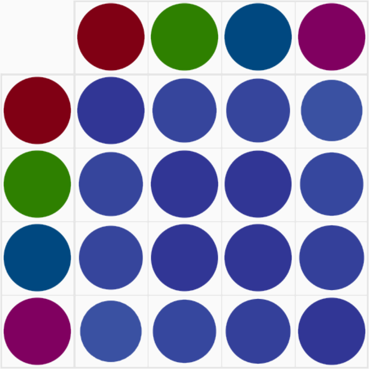
(a) Kernel matrix k_{q_t}.



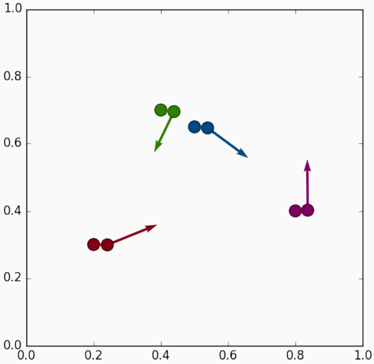
(b) Shooted cloud (q_t, p_t).

Figure 20: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2/2\sigma^2)$, $\sigma = 1..$

Influence of the kernel width, $\sigma = 1$.



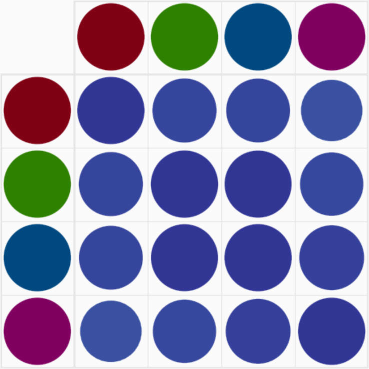
(a) Kernel matrix k_{q_t}.



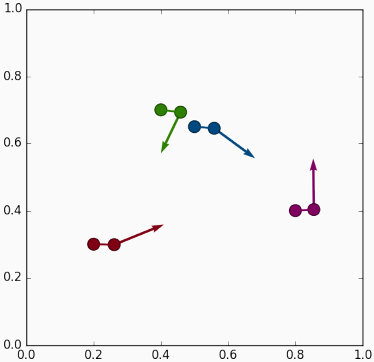
(b) Shooted cloud (q_t, p_t).

Figure 20: Geodesic shooting, $k(x - y) = \exp(-\|x - y\|^2 / 2\sigma^2)$, $\sigma = 1.$.

Influence of the kernel width, $\sigma = 1.$



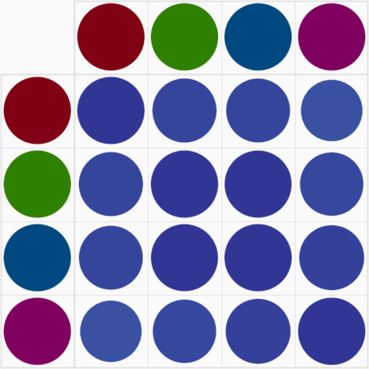
(a) Kernel matrix k_{q_t}.



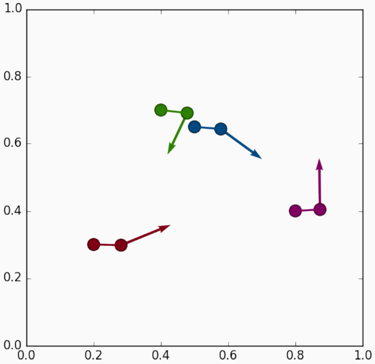
(b) Shooted cloud (q_t, p_t).

Figure 20: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2/2\sigma^2)$, $\sigma = 1.$.

Influence of the kernel width, $\sigma = 1.$



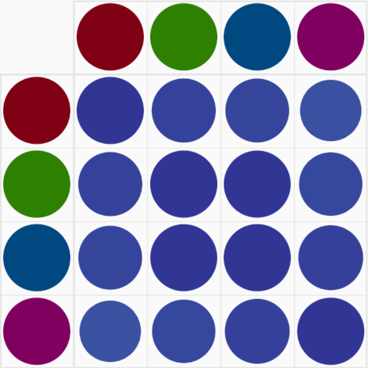
(a) Kernel matrix k_{q_t}.



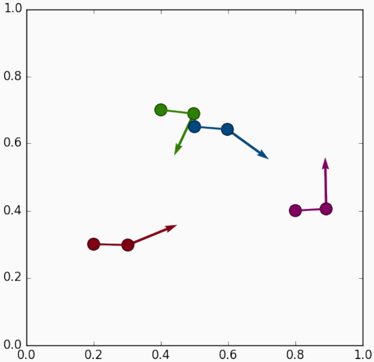
(b) Shooted cloud (q_t, p_t).

Figure 20: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2 / 2\sigma^2)$,
$\sigma = 1.$.

Influence of the kernel width, $\sigma = 1.$



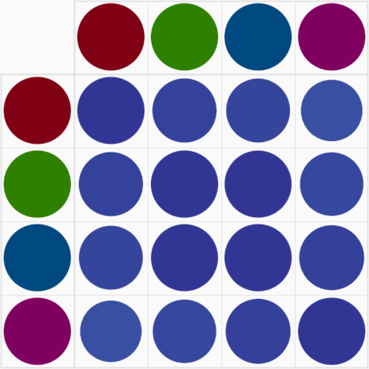
(a) Kernel matrix k_{q_t}.



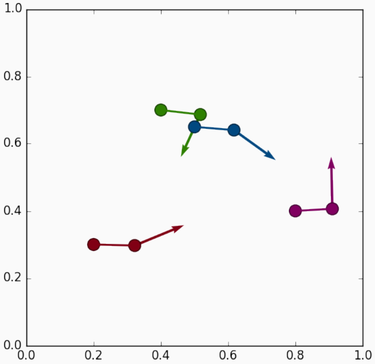
(b) Shooted cloud (q_t, p_t).

Figure 20: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2/2\sigma^2)$, $\sigma = 1..$

Influence of the kernel width, $\sigma = 1.$

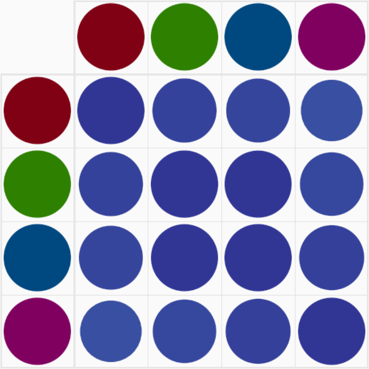


(a) Kernel matrix k_{q_t}.

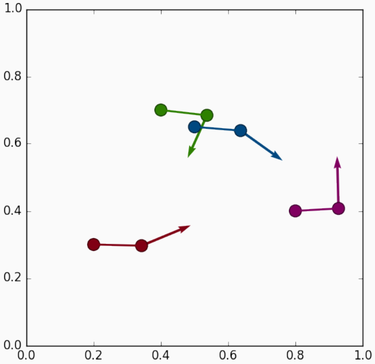


(b) Shooted cloud (q_t, p_t).

Figure 20: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2/2\sigma^2)$, $\sigma = 1..$



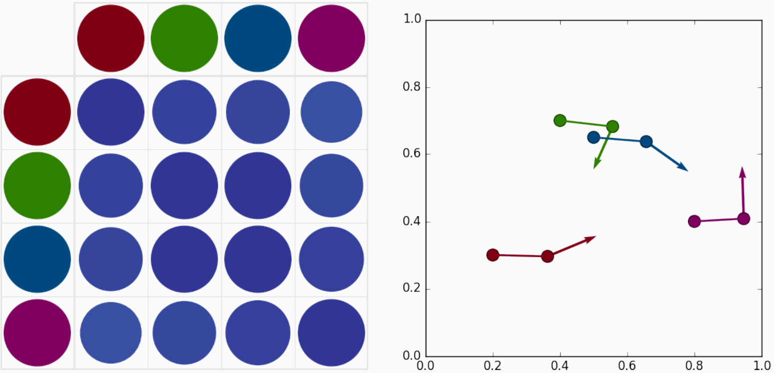
(a) Kernel matrix k_{q_t}.



(b) Shooted cloud (q_t, p_t).

Figure 20: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2/2\sigma^2)$, $\sigma = 1.$.

Influence of the kernel width, $\sigma = 1$.

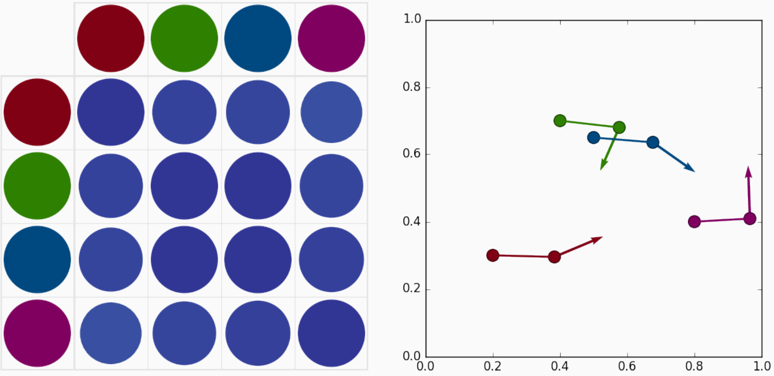


(a) Kernel matrix k_{q_t}.

(b) Shooted cloud (q_t, p_t).

Figure 20: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2/2\sigma^2)$, $\sigma = 1.$.

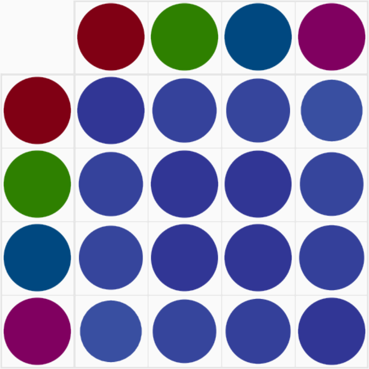
Influence of the kernel width, $\sigma = 1.$



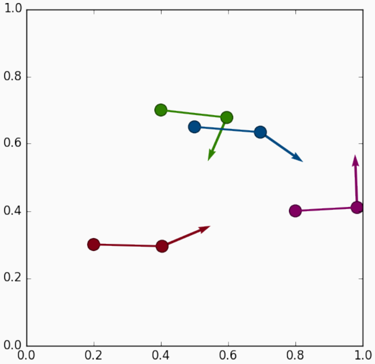
(a) Kernel matrix k_{q_t}. **(b)** Shooted cloud (q_t, p_t).

Figure 20: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2/2\sigma^2)$,
$\sigma = 1..$

Influence of the kernel width, $\sigma = 1.$



(a) Kernel matrix k_{q_t}.



(b) Shooted cloud (q_t, p_t).

Figure 20: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2/2\sigma^2)$, $\sigma = 1.$.

RKHS norms on velocity vector fields

Let k be a smooth enough kernel function, with $\widehat{k}(\omega) \in \mathbb{R}_+^\star$.
If $v : \mathbb{R}^D \to \mathbb{R}^D$ is a vector field on the ambient space, define

$$\|v\|_k^2 \;=\; \int_{\omega \in \mathbb{R}^D} \frac{1}{\widehat{k}(\omega)}\, |\widehat{v}(\omega)|^2 \,\mathrm{d}\omega. \tag{25}$$

- $V_k = \{ v \mid \|v\|_k < \infty \}$ is a Hilbert space of k-smooth vector fields
- We assume k is smooth enough, so that $\delta_x : v \mapsto v(x)$ belongs to the dual space $(V_k)^\star$: we link with the theory of **Reproducing Kernel Hilbert Spaces.**

RKHS norms on velocity vector fields

Let k be a smooth enough kernel function, with $\widehat{k}(\omega) \in \mathbb{R}^\star_+$.
If $v : \mathbb{R}^D \to \mathbb{R}^D$ is a vector field on the ambient space, define

$$\|v\|_k^2 = \int_{\omega \in \mathbb{R}^D} \frac{1}{\widehat{k}(\omega)} \, |\widehat{v}(\omega)|^2 \, \mathrm{d}\omega. \tag{25}$$

- $V_k = \{ v \mid \|v\|_k < \infty \}$ is a Hilbert space of k-smooth vector fields
- We assume k is smooth enough, so that $\delta_x : v \mapsto v(x)$ belongs to the dual space $(V_k)^\star$: we link with the theory of **Reproducing Kernel Hilbert Spaces.**

Integration of k-smooth vector flows

Assume that (v_t) is a time-varying vector field such that

$$\ell_k(v)^2 \;=\; \int_0^1 \|v_t\|_k^2 \,\mathrm{d}t \;<\; \infty. \tag{26}$$

According to Picard-Lindelöf theorem, we can integrate the flow, find a unique trajectory φ_t of diffeomorphisms such that for every point $x \in \mathbb{R}^D$ and time $t \in [0, 1]$:

$$\varphi_0(x) = x \qquad \text{and} \qquad \frac{\mathrm{d}}{\mathrm{d}t}\,[\varphi_t(x)] = v_t \circ \varphi_t(x),$$

$$\text{i.e.} \qquad \varphi_0 \;=\; \mathrm{Id}_{\mathbb{R}^D} \qquad \text{and} \qquad \varphi_t \;=\; \mathrm{Id}_{\mathbb{R}^D} + \int_{s=0}^{t} v_s \circ \varphi_s \,\mathrm{d}s.$$

Integration of k-smooth vector flows

Assume that (v_t) is a time-varying vector field such that

$$\ell_k(v)^2 \;=\; \int_0^1 \|v_t\|_k^2 \,\mathrm{d}t \;<\; \infty. \tag{26}$$

According to Picard-Lindelöf theorem, we can integrate the flow, find a unique trajectory φ_t of diffeomorphisms such that for every point $x \in \mathbb{R}^D$ and time $t \in [0, 1]$:

$$\varphi_0(x) = x \qquad \text{and} \qquad \frac{\mathrm{d}}{\mathrm{d}t}\,[\varphi_t(x)] = v_t \circ \varphi_t(x),$$

$$\text{i.e.} \qquad \varphi_0 \;=\; \mathrm{Id}_{\mathbb{R}^D} \qquad \text{and} \qquad \varphi_t \;=\; \mathrm{Id}_{\mathbb{R}^D} \;+\; \int_{s=0}^{t} v_s \circ \varphi_s \,\mathrm{d}s.$$

An infinite-dimensional matching problem

We define $G_k = \{\varphi_1 \mid \cdots\}$ the set of diffeomorphisms obtained by integrating *finite-cost* vector flows $(v_t) \in L^2(V_k)$.

G_k is an infinite-dimensional Riemannian manifold modeled on V_k. As diffeomorphisms carry around images and measures, we try to minimize

$$C^2(\varphi_1) \;=\; \ell_k(v)^2 \;=\; \int_0^1 \|v_t\|_k^2 \,\mathrm{d}t \;<\; \infty \tag{27}$$

under the constraint that

$$X \xrightarrow{\varphi_1} Y. \tag{28}$$

An infinite-dimensional matching problem

We define $G_k = \{\varphi_1 \mid \cdots\}$ the set of diffeomorphisms obtained by integrating *finite-cost* vector flows $(v_t) \in L^2(V_k)$.

G_k is an infinite-dimensional Riemannian manifold modeled on V_k. As diffeomorphisms carry around images and measures, we try to minimize

$$C^2(\varphi_1) = \ell_k(v)^2 = \int_0^1 \|v_t\|_k^2 \,\mathrm{d}t < \infty \tag{27}$$

under the constraint that

$$X \xrightarrow{\varphi_1} Y. \tag{28}$$

An infinite-dimensional matching problem

We define $G_k = \{\varphi_1 \mid \cdots\}$ the set of diffeomorphisms obtained by integrating *finite-cost* vector flows $(v_t) \in L^2(V_k)$.

G_k is an infinite-dimensional Riemannian manifold modeled on V_k. As diffeomorphisms carry around images and measures, we try to minimize

$$C^2(\varphi_1) \;=\; \ell_k(v)^2 \;=\; \int_0^1 \|v_t\|_k^2 \,\mathrm{d}t \;<\; \infty \tag{27}$$

under the constraint that

$$X \xrightarrow{\varphi_1} Y. \tag{28}$$

The kernel and diffeomorphic geodesics coincide

Reduction Principle

Let q_t be a time-dependent point cloud, k a kernel function.
Then, the two propositions below are equivalent :

i) q_t is a geodesic for the kernel cometric K_q, with momentum p_t associated to the Hamiltonian

$$H(q, p) \;=\; \frac{1}{2} p^\top K_q p. \tag{29}$$

ii) q_t is carried around by a locally optimal diffeomorphic trajectory $\varphi_t = \text{Flow}(v_t)$, and we have

$$v_t \;=\; k \star p_t \qquad \text{i.e.} \qquad v_t(x) \;=\; \sum_{m=1}^{M} k(q_t^m, x)\, p_t^m. \tag{30}$$

Hand-waving proof of the reduction principle, part 1

At any time t,

$$v_t \;=\; \arg\min\{\, \|v\|_k \;|\; \forall\, m, v(q_t^m) = v_t(q_t^m) \,\}\,. \tag{31}$$

Hence, as v_t does not have any superfluous component,

$$v_t \in \{\, v \mid \forall\, m, v(q_t^m) = 0 \,\}^{\perp_k} \tag{32}$$

$$\text{i.e.} \quad v_t \in \left(\bigcap_{m=1}^{M} \{\, v \mid \langle \delta_{q_t^m}, v \rangle = 0 \,\} \right)^{\perp_k}. \tag{33}$$

But we also know that :

$$\langle k \star \delta_{q_t^m}, v \rangle_k \;=\; \int_{\omega \in \mathbb{R}^D} \frac{1}{\widehat{k}(\omega)} \overline{\widehat{k \star \delta_{q_t^m}}(\omega)} \cdot \widehat{v}(\omega) \;\mathrm{d}\omega \tag{34}$$

$$= \int_{\omega \in \mathbb{R}^D} \overline{\widehat{\delta_{q_t^m}}(\omega)} \cdot \widehat{v}(\omega) \;\mathrm{d}\omega \tag{35}$$

$$= \langle \delta_{q_t^m}, v \rangle \;=\; v(q_t^m). \tag{36}$$

Hand-waving proof of the reduction principle, part 1

At any time t,

$$v_t \;=\; \arg\min\{\, \|v\|_k \;|\; \forall\, m, v(q_t^m) = v_t(q_t^m)\,\}\,. \tag{31}$$

Hence, as v_t does not have any superfluous component,

$$v_t \in \{\, v \mid \forall\, m, v(q_t^m) = 0\,\}^{\perp_k} \tag{32}$$

$$\text{i.e.}\quad v_t \in \left(\bigcap_{m=1}^{M} \{\, v \mid \langle \delta_{q_t^m}, v \rangle = 0 \,\} \right)^{\perp_k}. \tag{33}$$

But we also know that :

$$\langle k \star \delta_{q_t^m}, v \rangle_k \;=\; \int_{\omega\in\mathbb{R}^D} \frac{1}{\widehat{k}(\omega)}\, \overline{\widehat{k \star \delta_{q_t^m}}(\omega)} \cdot \widehat{v}(\omega)\; \mathrm{d}\omega \tag{34}$$

$$= \int_{\omega\in\mathbb{R}^D} \overline{\widehat{\delta_{q_t^m}}(\omega)} \cdot \widehat{v}(\omega)\; \mathrm{d}\omega \tag{35}$$

$$= \langle \delta_{q_t^m}, v \rangle \;=\; v(q_t^m). \tag{36}$$

Hand-waving proof of the reduction principle, part 1

At any time t,

$$v_t \;=\; \arg\min\{\, \|v\|_k \;|\; \forall\, m, v(q_t^m) = v_t(q_t^m) \,\}\,. \tag{31}$$

Hence, as v_t does not have any superfluous component,

$$v_t \in \{\, v \mid \forall\, m, v(q_t^m) = 0 \,\}^{\perp_k} \tag{32}$$

$$\text{i.e.} \quad v_t \in \left(\bigcap_{m=1}^{M} \{\, v \mid \langle \delta_{q_t^m}, v \rangle = 0 \,\} \right)^{\perp_k}. \tag{33}$$

But we also know that :

$$\langle k \star \delta_{q_t^m}, v \rangle_k \;=\; \int_{\omega\in\mathbb{R}^D} \frac{1}{\widehat{k}(\omega)}\, \overline{\widehat{k \star \delta_{q_t^m}}(\omega)} \cdot \widehat{v}(\omega)\;\mathrm{d}\omega \tag{34}$$

$$= \int_{\omega\in\mathbb{R}^D} \overline{\widehat{\delta_{q_t^m}}(\omega)} \cdot \widehat{v}(\omega)\;\mathrm{d}\omega \tag{35}$$

$$= \langle \delta_{q_t^m}, v \rangle \;=\; v(q_t^m). \tag{36}$$

Hand-waving proof of the reduction principle, part 2

Hence why, at any time t,

$$v_t \in \left(\bigcap_{m=1}^{M} \left\{ v \mid \left\langle k \star \delta_{q_t^m}, v \right\rangle_k = 0 \right\} \right)^{\perp_k} \tag{37}$$

$$= \bigcup_{m=1}^{M} \left(k \star \delta_{q_t^m} \right)^{\perp_k \perp_k} \tag{38}$$

$$= \mathrm{Vect}\left(k \star \delta_{q_t^m},\ m \in [\![1, M]\!] \right). \tag{39}$$

So, one can write

$$v_t = k \star \left(\sum_{m=1}^{M} p_t^m \, \delta_{q_t^m} \right) = k \star p_t, \tag{40}$$

and

$$\|v_t\|_k^2 = \left\langle k \star p_t, k^{(-1)} \star k \star p_t \right\rangle = \langle k \star p_t, p_t \rangle = p_t^\top K_{q_t} p_t. \tag{41}$$

Hand-waving proof of the reduction principle, part 2

Hence why, at any time t,

$$v_t \in \left(\bigcap_{m=1}^{M} \left\{ v \mid \left\langle k \star \delta_{q_t^m}, v \right\rangle_k = 0 \right\} \right)^{\perp_k} \tag{37}$$

$$= \bigcup_{m=1}^{M} \left(k \star \delta_{q_t^m} \right)^{\perp_k \perp_k} \tag{38}$$

$$= \mathrm{Vect}\left(k \star \delta_{q_t^m},\ m \in [\![1, M]\!] \right). \tag{39}$$

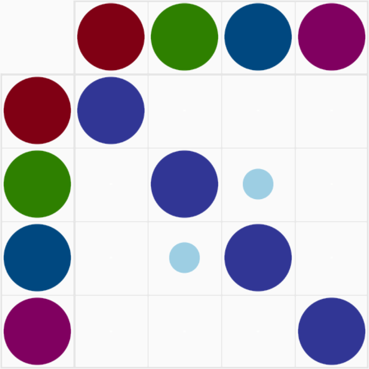
So, one can write

$$v_t \;=\; k \star \left(\sum_{m=1}^{M} p_t^m \, \delta_{q_t^m} \right) \;=\; k \star p_t, \tag{40}$$

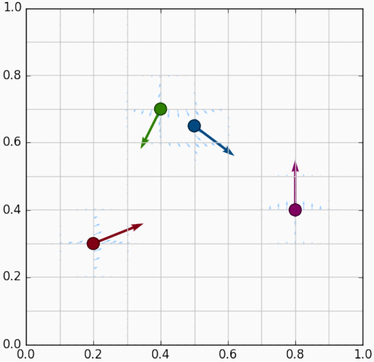
and

$$\|v_t\|_k^2 \;=\; \left\langle k \star p_t, k^{(-1)} \star k \star p_t \right\rangle \;=\; \langle k \star p_t, p_t \rangle \;=\; p_t^{\top} K_{q_t} p_t. \tag{41}$$

Influence of the kernel width, $\sigma = .25$



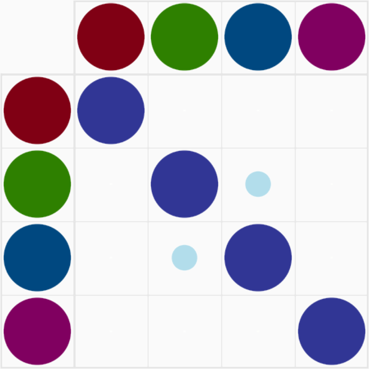
(a) Kernel matrix k_{q_t}.



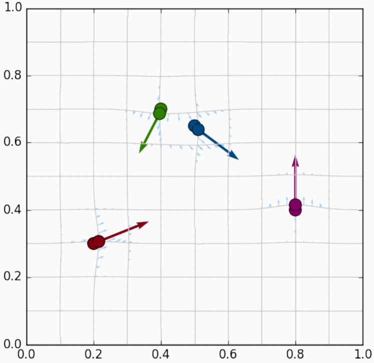
(b) Shooted cloud (q_t, p_t).

Figure 21: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2/2\sigma^2)$, $\sigma = .25$.

Influence of the kernel width, $\sigma = .25$



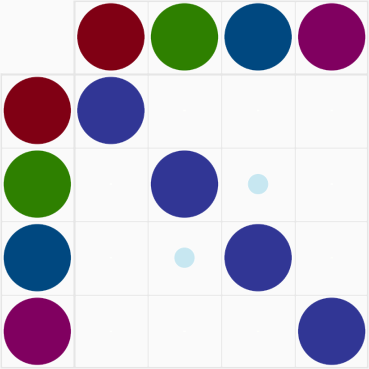
(a) Kernel matrix k_{q_t}.



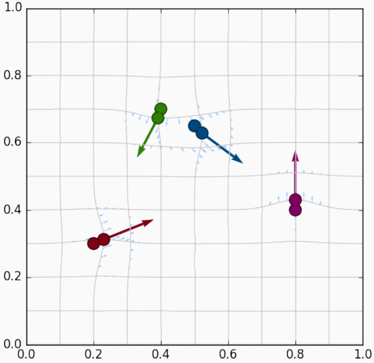
(b) Shooted cloud (q_t, p_t).

Figure 21: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2 / 2\sigma^2)$, $\sigma = .25$.

Influence of the kernel width, $\sigma = .25$



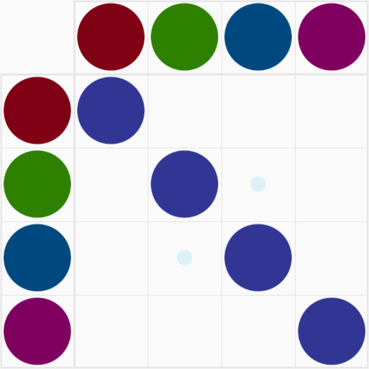
(a) Kernel matrix k_{q_t}.



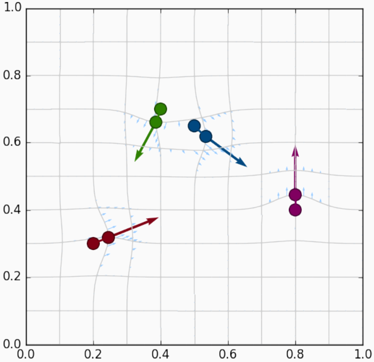
(b) Shooted cloud (q_t, p_t).

Figure 21: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2/2\sigma^2)$, $\sigma = .25$.

Influence of the kernel width, $\sigma = .25$



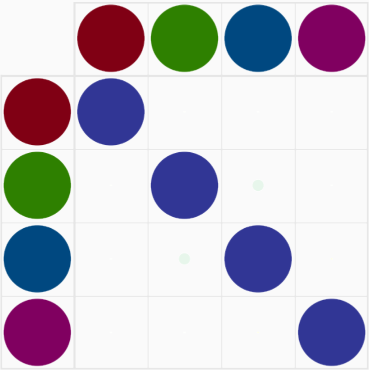
(a) Kernel matrix k_{q_t}.



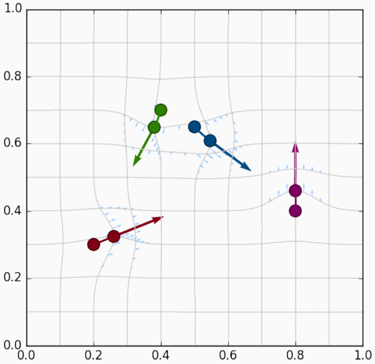
(b) Shooted cloud (q_t, p_t).

Figure 21: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2 / 2\sigma^2)$, $\sigma = .25$.

Influence of the kernel width, $\sigma = .25$



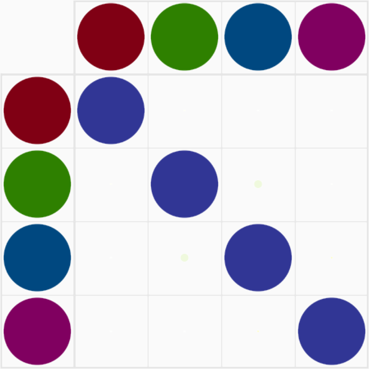
(a) Kernel matrix k_{q_t}.



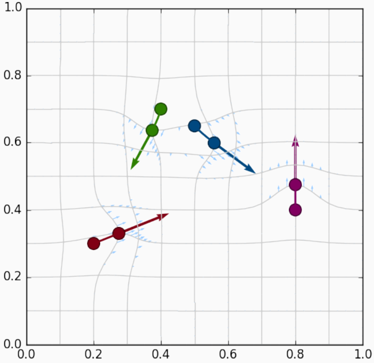
(b) Shooted cloud (q_t, p_t).

Figure 21: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2 / 2\sigma^2)$, $\sigma = .25$.

Influence of the kernel width, $\sigma = .25$



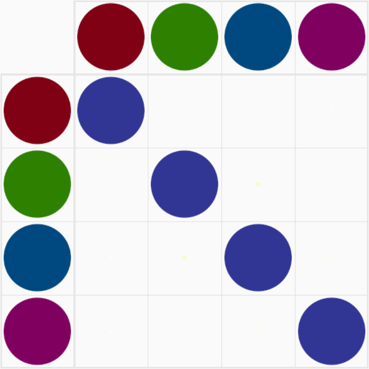
(a) Kernel matrix k_{q_t}.



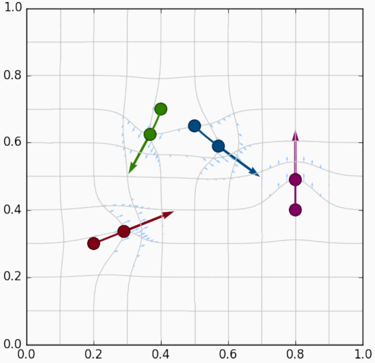
(b) Shooted cloud (q_t, p_t).

Figure 21: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2/2\sigma^2)$, $\sigma = .25$.

Influence of the kernel width, $\sigma = .25$



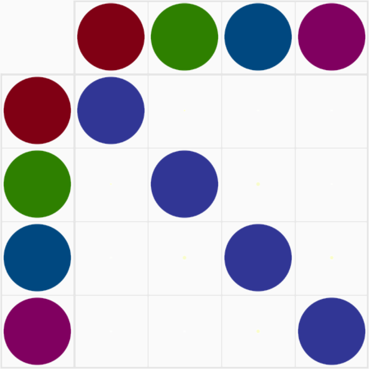
(a) Kernel matrix k_{q_t}.



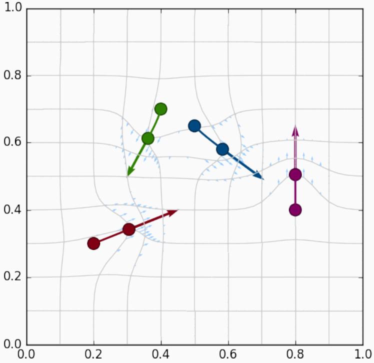
(b) Shooted cloud (q_t, p_t).

Figure 21: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2/2\sigma^2)$, $\sigma = .25$.

Influence of the kernel width, $\sigma = .25$



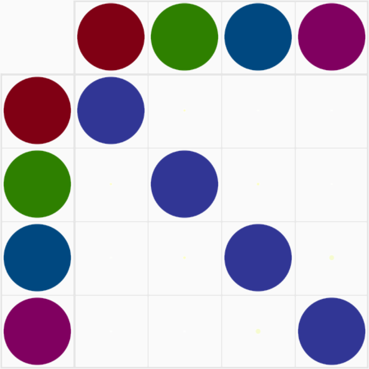
(a) Kernel matrix k_{q_t}.



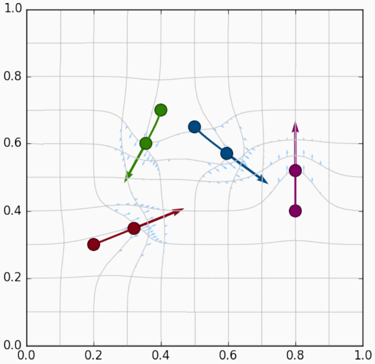
(b) Shooted cloud (q_t, p_t).

Figure 21: Geodesic shooting, $k(x - y) = \exp(-\|x - y\|^2 / 2\sigma^2)$, $\sigma = .25$.

Influence of the kernel width, $\sigma = .25$



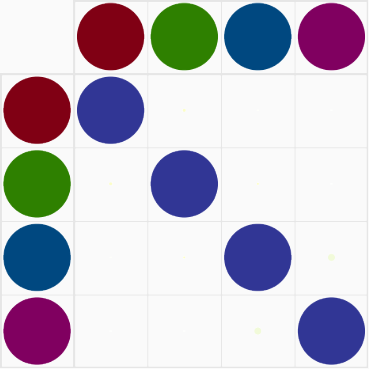
(a) Kernel matrix k_{q_t}.



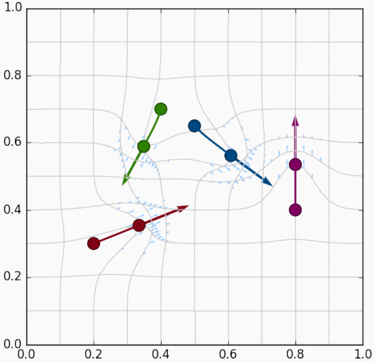
(b) Shooted cloud (q_t, p_t).

Figure 21: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2/2\sigma^2)$, $\sigma = .25$.

Influence of the kernel width, $\sigma = .25$



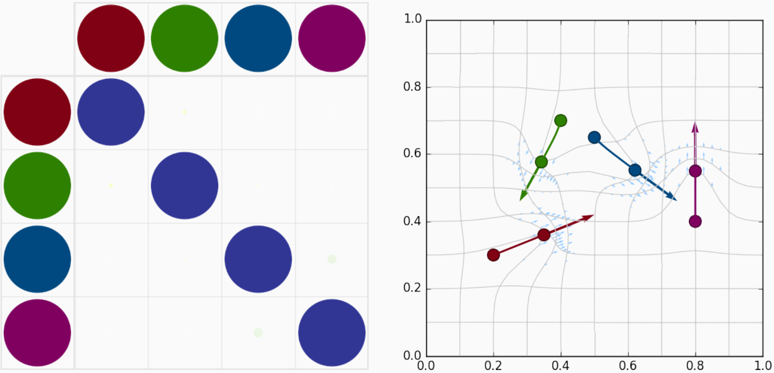
(a) Kernel matrix k_{q_t}.



(b) Shooted cloud (q_t, p_t).

Figure 21: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2/2\sigma^2)$, $\sigma = .25$.

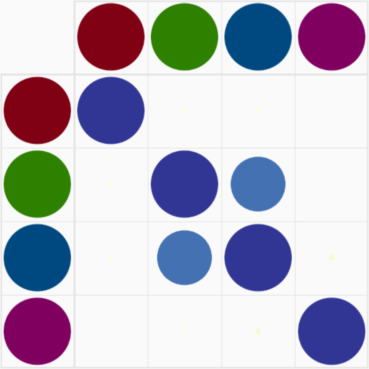
Influence of the kernel width, $\sigma = .25$



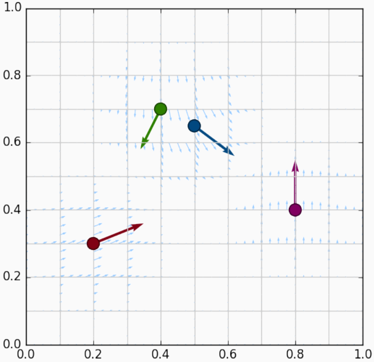
(a) Kernel matrix k_{q_t}. **(b)** Shooted cloud (q_t, p_t).

Figure 21: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2 / 2\sigma^2)$, $\sigma = .25$.

Influence of the kernel width, $\sigma = .35$



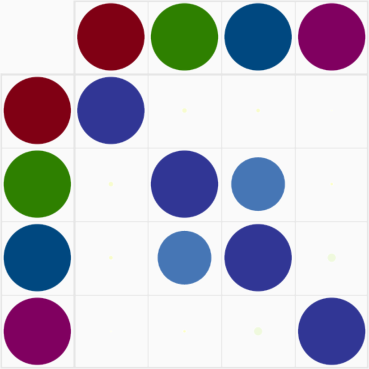
(a) Kernel matrix k_{q_t}.



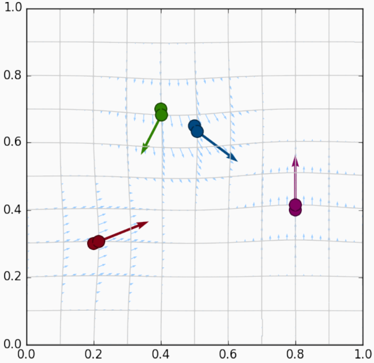
(b) Shooted cloud (q_t, p_t).

Figure 22: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2 / 2\sigma^2)$, $\sigma = .35$.

Influence of the kernel width, $\sigma = .35$



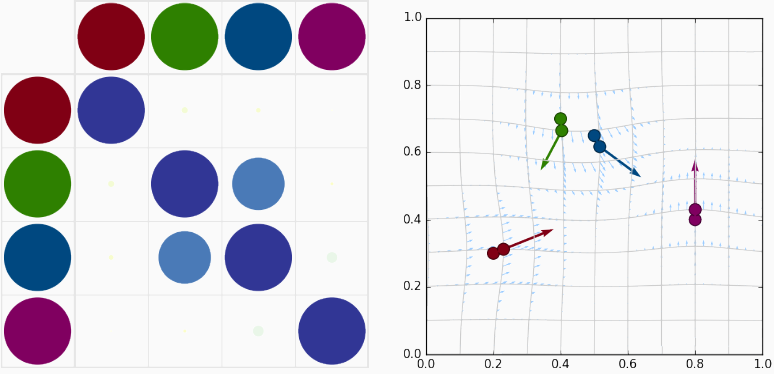
(a) Kernel matrix k_{q_t}.



(b) Shooted cloud (q_t, p_t).

Figure 22: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2 / 2\sigma^2)$, $\sigma = .35$.

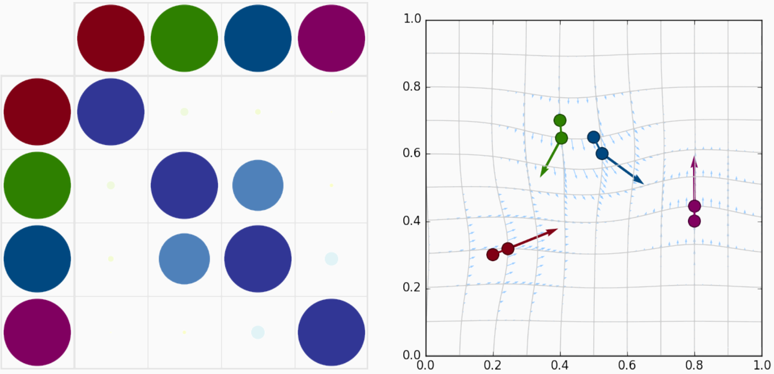
Influence of the kernel width, $\sigma = .35$



(a) Kernel matrix k_{q_t}. **(b)** Shooted cloud (q_t, p_t).

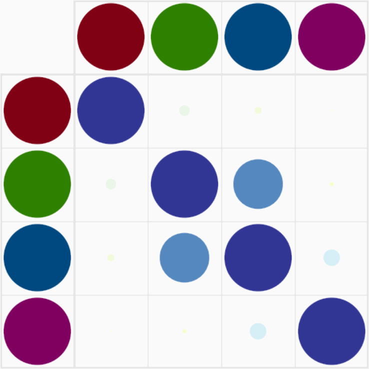
Figure 22: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2/2\sigma^2)$, $\sigma = .35$.

Influence of the kernel width, $\sigma = .35$

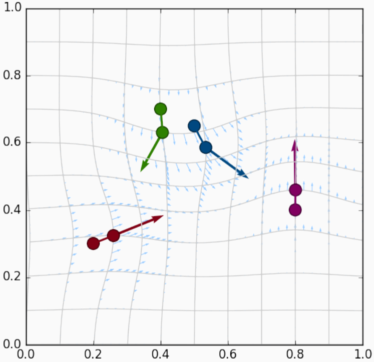


(a) Kernel matrix k_{q_t}. **(b)** Shooted cloud (q_t, p_t).

Figure 22: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2/2\sigma^2)$, $\sigma = .35$.



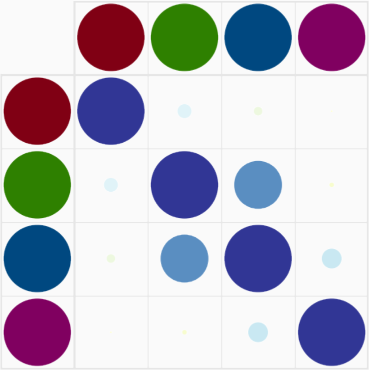
(a) Kernel matrix k_{q_t}.



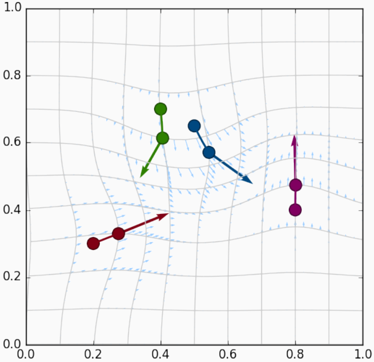
(b) Shooted cloud (q_t, p_t).

Figure 22: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2 / 2\sigma^2)$, $\sigma = .35$.

Influence of the kernel width, $\sigma = .35$



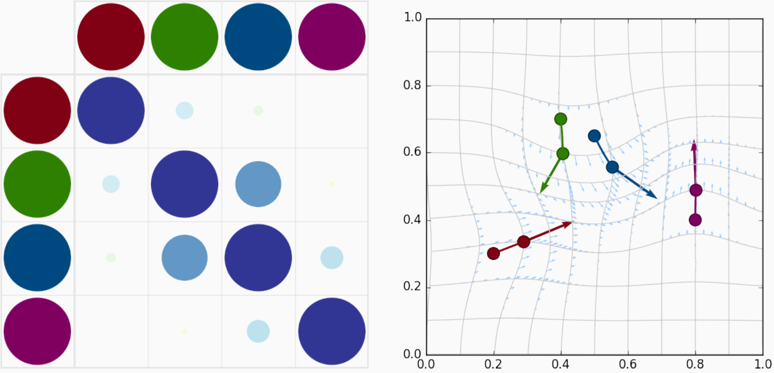
(a) Kernel matrix k_{q_t}.



(b) Shooted cloud (q_t, p_t).

Figure 22: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2 / 2\sigma^2)$, $\sigma = .35$.

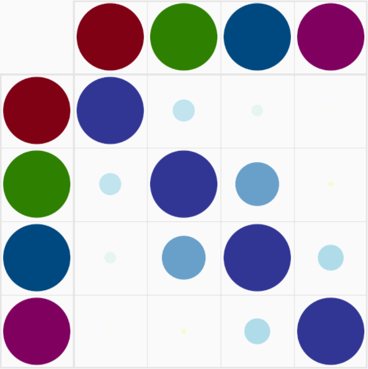
Influence of the kernel width, $\sigma = .35$



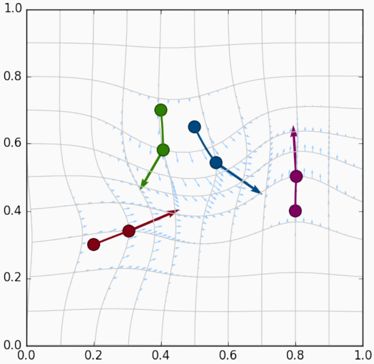
(a) Kernel matrix k_{q_t}. **(b)** Shooted cloud (q_t, p_t).

Figure 22: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2 / 2\sigma^2)$, $\sigma = .35$.

Influence of the kernel width, $\sigma = .35$



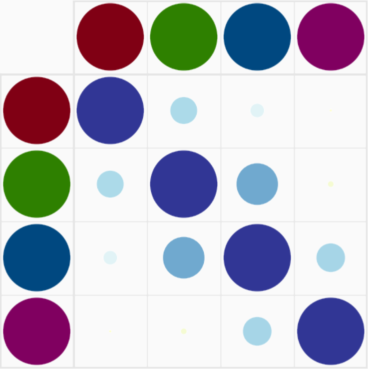
(a) Kernel matrix k_{q_t}.



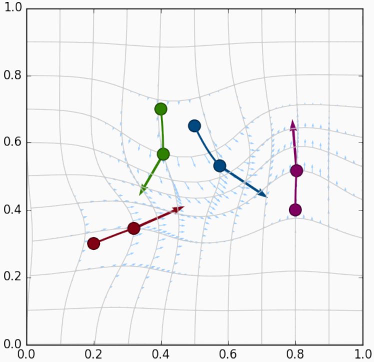
(b) Shooted cloud (q_t, p_t).

Figure 22: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2 / 2\sigma^2)$, $\sigma = .35$.

Influence of the kernel width, $\sigma = .35$



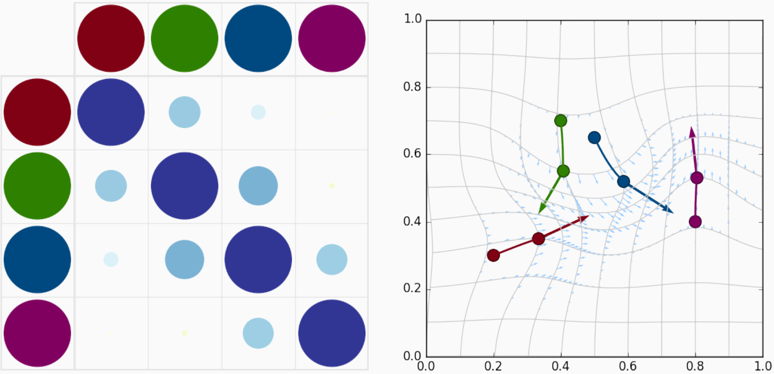
(a) Kernel matrix k_{q_t}.



(b) Shooted cloud (q_t, p_t).

Figure 22: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2/2\sigma^2)$, $\sigma = .35$.

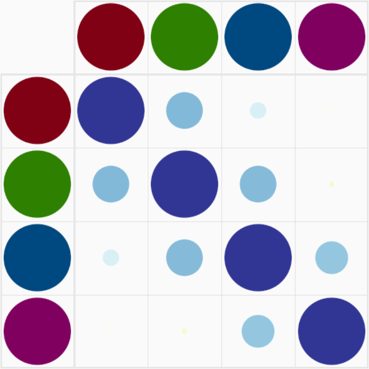
Influence of the kernel width, $\sigma = .35$



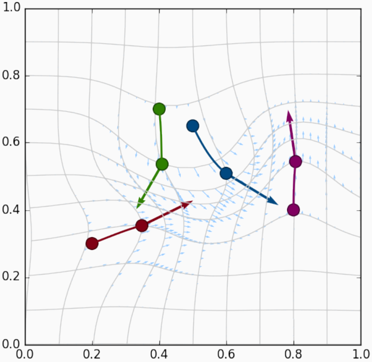
(a) Kernel matrix k_{q_t}. **(b)** Shooted cloud (q_t, p_t).

Figure 22: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2 / 2\sigma^2)$, $\sigma = .35$.

Influence of the kernel width, $\sigma = .35$



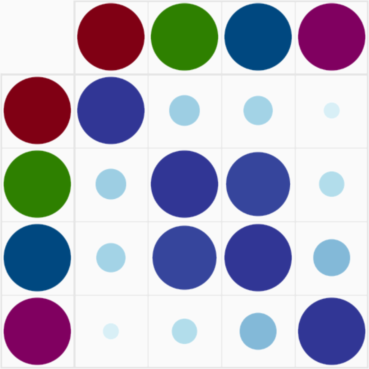
(a) Kernel matrix k_{q_t}.



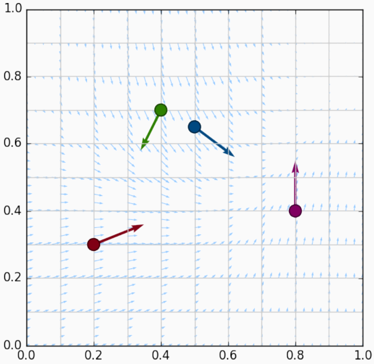
(b) Shooted cloud (q_t, p_t).

Figure 22: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2/2\sigma^2)$, $\sigma = .35$.

Influence of the kernel width, $\sigma = .50$



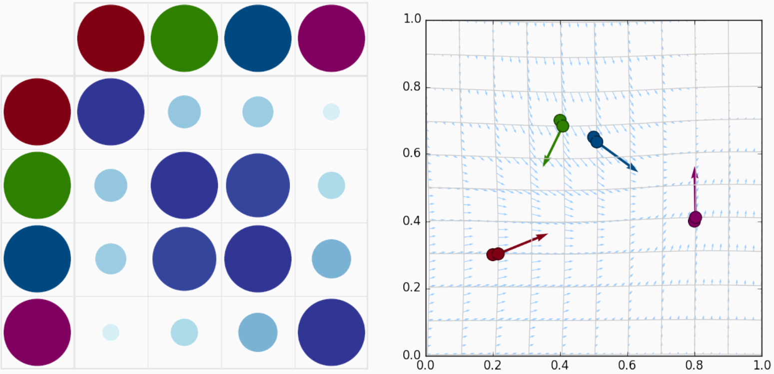
(a) Kernel matrix k_{q_t}.



(b) Shooted cloud (q_t, p_t).

Figure 23: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2 / 2\sigma^2)$, $\sigma = .50$.

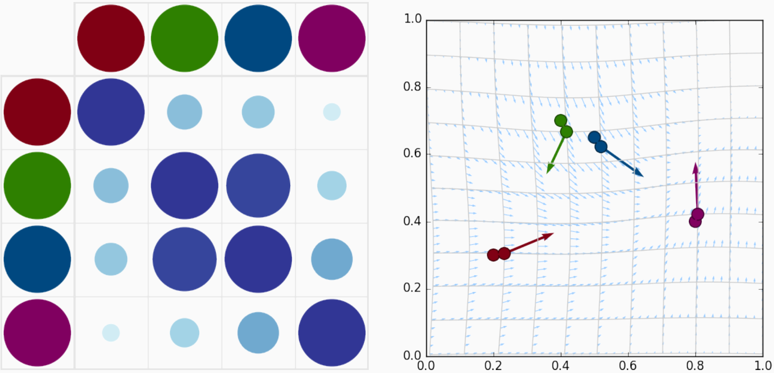
Influence of the kernel width, $\sigma = .50$



(a) Kernel matrix k_{q_t}. **(b)** Shooted cloud (q_t, p_t).

Figure 23: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2 / 2\sigma^2)$, $\sigma = .50$.

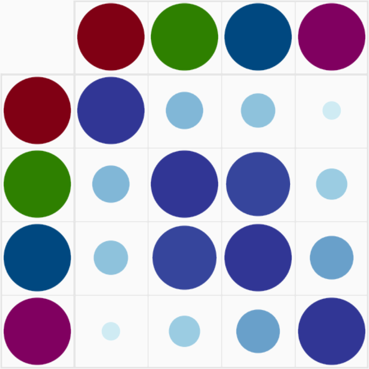
Influence of the kernel width, $\sigma = .50$



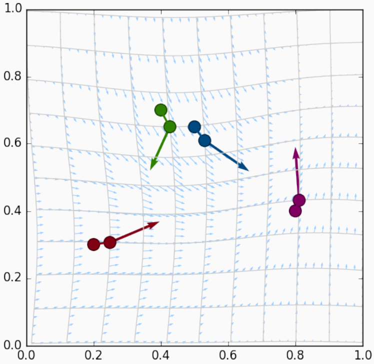
(a) Kernel matrix k_{q_t}. **(b)** Shooted cloud (q_t, p_t).

Figure 23: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2 / 2\sigma^2)$, $\sigma = .50$.

Influence of the kernel width, $\sigma = .50$



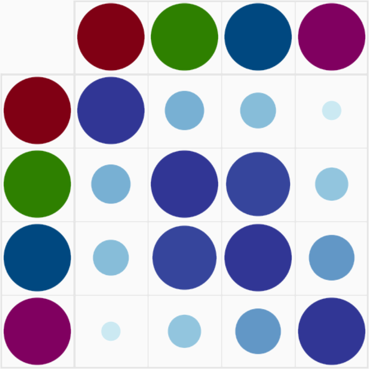
(a) Kernel matrix k_{q_t}.



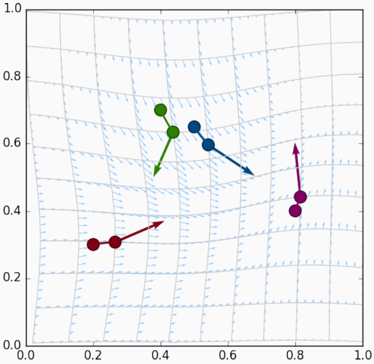
(b) Shooted cloud (q_t, p_t).

Figure 23: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2 / 2\sigma^2)$, $\sigma = .50$.

Influence of the kernel width, $\sigma = \texttt{.50}$



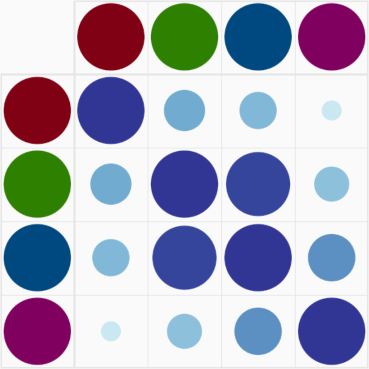
(a) Kernel matrix k_{q_t}.



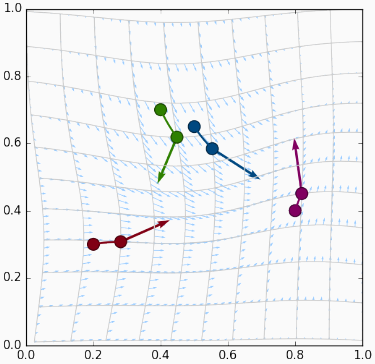
(b) Shooted cloud (q_t, p_t).

Figure 23: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2 / 2\sigma^2)$, $\sigma = \texttt{.50}$.

Influence of the kernel width, $\sigma = .50$



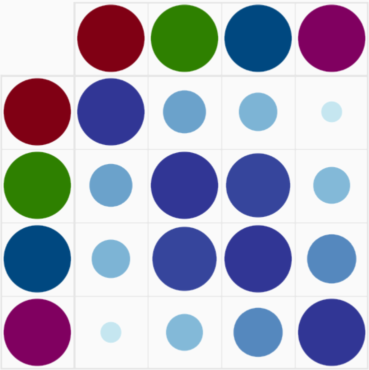
(a) Kernel matrix k_{q_t}.



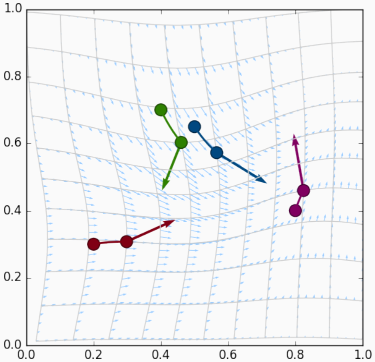
(b) Shooted cloud (q_t, p_t).

Figure 23: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2/2\sigma^2)$, $\sigma = .50$.

Influence of the kernel width, $\sigma = \texttt{.50}$



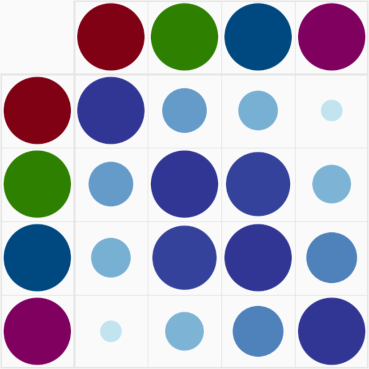
(a) Kernel matrix k_{q_t}.



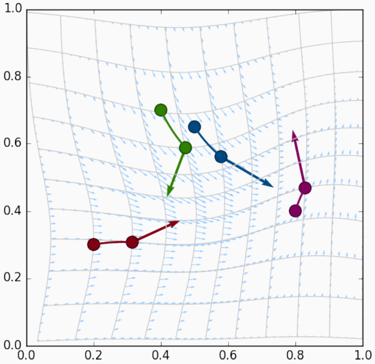
(b) Shooted cloud (q_t, p_t).

Figure 23: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2 / 2\sigma^2)$, $\sigma = \texttt{.50}$.

Influence of the kernel width, $\sigma = .50$



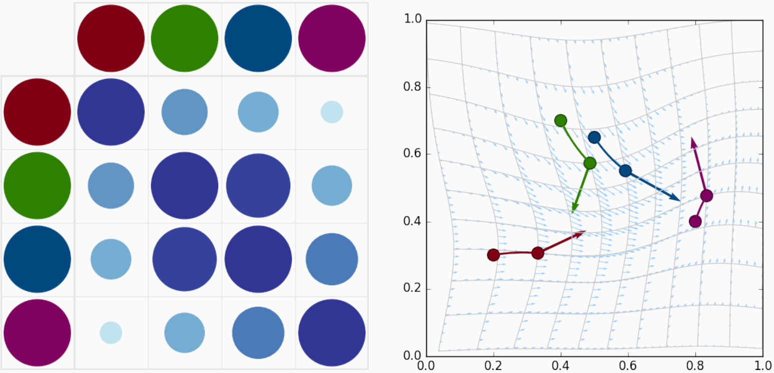
(a) Kernel matrix k_{q_t}.



(b) Shooted cloud (q_t, p_t).

Figure 23: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2 / 2\sigma^2)$, $\sigma = .50$.

Influence of the kernel width, $\sigma = .50$

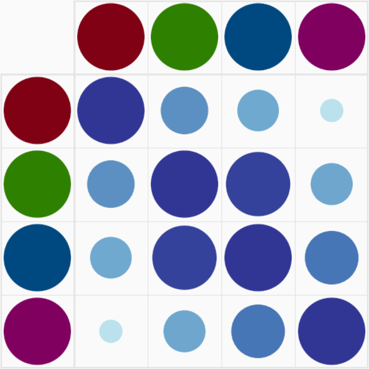


(a) Kernel matrix k_{q_t}.

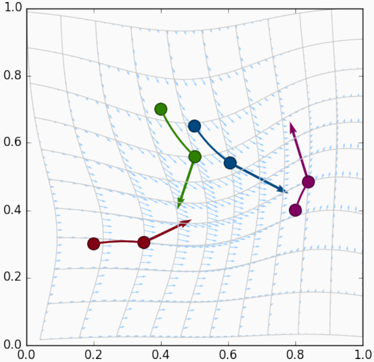
(b) Shooted cloud (q_t, p_t).

Figure 23: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2 / 2\sigma^2)$, $\sigma = .50$.

Influence of the kernel width, $\sigma = \texttt{.50}$



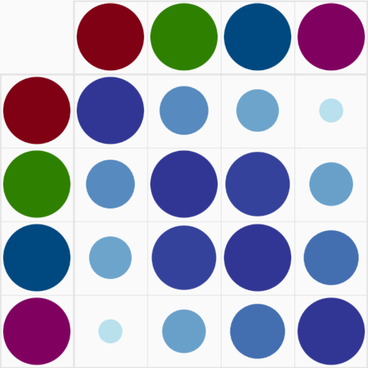
(a) Kernel matrix k_{q_t}.



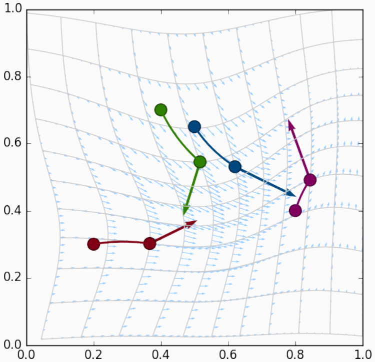
(b) Shooted cloud (q_t, p_t).

Figure 23: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2 / 2\sigma^2)$, $\sigma = \texttt{.50}$.

Influence of the kernel width, $\sigma = .50$



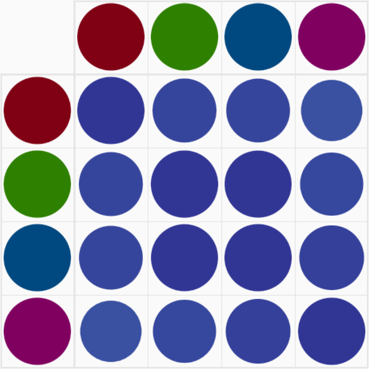
(a) Kernel matrix k_{q_t}.



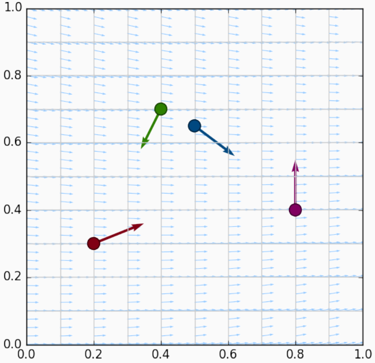
(b) Shooted cloud (q_t, p_t).

Figure 23: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2 / 2\sigma^2)$, $\sigma = .50$.

Influence of the kernel width, $\sigma = 1.$



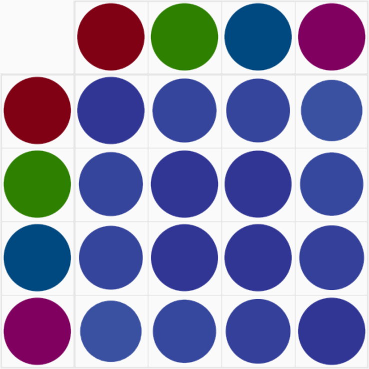
(a) Kernel matrix k_{q_t}.



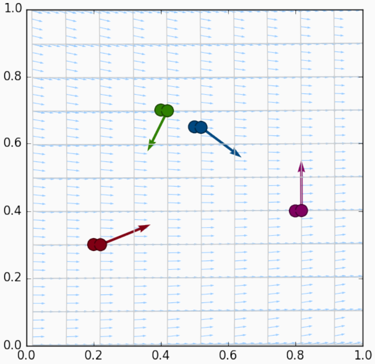
(b) Shooted cloud (q_t, p_t).

Figure 24: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2/2\sigma^2)$, $\sigma = 1..$

Influence of the kernel width, $\sigma = 1$.



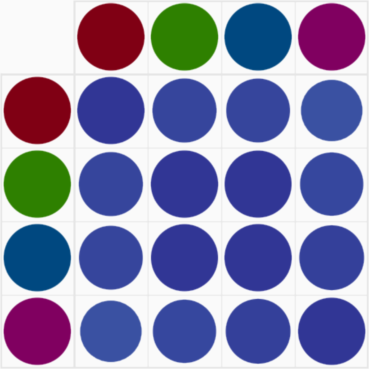
(a) Kernel matrix k_{q_t}.



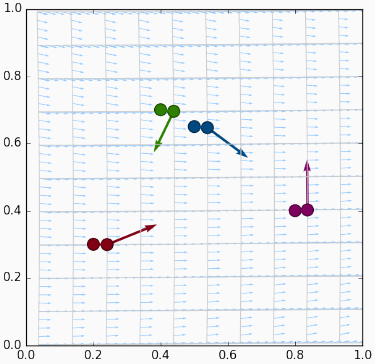
(b) Shooted cloud (q_t, p_t).

Figure 24: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2 / 2\sigma^2)$, $\sigma = 1.$.

Influence of the kernel width, $\sigma = 1.$



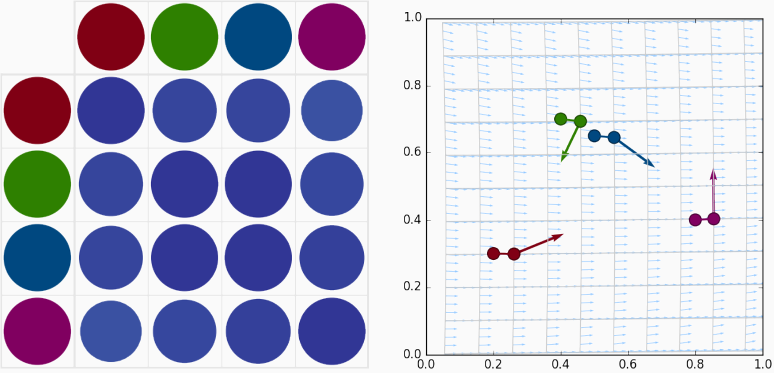
(a) Kernel matrix k_{q_t}.



(b) Shooted cloud (q_t, p_t).

Figure 24: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2/2\sigma^2)$, $\sigma = 1..$

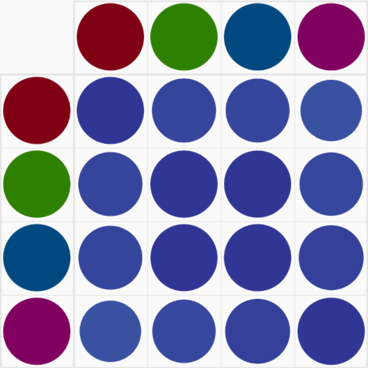
Influence of the kernel width, $\sigma = 1.$



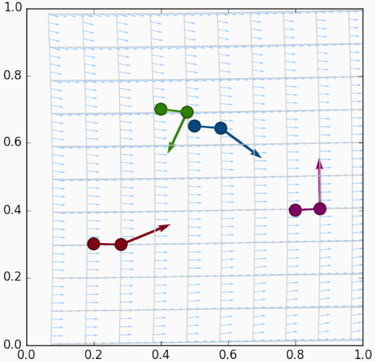
(a) Kernel matrix k_{q_t}.   **(b)** Shooted cloud (q_t, p_t).

Figure 24: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2/2\sigma^2)$, $\sigma = 1.$.

Influence of the kernel width, $\sigma = 1.$



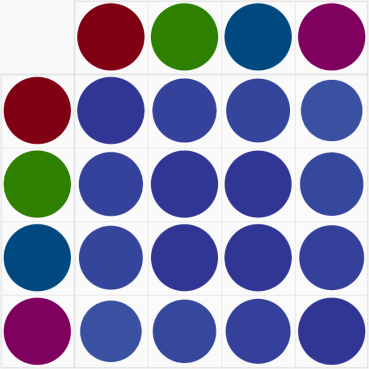
(a) Kernel matrix k_{q_t}.



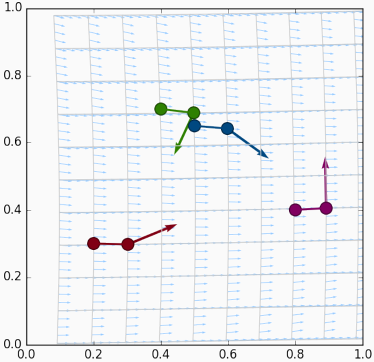
(b) Shooted cloud (q_t, p_t).

Figure 24: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2/2\sigma^2)$, $\sigma = 1.$.

Influence of the kernel width, $\sigma = 1.$



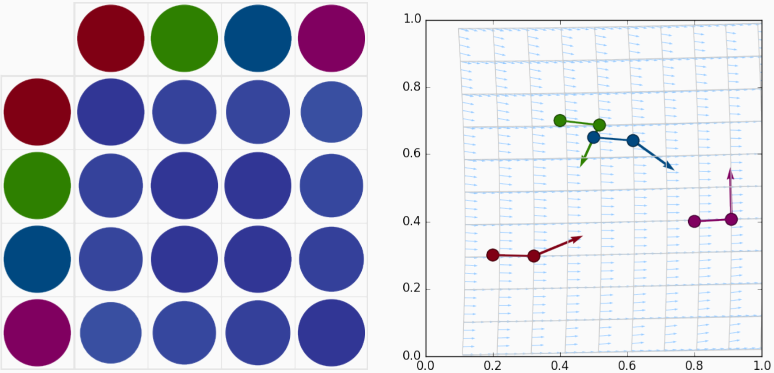
(a) Kernel matrix k_{q_t}.



(b) Shooted cloud (q_t, p_t).

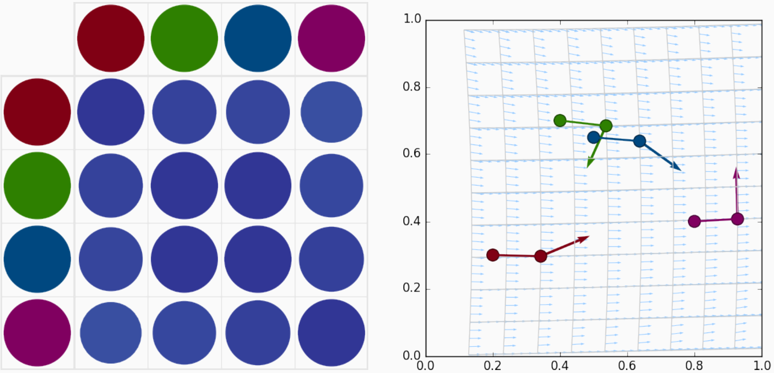
Figure 24: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2 / 2\sigma^2)$, $\sigma = 1.$.

Influence of the kernel width, $\sigma = 1.$



(a) Kernel matrix k_{q_t}. **(b)** Shooted cloud (q_t, p_t).

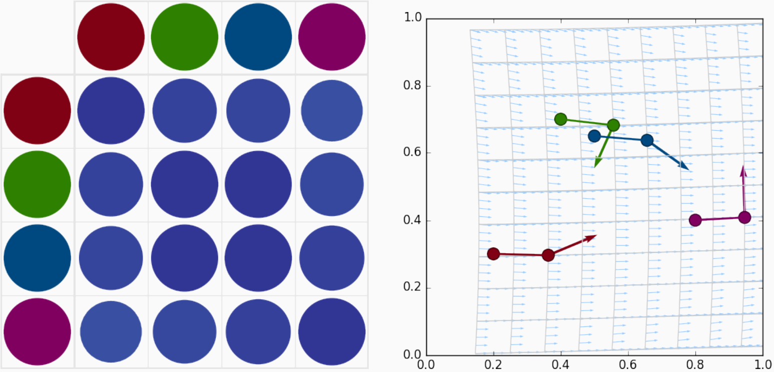
Figure 24: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2/2\sigma^2)$, $\sigma = 1.$.



(a) Kernel matrix k_{q_t}. **(b)** Shooted cloud (q_t, p_t).

Figure 24: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2/2\sigma^2)$, $\sigma = 1.$.

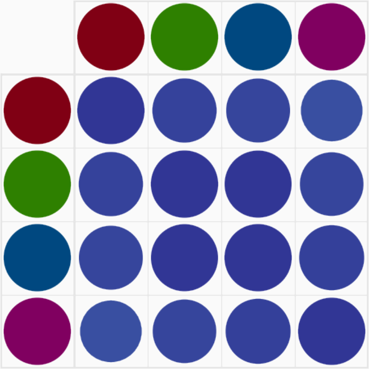
Influence of the kernel width, $\sigma = 1.$



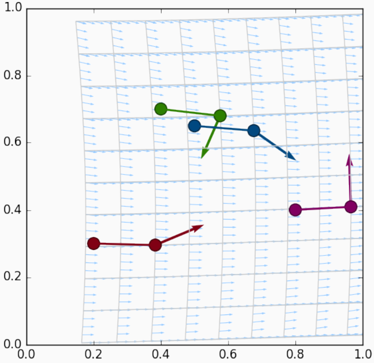
(a) Kernel matrix k_{q_t}. **(b)** Shooted cloud (q_t, p_t).

Figure 24: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2 / 2\sigma^2)$, $\sigma = 1.$.

Influence of the kernel width, $\sigma = 1.$



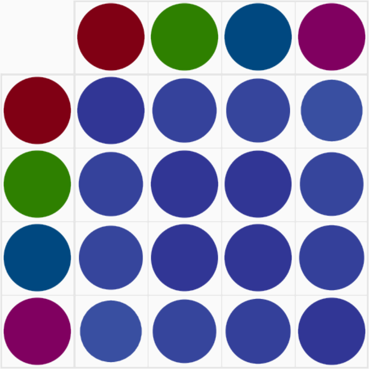
(a) Kernel matrix k_{q_t}.



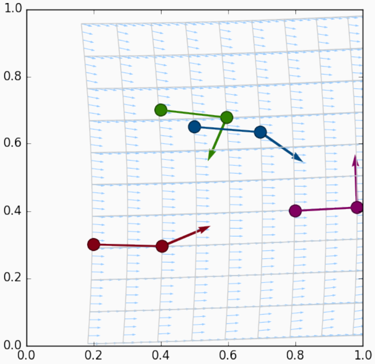
(b) Shooted cloud (q_t, p_t).

Figure 24: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2/2\sigma^2)$, $\sigma = 1.$.

Influence of the kernel width, $\sigma = 1.$



(a) Kernel matrix k_{q_t}.



(b) Shooted cloud (q_t, p_t).

Figure 24: Geodesic shooting, $k(x-y) = \exp(-\|x-y\|^2/2\sigma^2)$, $\sigma = 1.$.

Conclusion

We have now presented the *Large Deformation Diffeomorphic Metric Mapping*, or LDDMM setting :

- OT $(\sigma = 0) \xrightarrow{\sigma++} G_k \xrightarrow{\sigma++} (\sigma = +\infty)$ Translations
- Deformations computed through geodesic shooting

The (basic) framework relies on three pillars :

- Hamilton's theorem $(g_q \longrightarrow K_q)$
- The current availability of GPUs (parallelism)
- The Reduction Principle $((q_t, p_t) \longleftrightarrow \varphi_t)$

Conclusion

We have now presented the *Large Deformation Diffeomorphic Metric Mapping*, or LDDMM setting :

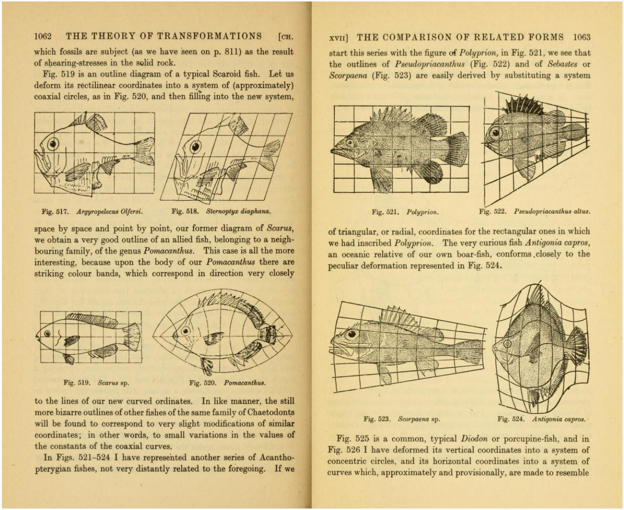
- OT $\quad (\sigma = 0) \xrightarrow{\sigma++} G_k \xrightarrow{\sigma++} (\sigma = +\infty)$ Translations
- Deformations computed through geodesic shooting

The (basic) framework relies on three pillars :

- Hamilton's theorem $\qquad (g_q \longrightarrow K_q)$
- The current availability of GPUs $\qquad$ (parallelism)
- The Reduction Principle $\qquad ((q_t, p_t) \longleftrightarrow \varphi_t)$

Conclusion

We can now emulate D’Arcy Thompson’s work



1062 THE THEORY OF TRANSFORMATIONS [CH.

which fossils are subject (as we have seen on p. 811) as the result of shearing-stresses in the solid rock.

Fig. 519 is an outline diagram of a typical Scaroid fish. Let us deform its rectilinear coordinates into a system of (approximately) coaxial circles, as in Fig. 520, and then filling into the new system,

Fig. 517. *Argyropelecus Olfersi.* Fig. 518. *Sternoptyx diaphana.*

space by space and point by point, our former diagram of *Scarus*, we obtain a very good outline of an allied fish, belonging to a neighbouring family, of the genus *Pomacanthus*. This case is all the more interesting, because upon the body of our *Pomacanthus* there are striking colour bands, which correspond in direction very closely

Fig. 519. *Scarus* sp. Fig. 520. *Pomacanthus.*

to the lines of our new curved ordinates. In like manner, the still more bizarre outlines of other fishes of the same family of Chaetodonts will be found to correspond to very slight modifications of similar coordinates; in other words, to small variations in the values of the constants of the coaxial curves.

In Figs. 521–524 I have represented another series of Acanthopterygian fishes, not very distantly related to the foregoing. If we

XVII] THE COMPARISON OF RELATED FORMS 1063

start this series with the figure of *Polyprion*, in Fig. 521, we see that the outlines of *Pseudopriacanthus* (Fig. 522) and of *Sebastes* or *Scorpaena* (Fig. 523) are easily derived by substituting a system

Fig. 521. *Polyprion.* Fig. 522. *Pseudopriacanthus altus.*

of triangular, or radial, coordinates for the rectangular ones in which we had inscribed *Polyprion*. The very curious fish *Antigonia capros*, an oceanic relative of our own boar-fish, conforms closely to the peculiar deformation represented in Fig. 524.

Fig. 523. *Scorpaena* sp. Fig. 524. *Antigonia capros.*

Fig. 525 is a common, typical *Diodon* or porcupine-fish, and in Fig. 526 I have deformed its vertical coordinates into a system of concentric circles, and its horizontal coordinates into a system of curves which, approximately and provisionally, are made to resemble

Figure 25: Excerpt from the seminal book of D’Arcy Wentworth Thompson (1860-1948), *On Growth and Forms*.

Statistics on a Riemannian manifold

Biologists, Neurologists and Physicians would like to conduct statistical surveys such as :

- Linear regression
- Mean computation + Principal Component Analysis
- Transport of tangential information

Problem : no meaningful algebraic structure $(+, \times)$ on shapes.

Given a mere Riemannian distance, we provide :

- Geodesic regression
- Fréchet Mean + PCA on shooting momentums
- Parallel transport

Statistics on a Riemannian manifold

Biologists, Neurologists and Physicians would like to conduct statistical surveys such as :

- Linear regression
- Mean computation + Principal Component Analysis
- Transport of tangential information

Problem : no meaningful algebraic structure $(+, \times)$ on shapes.

Given a mere Riemannian distance, we provide :

- Geodesic regression
- Fréchet Mean + PCA on shooting momentums
- Parallel transport

Statistics on a Riemannian manifold

Biologists, Neurologists and Physicians would like to conduct statistical surveys such as :

- Linear regression
- Mean computation + Principal Component Analysis
- Transport of tangential information

Problem : no meaningful algebraic structure $(+, \times)$ on shapes.

Given a mere Riemannian distance, we provide :

- Geodesic regression
- Fréchet Mean + PCA on shooting momentums
- Parallel transport

Transfer of anatomical data: animated silhouettes

Figure 26: Video presentation of the (non-LDDMM) paper *Anatomy Transfer Fast Forward*, Siggraph Asia 2013 by Ali-Hamadi, Liu, Gilles et al.

Transfer of anatomical data: medical applications

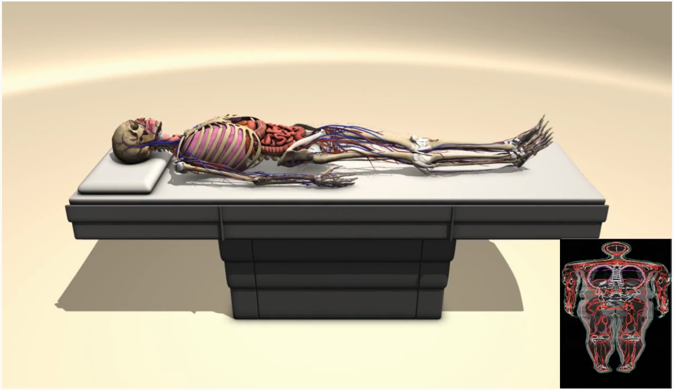


Figure 27: Video presentation of the (non-LDDMM) paper *Anatomy Transfer Fast Forward*, Siggraph Asia 2013 by Ali-Hamadi, Liu, Gilles et al.

Construction of anatomical atlases

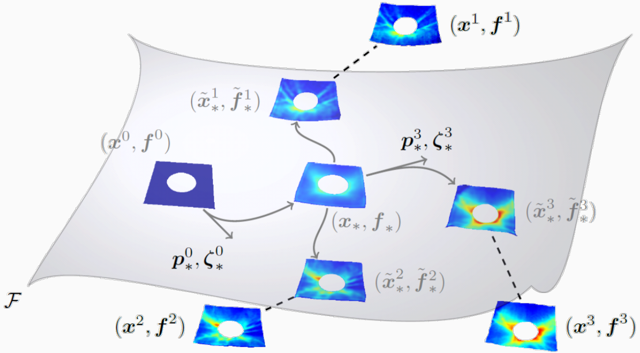


Figure 28: Building an atlas from the retina dataset [5].

A continuum of professions

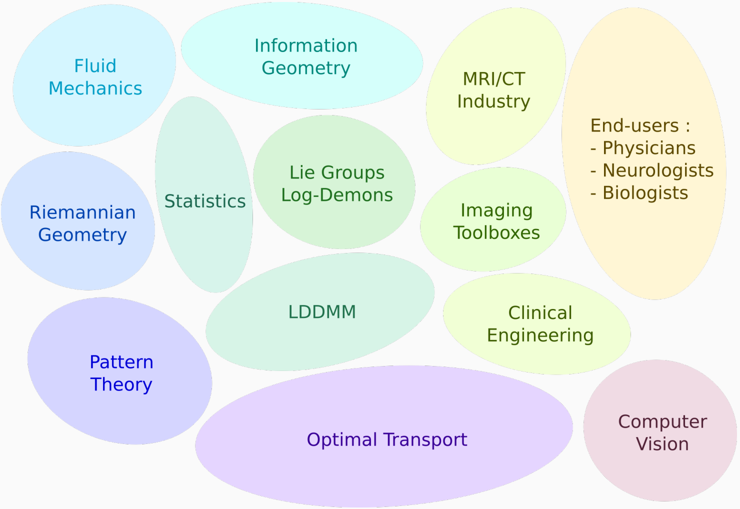


Figure 29: A (very) schematic view of the fields related to Computational Anatomy.

A continuum of professions

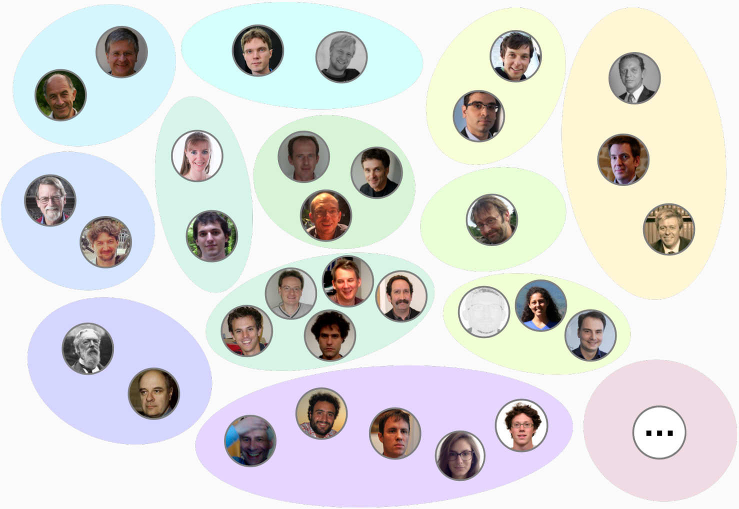


Figure 29: The people behind the labels.

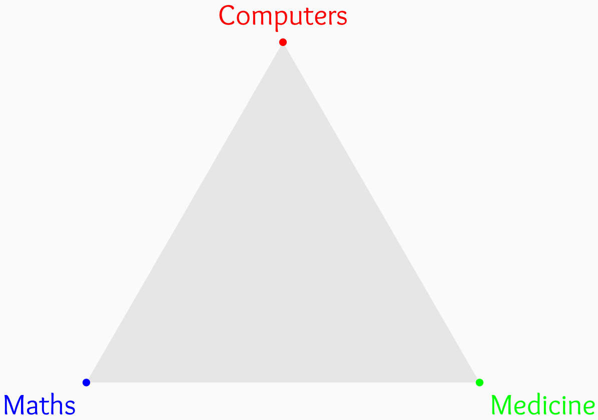


Figure 30: The space of anatomical models.

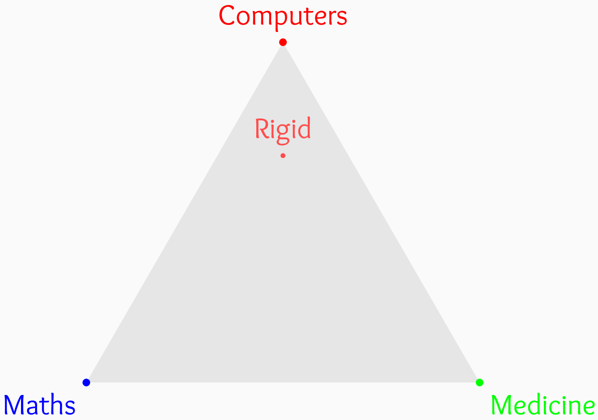


Figure 30: The space of anatomical models.

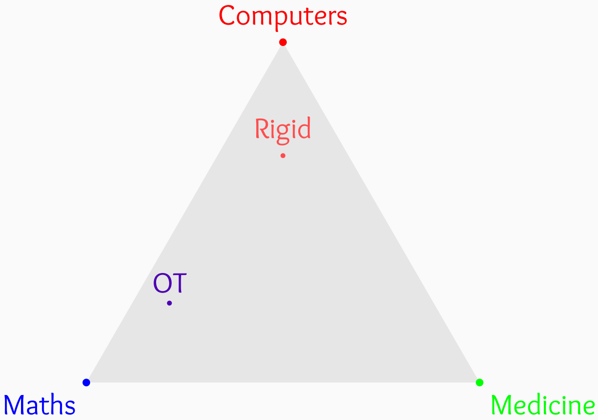


Figure 30: The space of anatomical models.

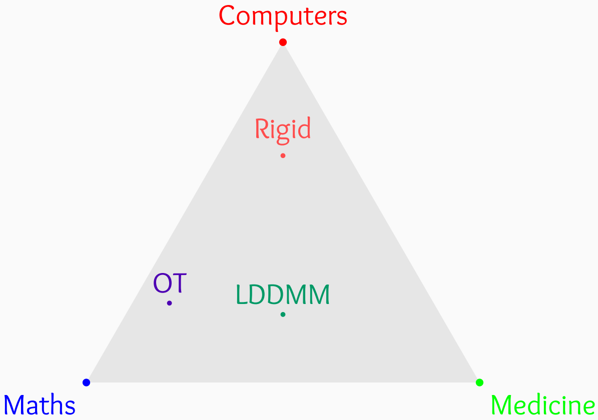


Figure 30: The space of anatomical models.

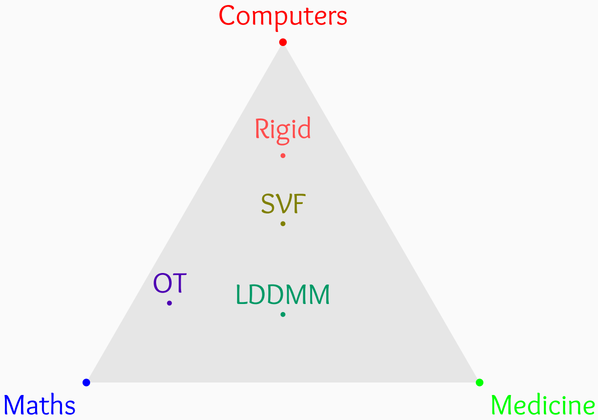


Figure 30: The space of anatomical models.

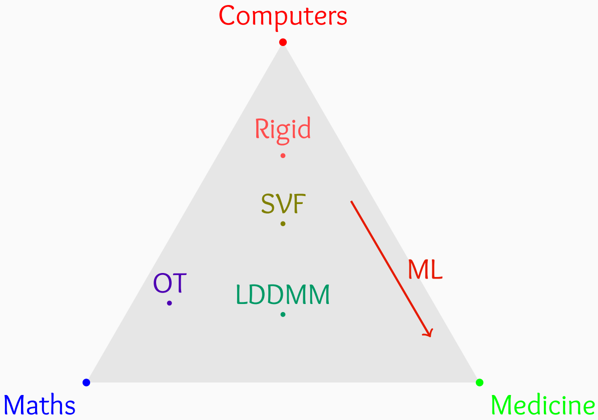


Figure 30: The space of anatomical models.

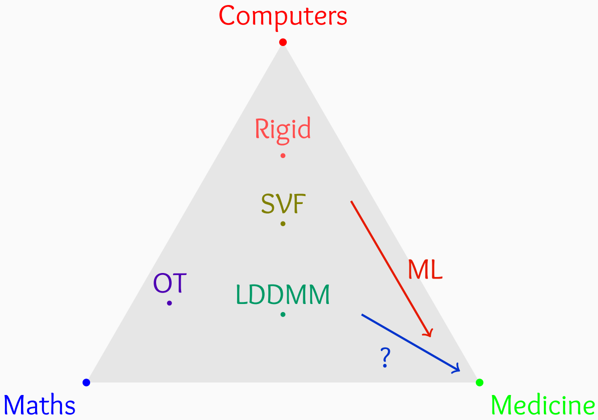


Figure 30: The space of anatomical models.

Thank you for your attention.

The `pytorch` library: symbolic maths on the GPU

```python
def _Hqp(q, p, sigma) :
    "The hamiltonian, or kinetic energy of the shape q with momentum p."
    pKqp =  _k(q, q, sigma) * (p @ p.t()) # pKqp_{i,j} = k(q_i, q_j) <p_i, p_j>_2
    return .5 * pKqp.sum()                # H(q, p) = 1/2 \sum_{i,j} k(q_i, q_j) p_i.p_j

# The partial derivatives of the Hamiltonian are automatically computed !
def _dq_Hqp(q,p,sigma) :
    return torch.autograd.grad(_Hqp(q,p,sigma), q, create_graph=True)[0]
def _dp_Hqp(q,p,sigma) :
    return torch.autograd.grad(_Hqp(q,p,sigma), p, create_graph=True)[0]

def _HamiltonianShooting(q, p, sigma) :
    "Shoots to time 1 a k-geodesic starting (at time 0) from q with momentum p."
    for t in range(10) :                       # Let's hardcode the "dt = .1".
        q,p = [q + .1 * _dp_Hqp(q,p,sigma) , # Euler steps for the Hamiltonian flow
               p - .1 * _dq_Hqp(q,p,sigma) ] # in the cotangent bundle.
    return [q,p]                               # Return the final state + momentum.
```

Shooting routines emulate the Exponential map

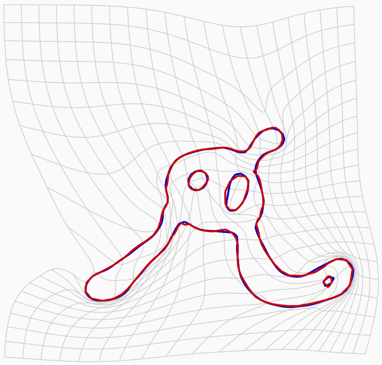


Figure: Matching a curve to another.

A deformation $\varphi_t(X)$ is encoded as a shooting momentum $p_0 \in T^\star_X \mathcal{M}$.

Find the momentum $X \xrightarrow{\varphi \simeq p_0} Y$ through gradient descent.

Shooting routines emulate the Exponential map

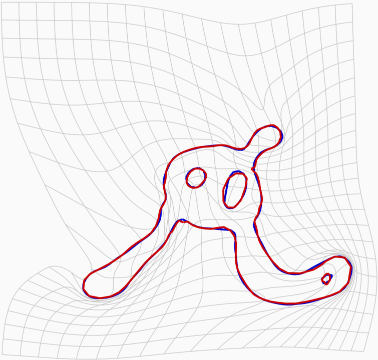
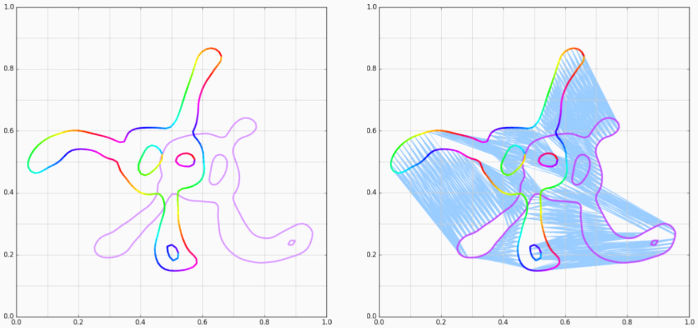


Figure: Matching a curve to another.

A deformation $\varphi_t(X)$ is encoded as a shooting momentum $p_0 \in T^\star_X \mathcal{M}$.

Find the momentum $X \xrightarrow{\varphi \simeq p_0} Y$ through gradient descent.

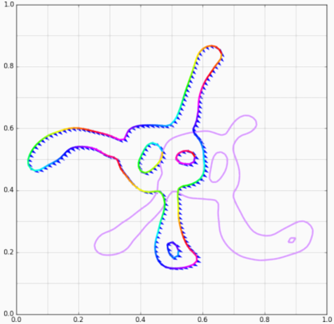
Typical run with OT fidelity



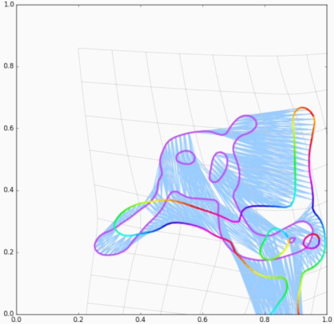
(a) Momentum p_0. (b) Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 0.

Typical run with OT fidelity



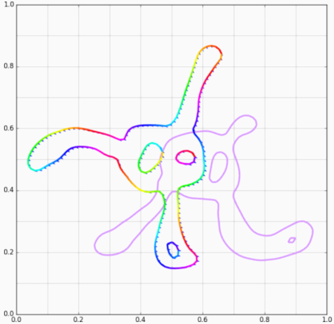
(a) Momentum p_0.



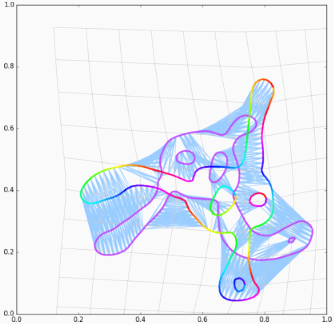
(b) Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 3.

Typical run with OT fidelity



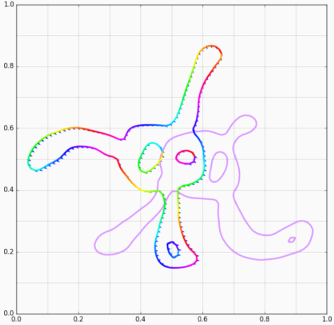
(a) Momentum p_0.



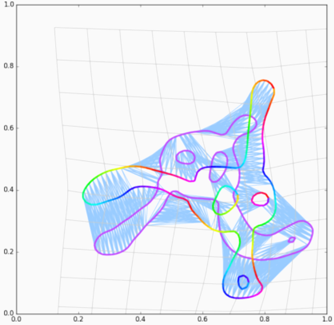
(b) Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 4.

Typical run with OT fidelity



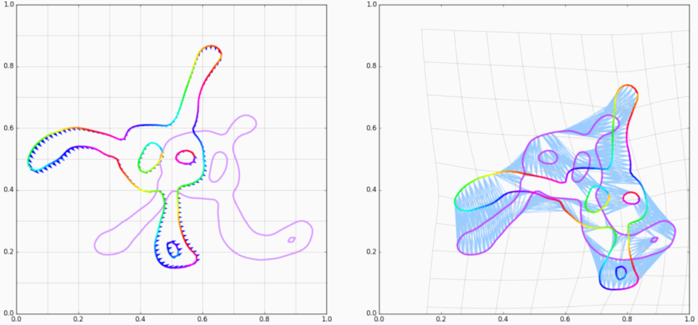
(a) Momentum p_0.



(b) Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 5.

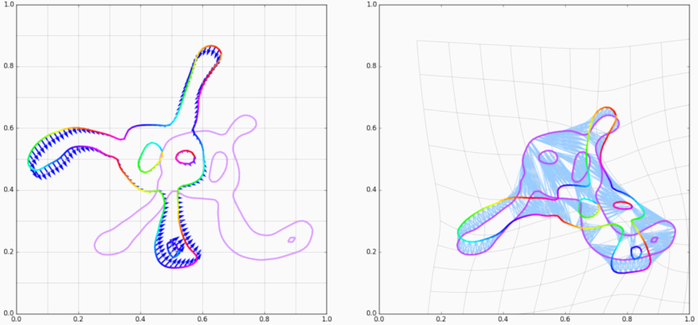
Typical run with OT fidelity



(a) Momentum p_0. **(b)** Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 6.

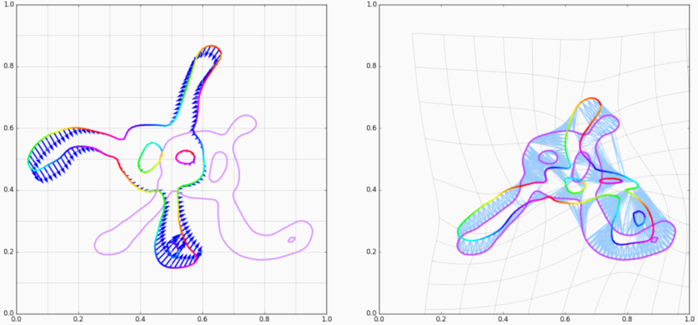
Typical run with OT fidelity



(a) Momentum p_0. **(b)** Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 7.

Typical run with OT fidelity

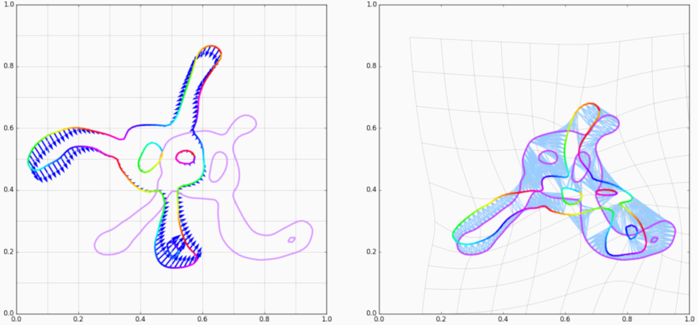


(a) Momentum p_0.

(b) Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 8.

Typical run with OT fidelity

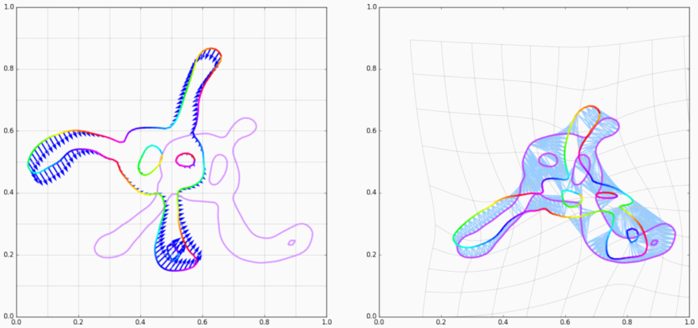


(a) Momentum p_0.

(b) Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 9.

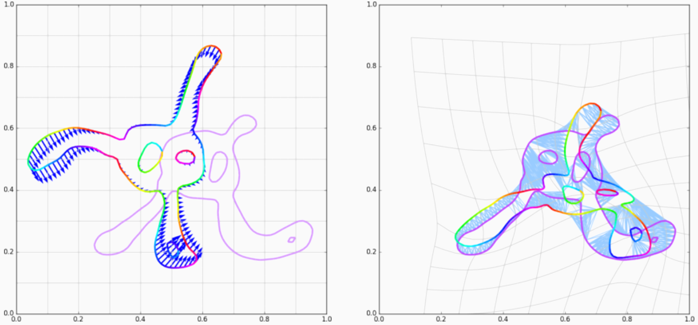
Typical run with OT fidelity



(a) Momentum p_0. **(b)** Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 10.

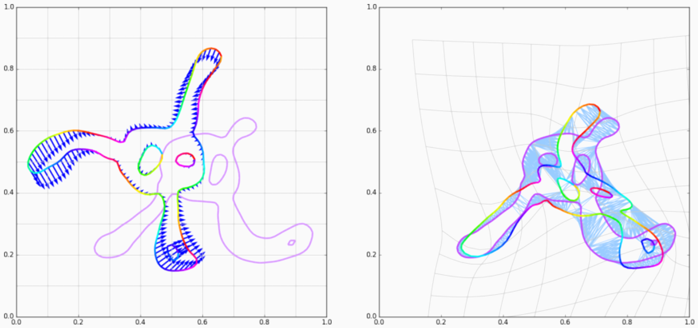
Typical run with OT fidelity



(a) Momentum p_0. **(b)** Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 11.

Typical run with OT fidelity

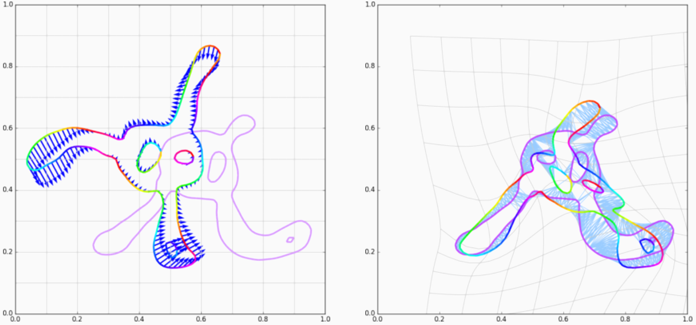


(a) Momentum p_0.

(b) Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 12.

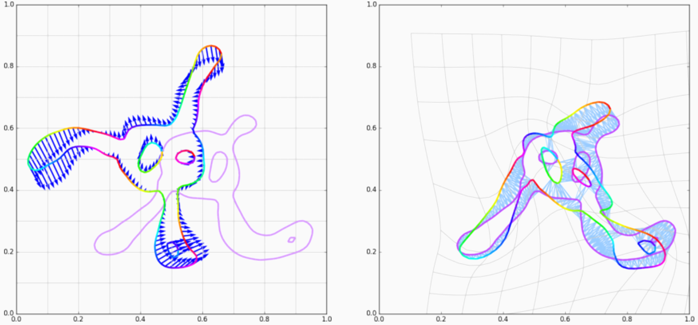
Typical run with OT fidelity



(a) Momentum p_0. **(b)** Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 13.

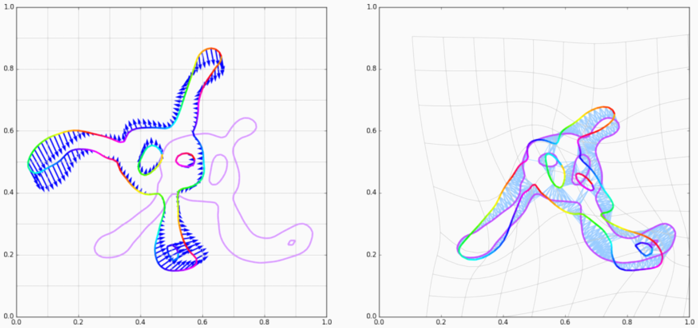
Typical run with OT fidelity



(a) Momentum p_0. **(b)** Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 14.

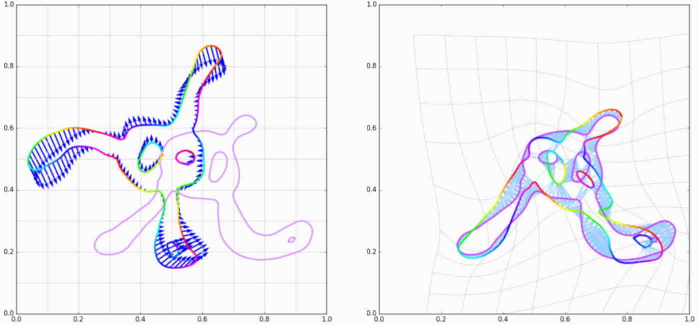
Typical run with OT fidelity



(a) Momentum p_0. **(b)** Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 15.

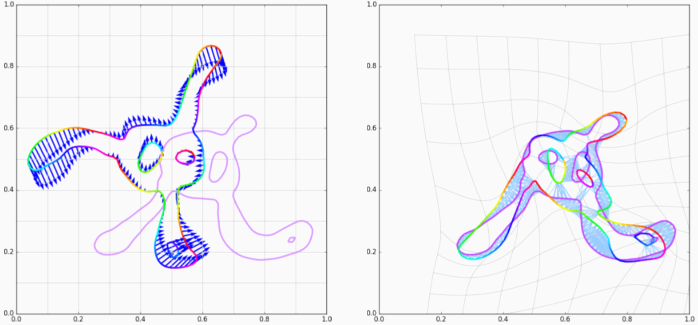
Typical run with OT fidelity



(a) Momentum p_0. **(b)** Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 16.

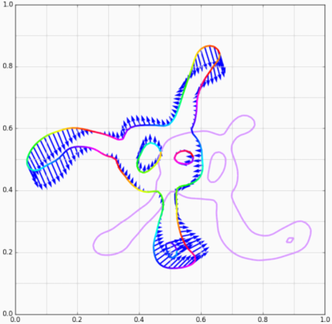
Typical run with OT fidelity



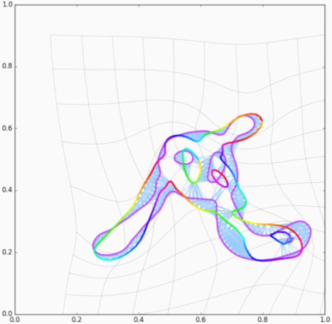
(a) Momentum p_0. **(b)** Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 17.

Typical run with OT fidelity



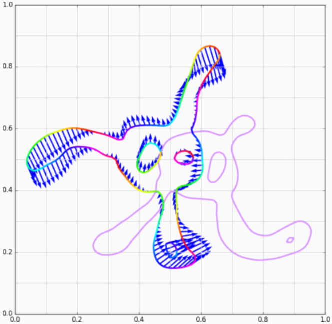
(a) Momentum p_0.



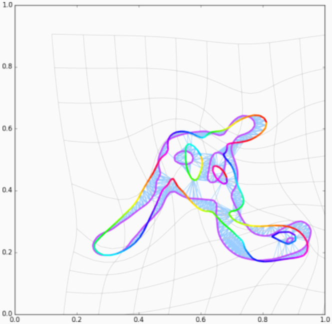
(b) Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 18.

Typical run with OT fidelity



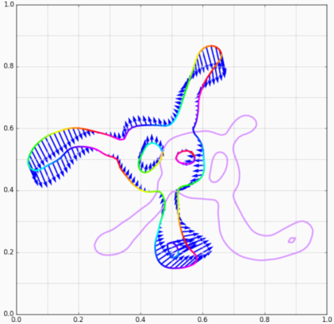
(a) Momentum p_0.



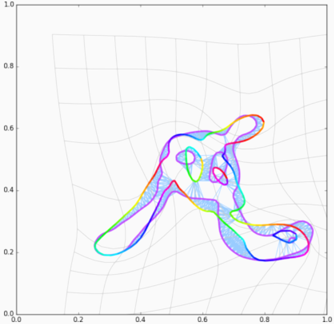
(b) Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 19.

Typical run with OT fidelity



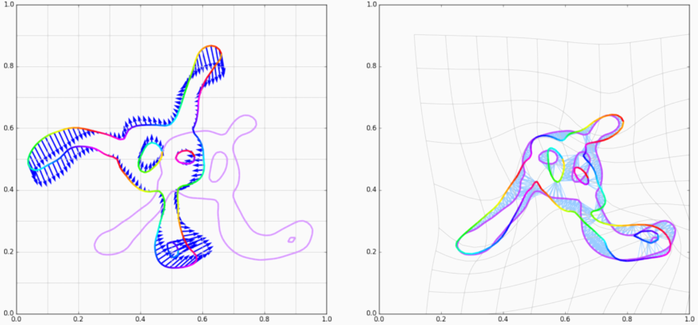
(a) Momentum p_0.



(b) Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 20.

Typical run with OT fidelity

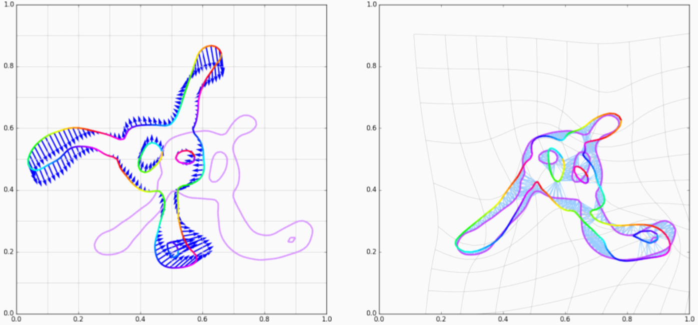


(a) Momentum p_0.

(b) Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 21.

Typical run with OT fidelity

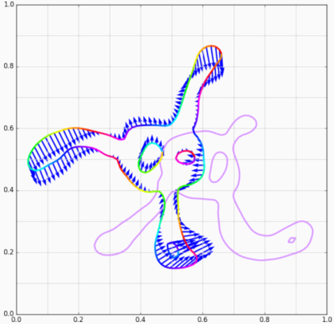


(a) Momentum p_0.

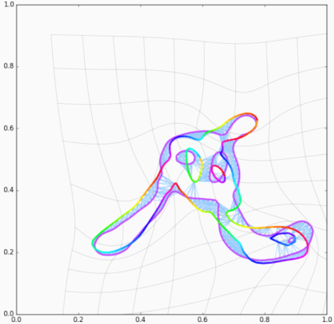
(b) Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 22.

Typical run with OT fidelity



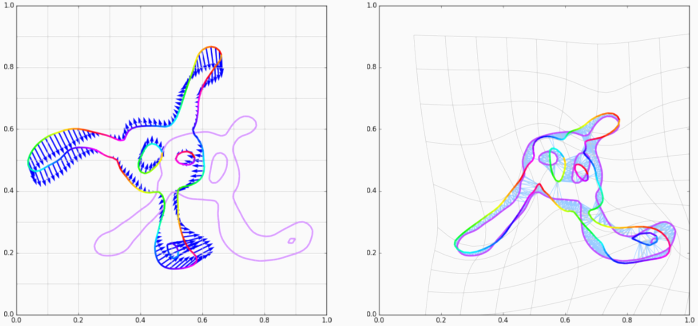
(a) Momentum p_0.



(b) Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 23.

Typical run with OT fidelity

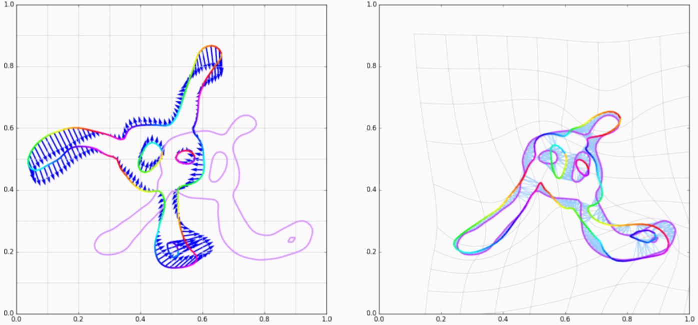


(a) Momentum p_0.

(b) Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 24.

Typical run with OT fidelity

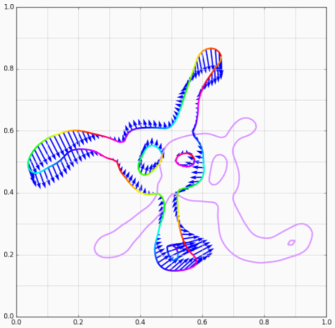


(a) Momentum p_0.

(b) Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 25.

Typical run with OT fidelity

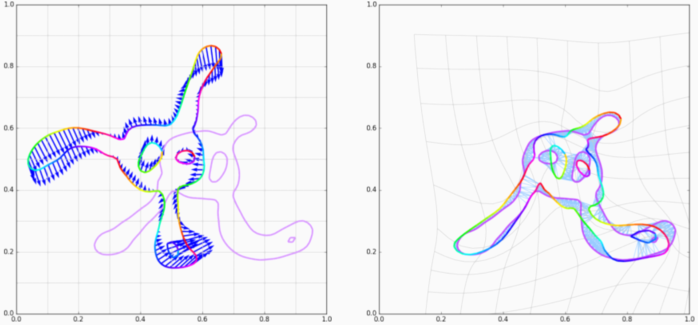


(a) Momentum p_0.

(b) Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 26.

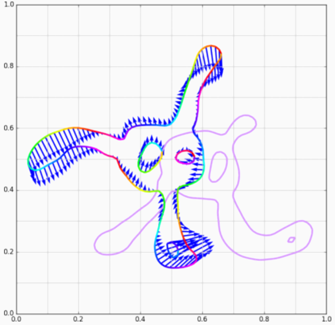
Typical run with OT fidelity



(a) Momentum p_0. **(b)** Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 27.

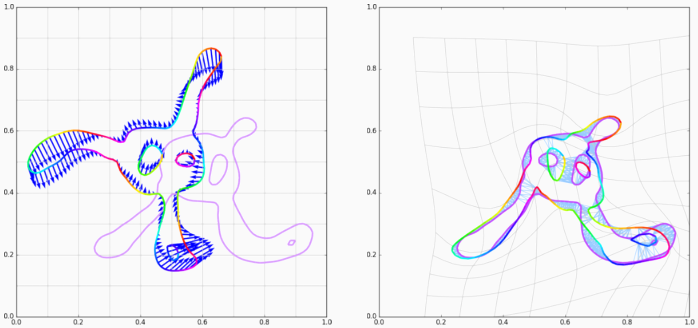
Typical run with OT fidelity



(a) Momentum p_0.   **(b)** Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 28.

Typical run with OT fidelity

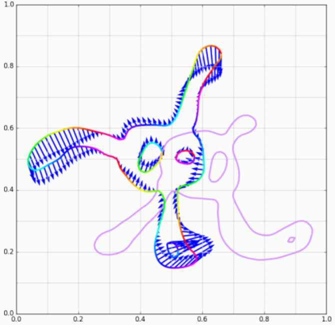


(a) Momentum p_0.

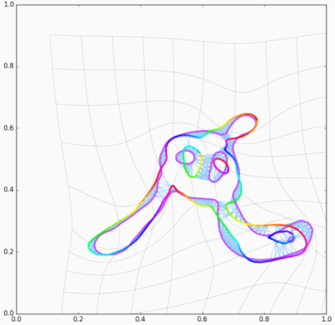
(b) Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 29.

Typical run with OT fidelity



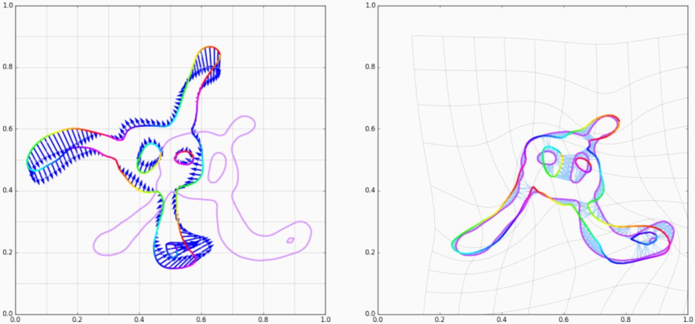
(a) Momentum p_0.



(b) Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 30.

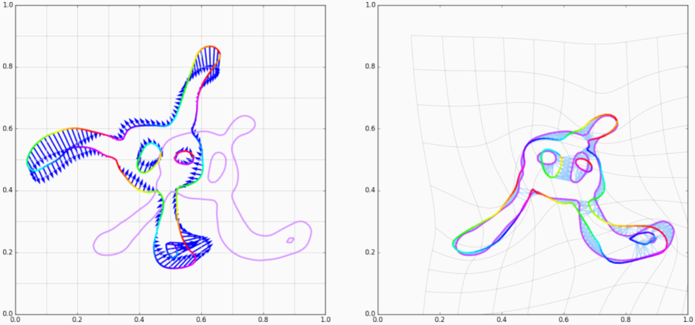
Typical run with OT fidelity



(a) Momentum p_0. **(b)** Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 31.

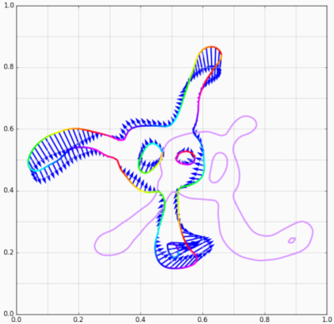
Typical run with OT fidelity



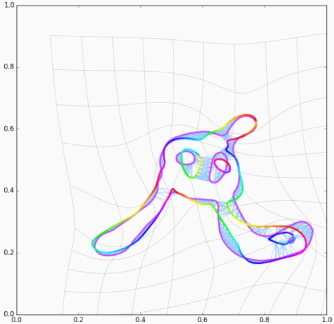
(a) Momentum p_0. **(b)** Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 32.

Typical run with OT fidelity



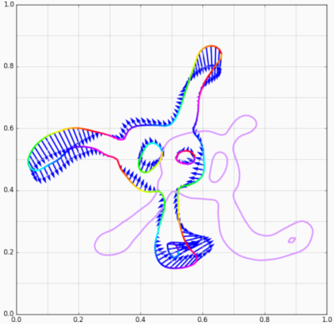
(a) Momentum p_0.



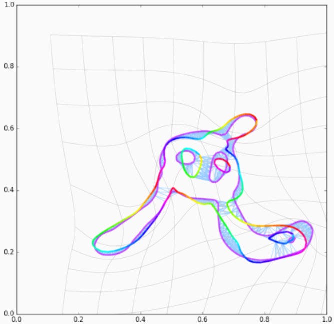
(b) Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 33.

Typical run with OT fidelity



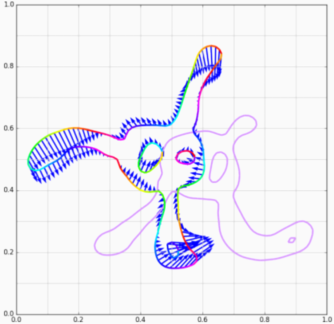
(a) Momentum p_0.



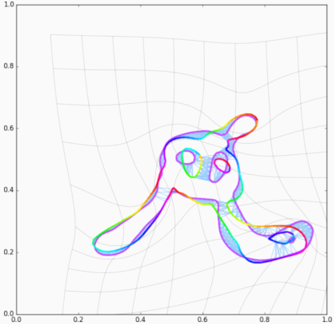
(b) Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 34.

Typical run with OT fidelity



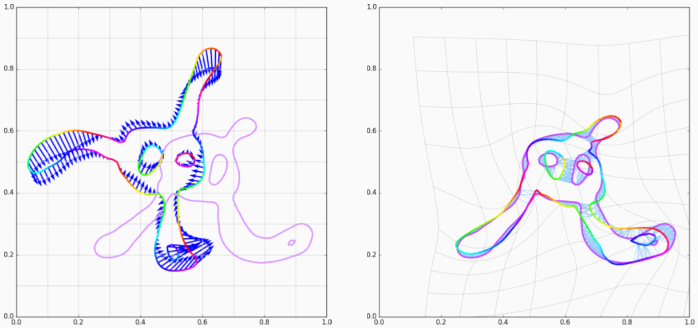
(a) Momentum p_0.



(b) Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 35.

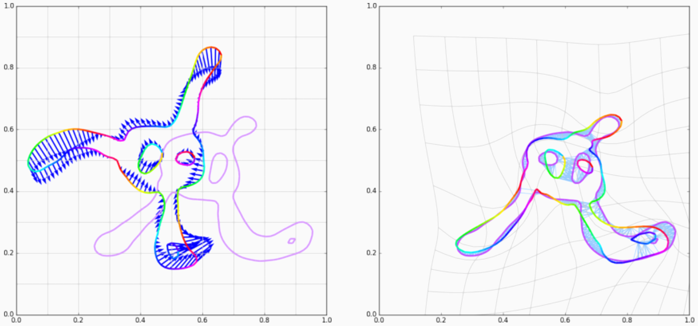
Typical run with OT fidelity



(a) Momentum p_0. **(b)** Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 36.

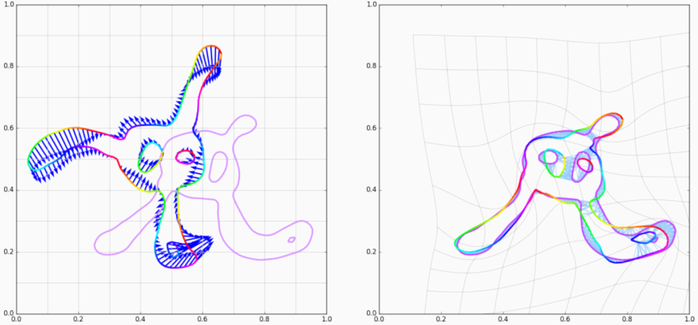
Typical run with OT fidelity



(a) Momentum p_0.   **(b)** Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 37.

Typical run with OT fidelity

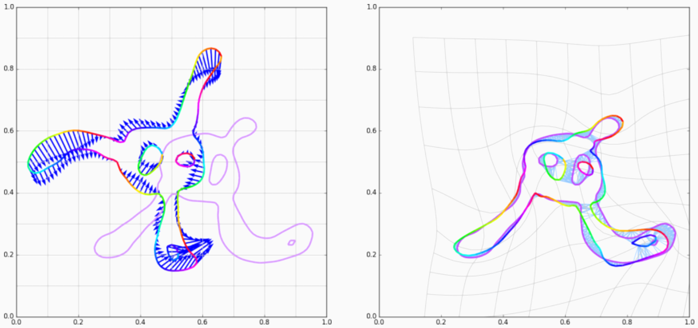


(a) Momentum p_0.

(b) Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 38.

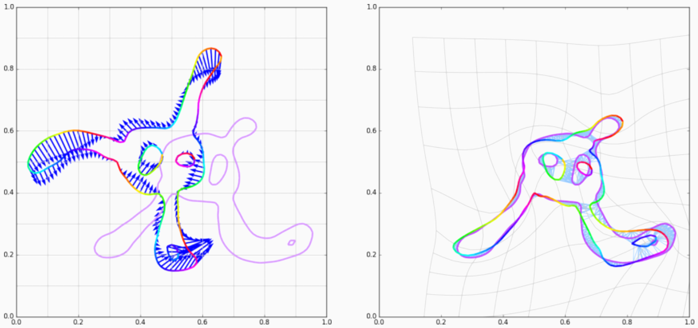
Typical run with OT fidelity



(a) Momentum p_0. **(b)** Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 39.

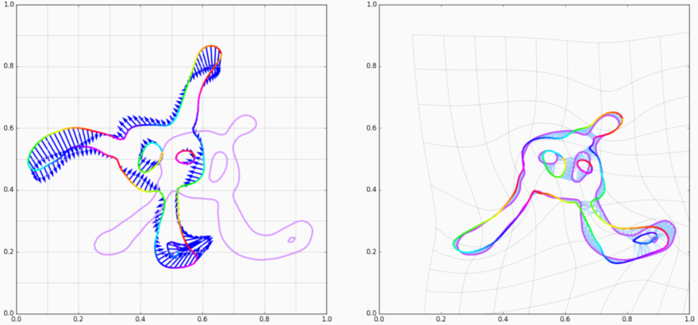
Typical run with OT fidelity



(a) Momentum p_0. **(b)** Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 41.

Typical run with OT fidelity

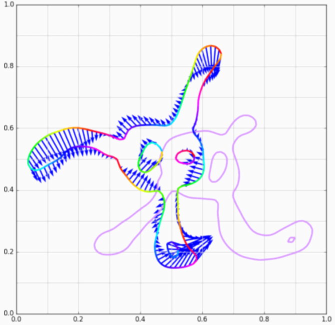


(a) Momentum p_0.

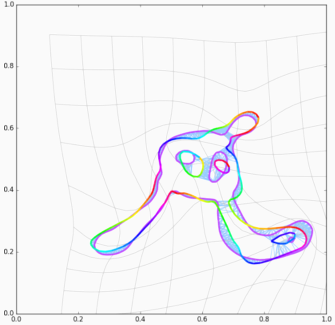
(b) Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 42.

Typical run with OT fidelity



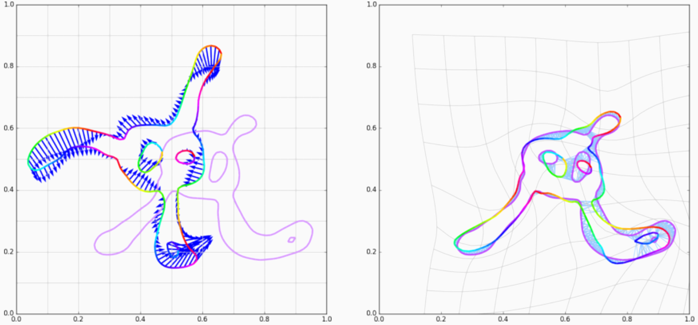
(a) Momentum p_0.



(b) Model $q_1 = \text{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 43.

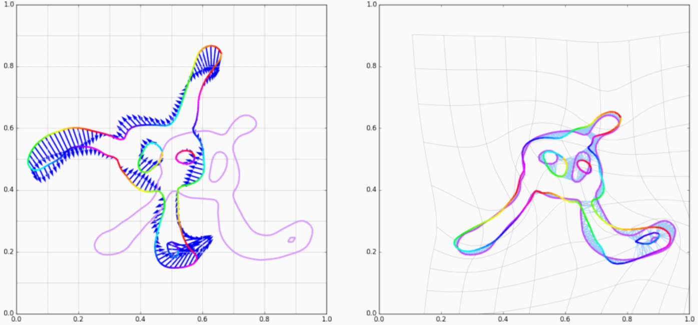
Typical run with OT fidelity



(a) Momentum p_0. **(b)** Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 44.

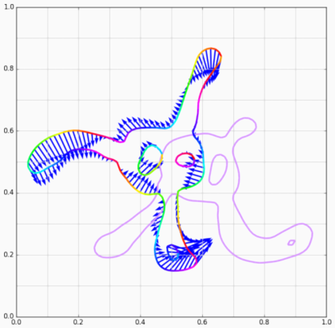
Typical run with OT fidelity



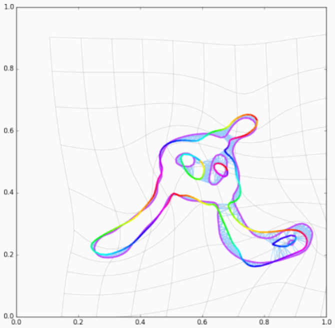
(a) Momentum p_0. **(b)** Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 46.

Typical run with OT fidelity



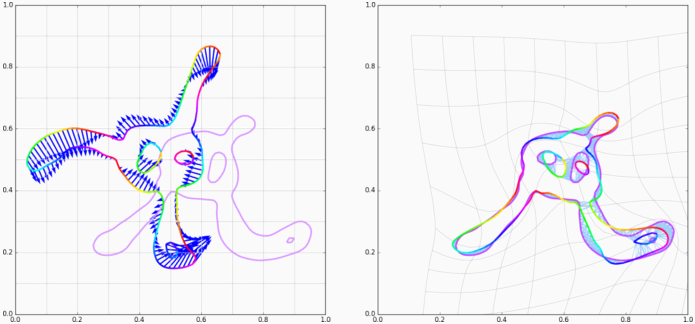
(a) Momentum p_0.



(b) Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 47.

Typical run with OT fidelity

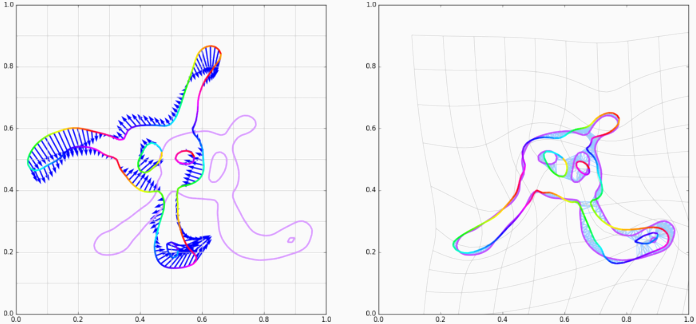


(a) Momentum p_0.

(b) Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 48.

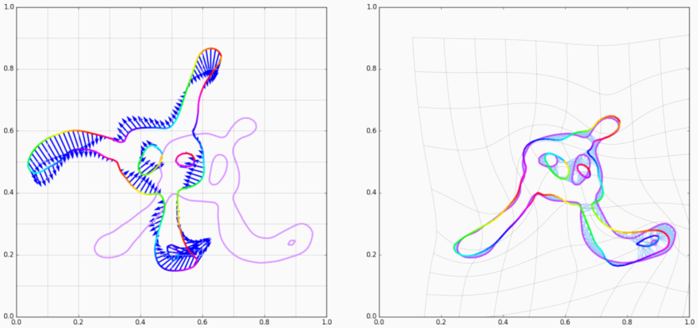
Typical run with OT fidelity



(a) Momentum p_0. **(b)** Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 49.

Typical run with OT fidelity

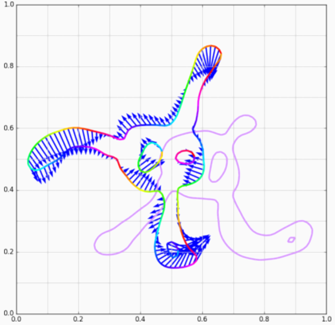


(a) Momentum p_0.

(b) Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 50.

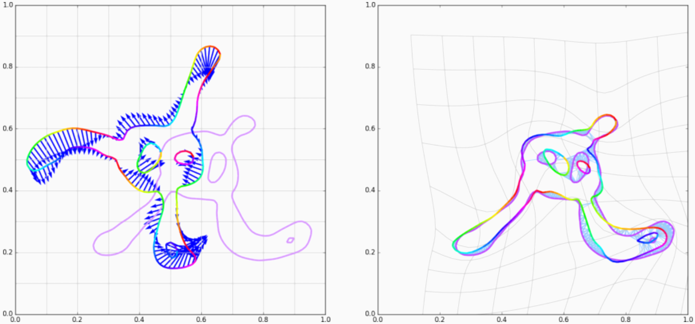
Typical run with OT fidelity



(a) Momentum p_0. **(b)** Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 52.

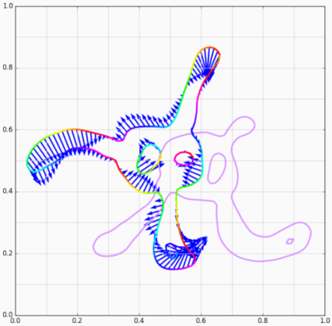
Typical run with OT fidelity



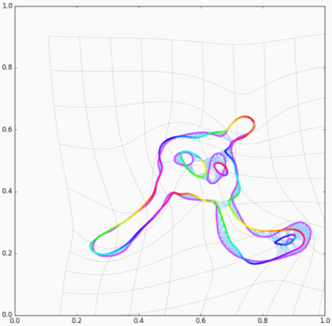
(a) Momentum p_0. **(b)** Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 53.

Typical run with OT fidelity



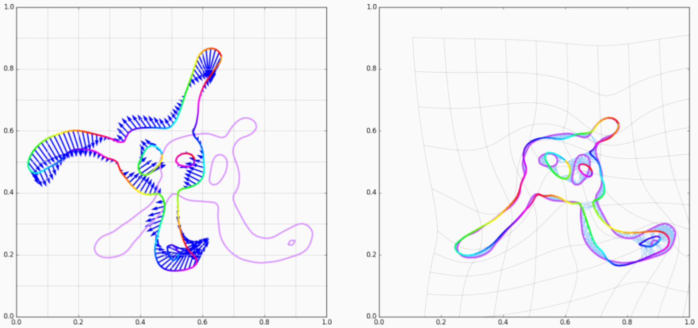
(a) Momentum p_0.



(b) Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 54.

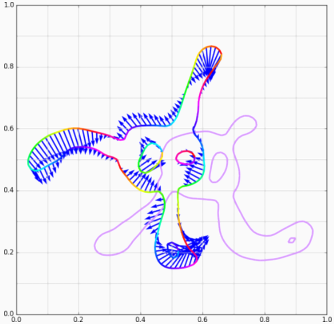
Typical run with OT fidelity



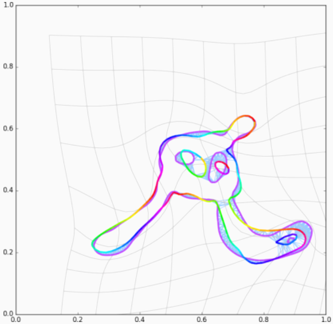
(a) Momentum p_0. **(b)** Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 55.

Typical run with OT fidelity



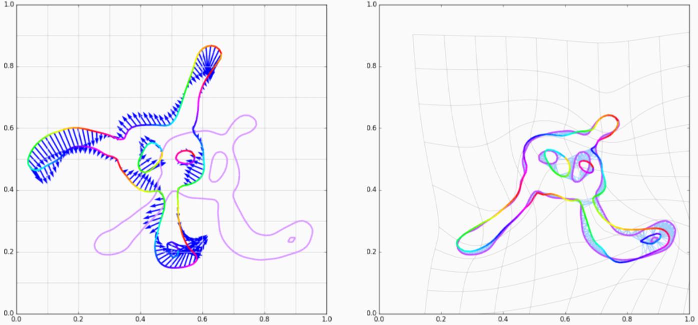
(a) Momentum p_0.



(b) Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 56.

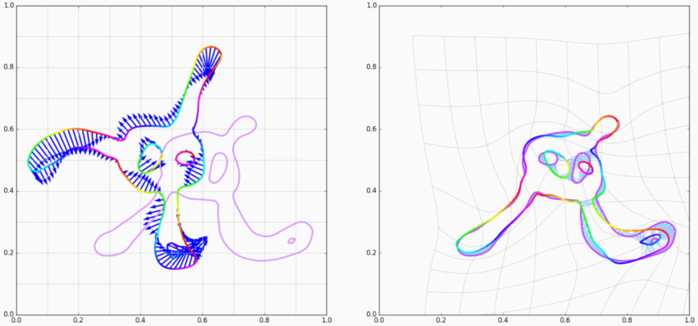
Typical run with OT fidelity



(a) Momentum p_0. **(b)** Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 57.

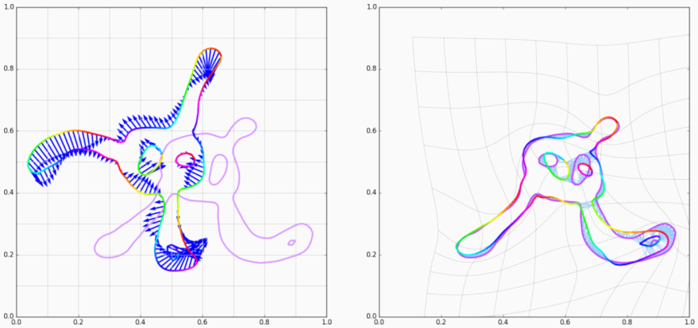
Typical run with OT fidelity



(a) Momentum p_0. **(b)** Model $q_1 = \operatorname{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 58.

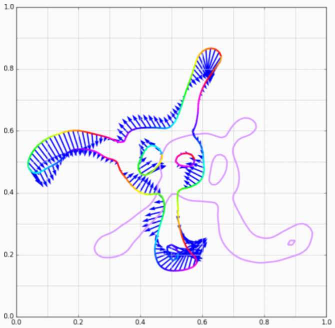
Typical run with OT fidelity



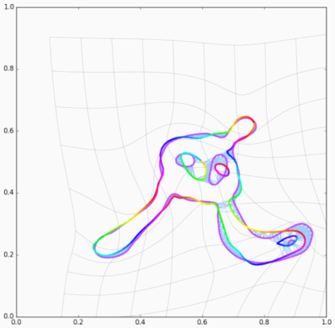
(a) Momentum p_0. **(b)** Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 59.

Typical run with OT fidelity



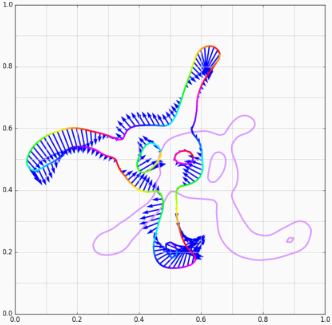
(a) Momentum p_0.



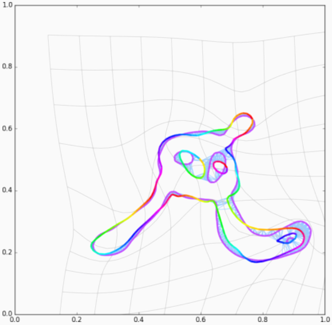
(b) Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 60.

Typical run with OT fidelity



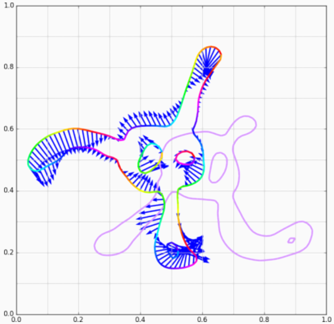
(a) Momentum p_0.



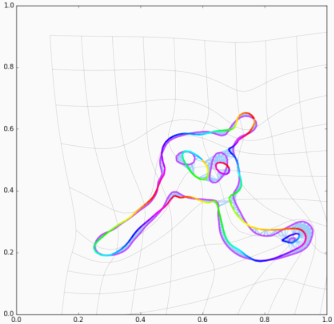
(b) Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 61.

Typical run with OT fidelity



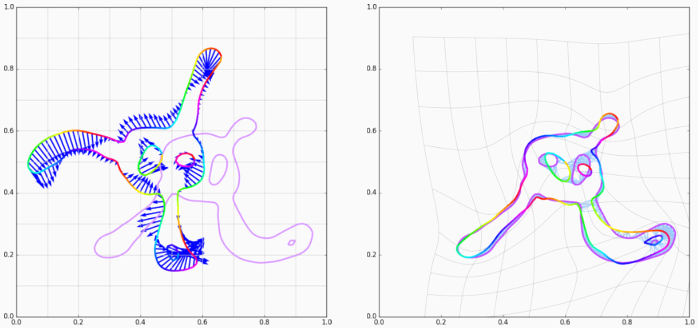
(a) Momentum p_0.



(b) Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 62.

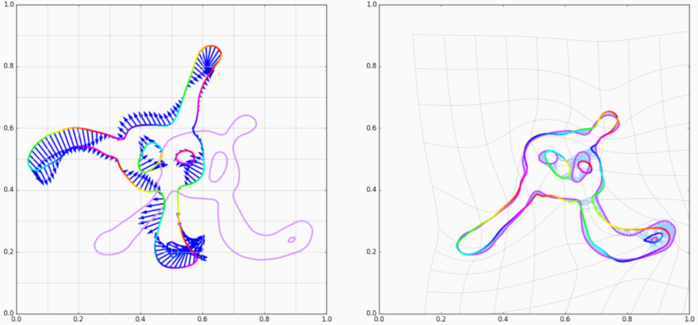
Typical run with OT fidelity



(a) Momentum p_0. **(b)** Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 64.

Typical run with OT fidelity

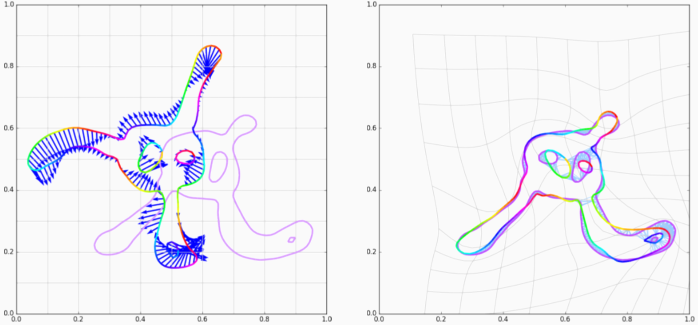


(a) Momentum p_0.

(b) Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 65.

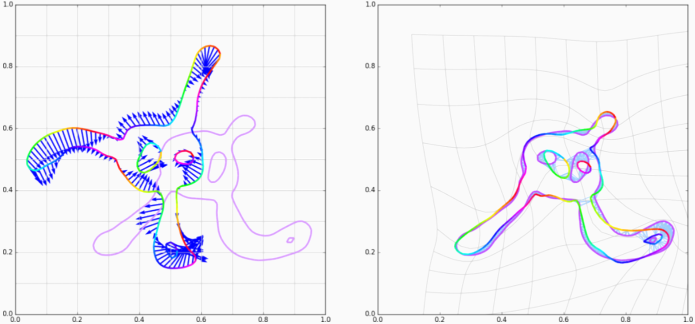
Typical run with OT fidelity



(a) Momentum p_0. **(b)** Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 66.

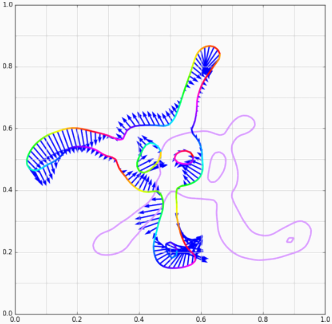
Typical run with OT fidelity



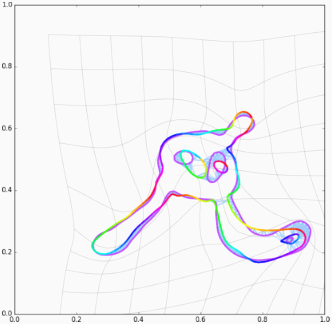
(a) Momentum p_0. **(b)** Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 67.

Typical run with OT fidelity



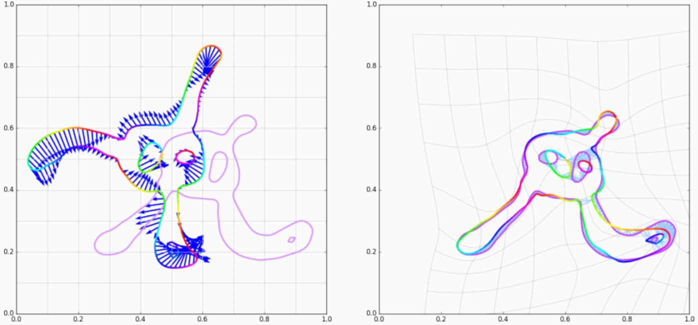
(a) Momentum p_0.



(b) Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 68.

Typical run with OT fidelity

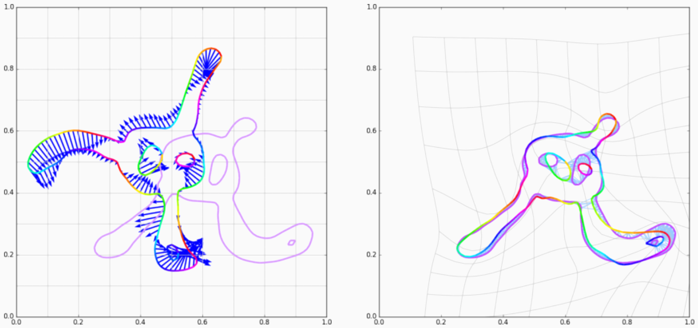


(a) Momentum p_0.

(b) Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 69.

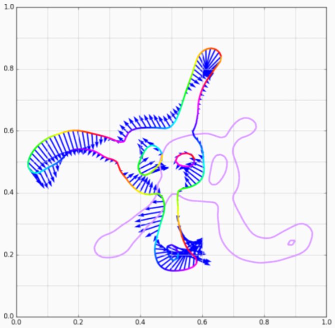
Typical run with OT fidelity



(a) Momentum p_0. **(b)** Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 70.

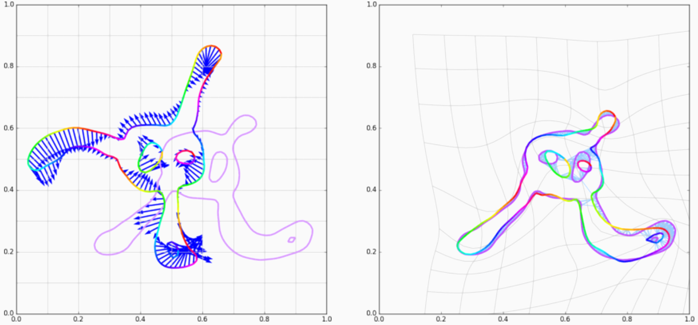
Typical run with OT fidelity



(a) Momentum p_0. **(b)** Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 71.

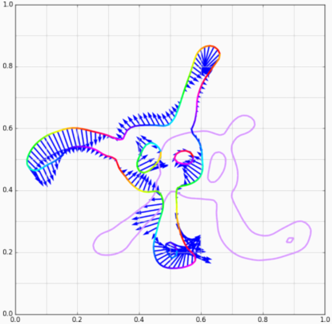
Typical run with OT fidelity



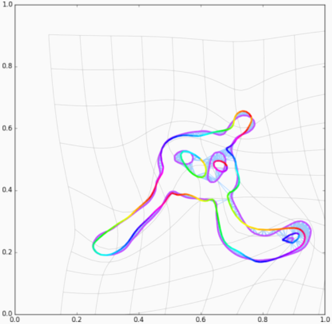
(a) Momentum p_0. **(b)** Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 72.

Typical run with OT fidelity



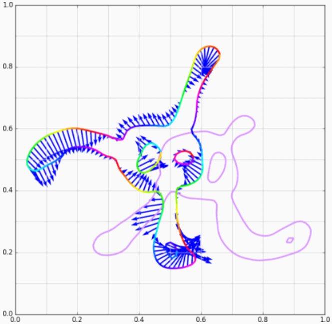
(a) Momentum p_0.



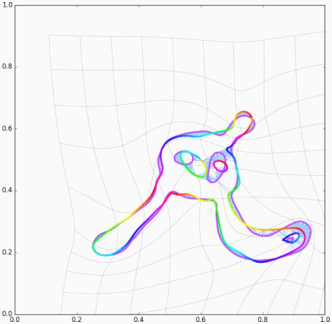
(b) Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 73.

Typical run with OT fidelity



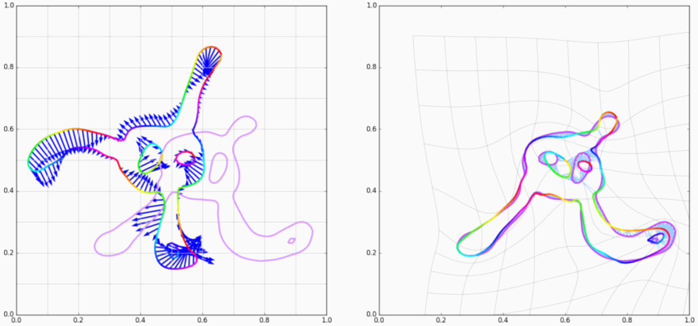
(a) Momentum p_0.



(b) Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 74.

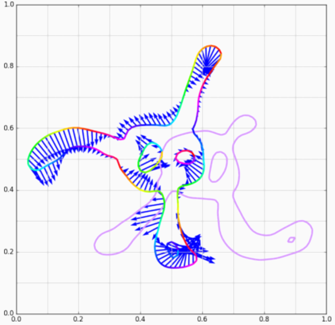
Typical run with OT fidelity



(a) Momentum p_0. **(b)** Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 75.

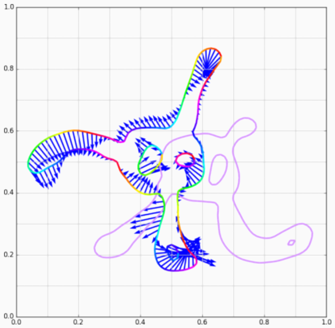
Typical run with OT fidelity



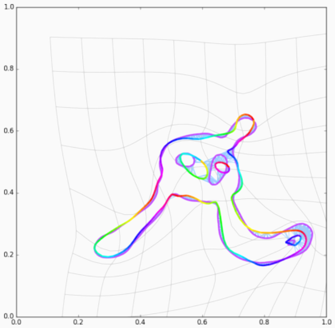
(a) Momentum p_0. **(b)** Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 77.

Typical run with OT fidelity



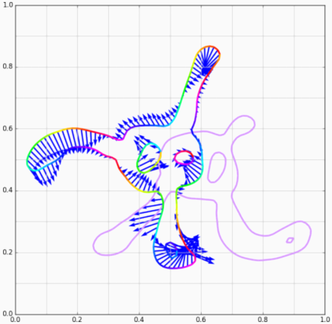
(a) Momentum p_0.



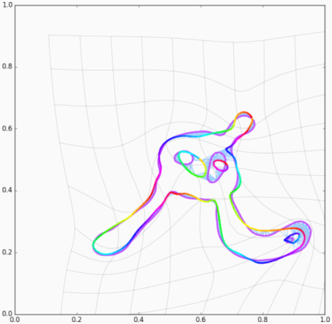
(b) Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 78.

Typical run with OT fidelity



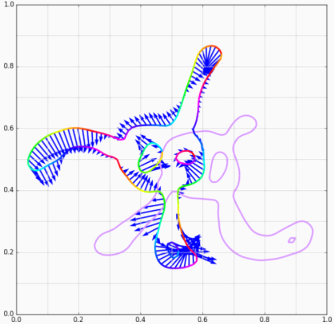
(a) Momentum p_0.



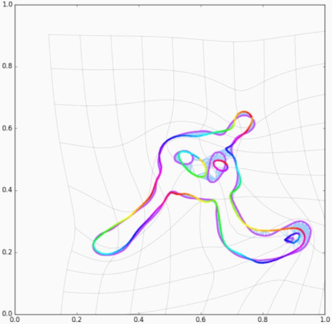
(b) Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 79.

Typical run with OT fidelity



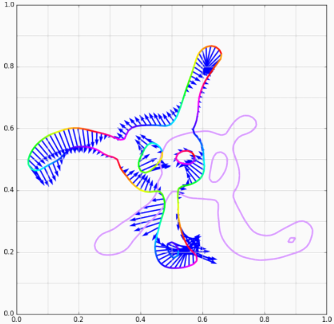
(a) Momentum p_0.



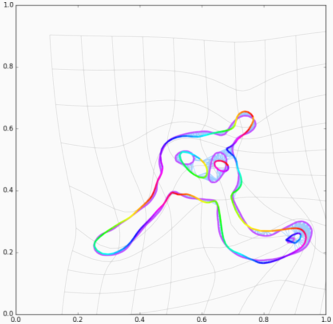
(b) Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 80.

Typical run with OT fidelity



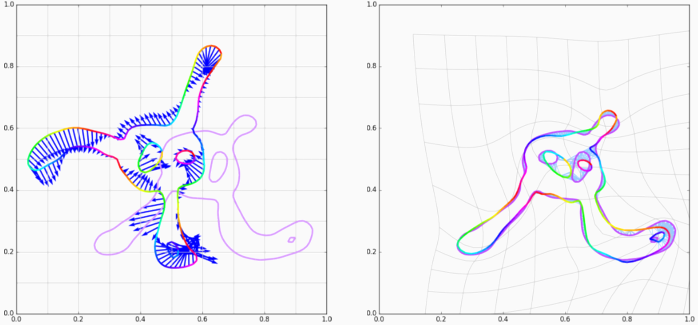
(a) Momentum p_0.



(b) Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 81.

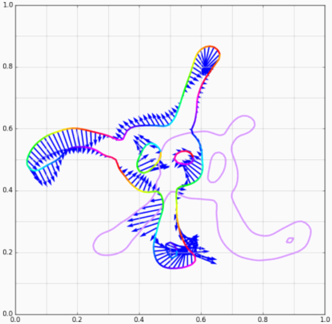
Typical run with OT fidelity



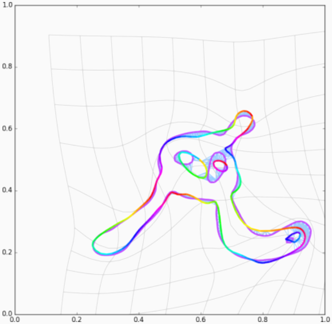
(a) Momentum p_0. **(b)** Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 82.

Typical run with OT fidelity



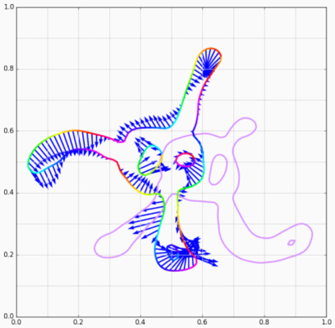
(a) Momentum p_0.



(b) Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 83.

Typical run with OT fidelity

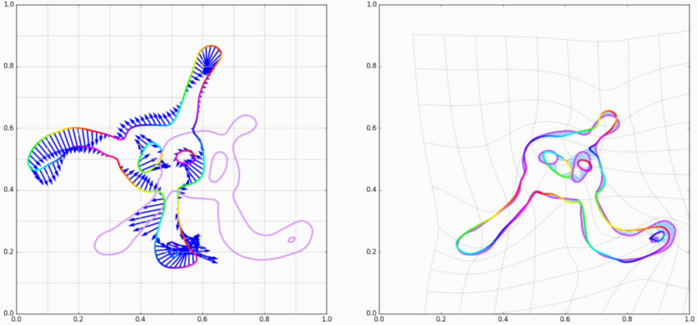


(a) Momentum p_0.

(b) Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 85.

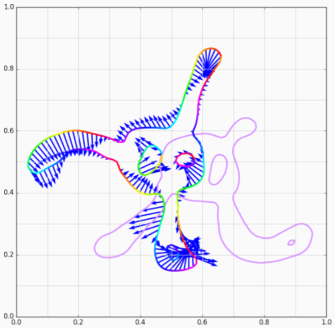
Typical run with OT fidelity



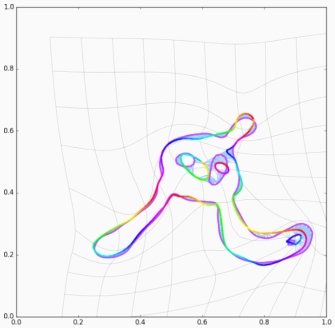
(a) Momentum p_0. **(b)** Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 86.

Typical run with OT fidelity



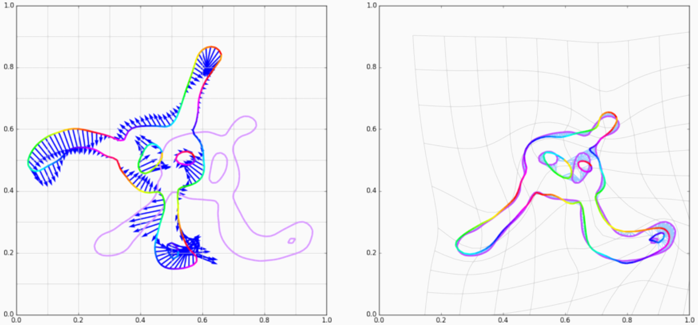
(a) Momentum p_0.



(b) Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 87.

Typical run with OT fidelity

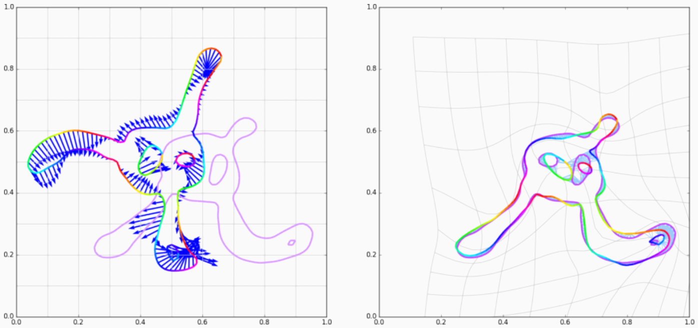


(a) Momentum p_0.

(b) Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 88.

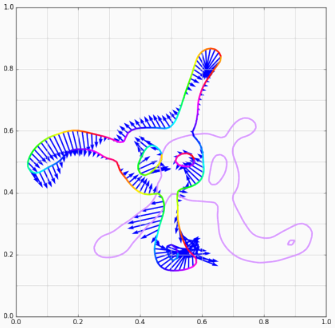
Typical run with OT fidelity



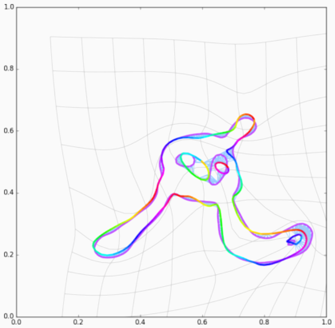
(a) Momentum p_0. **(b)** Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 89.

Typical run with OT fidelity



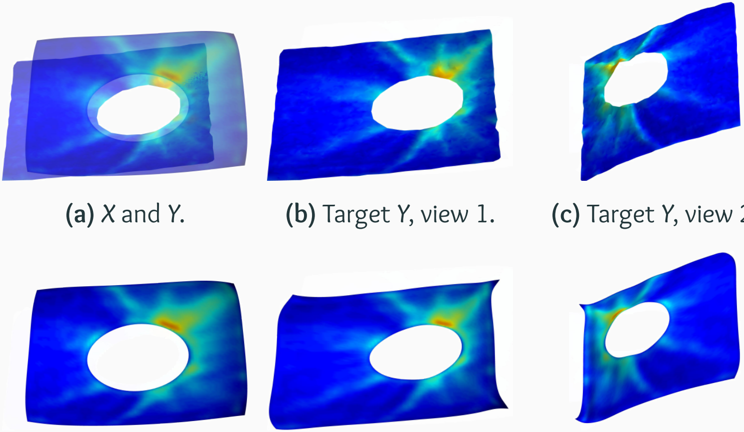
(a) Momentum p_0.



(b) Model $q_1 = \mathrm{Exp}_{q_0}(p_0)$.

Figure 31: Iteration 90.

Matchings of partially observed shapes



(a) X and Y. (b) Target Y, view 1. (c) Target Y, view 2.

(d) Source X. (e) $f(X)$, view 1. (f) $f(X)$, view 2.

Figure 32: Matching artifacts for the *retina* dataset.

References I

P. Addis, P. Melis, R. Cannas, M. S. F. Tinti, C. Piccinetti, and A. Cau.
A morphometric approach for the analysis of body shape in bluefin tuna: preliminary results.
Collect. Vol. Sci. Pap. ICCAT, 65(3):982–987, 2010.

S. P. Awate and R. T. Whitaker.
Feature-preserving mri denoising: a nonparametric empirical bayes approach.
IEEE Transactions on Medical Imaging, 26(9):1242–1255, 2007.

References II

S. Clausen, K. Greiner, O. Andersen, K.-A. Lie, H. Schulerud, and T. Kavli.
Automatic segmentation of overlapping fish using shape priors.
In *Scandinavian conference on Image analysis*, pages 11–20. Springer, 2007.

S. Gerber, T. Tasdizen, P. T. Fletcher, S. Joshi, R. Whitaker, A. D. N. Initiative, et al.
Manifold modeling for brain population analysis.
Medical image analysis, 14(5):643–653, 2010.

References III

S. Lee, N. Charon, B. Charlier, K. Popuri, E. Lebed, M. V. Sarunic, A. Trouvé, and M. F. Beg.
Atlas-based shape analysis and classification of retinal optical coherence tomography images using the functional shape (fshape) framework.
Medical image analysis, 35:570–581, 2017.

Y. Nikulin and R. Novak.
Exploring the neural algorithm of artistic style.
arXiv preprint arXiv:1602.07188, 2016.

References IV

S. M. Smith, M. Jenkinson, M. W. Woolrich, C. F. Beckmann, T. E. Behrens, H. Johansen-Berg, P. R. Bannister, M. De Luca, I. Drobnjak, D. E. Flitney, et al.
Advances in functional and structural mr image analysis and implementation as fsl.
Neuroimage, 23:S208–S219, 2004.