Optimal transport with Laplacian regularization

Applications to domain adaptation and shape matching

R. Flamary, N. Courty, A. Rakotomamonjy, D. Tuia

NIPS 2014 Workshop on Optimal Transport & Machine Learning

December 13, 2014

Objective

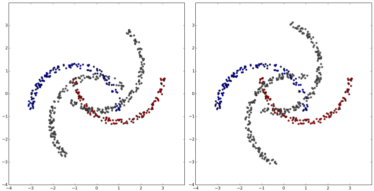
Approach

- Investigate the use of optimal transport to transport the samples from one distribution to another.
- Promote graph regularization on the transported samples.

Applications:

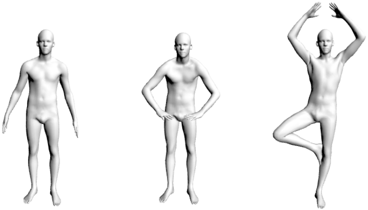
Domain adaptation

Transport samples to the new domain then train classifier.



Shape matching

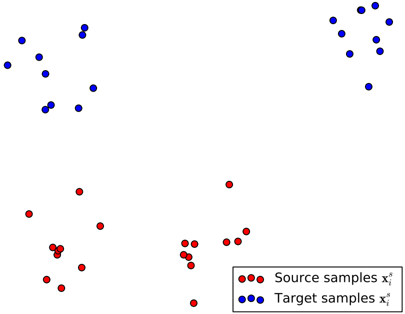
Align meshes in computer graphics using OT.



Optimal transport for discrete distribution

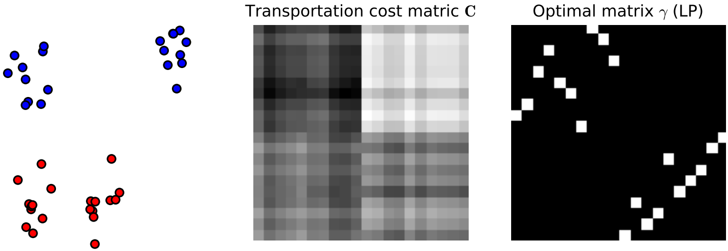
Dataset and discrete distributions

$$\mu_s = \sum_{i=1}^{n_s} p_i^s \delta_{\mathbf{x}_i^s}, \quad \mu_t = \sum_{i=1}^{n_t} p_i^t \delta_{\mathbf{x}_i^t} \qquad (1)$$



- $\delta_{\mathbf{x}_i}$ is the Dirac at location $\mathbf{x}_i \in \mathbb{R}^d$.
- p_i^s and p_i^t are probability masses.
- $\sum_{i=1}^{n_s} p_i^s = \sum_{i=1}^{n_t} p_i^t = 1$
- In this work $p_i^s = \frac{1}{n_s}$ and $p_i^t = \frac{1}{n_t}$.
- Samples stored in matrices
  $\mathbf{X}_s = [\mathbf{x}_1^s, \dots, \mathbf{x}_{ns}^s]^\top \in \mathbb{R}^{n_s \times d}$
  $\mathbf{X}_t = [\mathbf{x}_1^t, \dots, \mathbf{x}_{nt}^t]^\top \in \mathbb{R}^{n_t \times d}$

Regularized optimal transport



Optimization problem

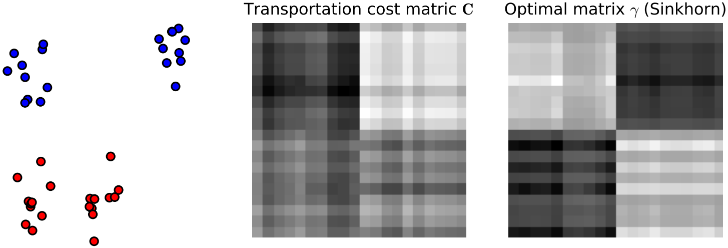
$$\boldsymbol{\gamma}_0 = \underset{\boldsymbol{\gamma} \in \mathcal{P}}{\arg\min} \quad \langle \boldsymbol{\gamma}, \mathbf{C} \rangle_F + \lambda \Omega(\boldsymbol{\gamma}) \tag{2}$$

where $\mathbf{C}$ is a transportation cost matrix, $\Omega(\cdot)$ is a regularization term and

$$\mathcal{P} = \left\{ \boldsymbol{\gamma} \in (\mathbb{R}^+)^{\mathbf{n_s} \times \mathbf{n_t}} | \, \boldsymbol{\gamma} \mathbf{1_{n_t}} = \boldsymbol{\mu_s}, \boldsymbol{\gamma}^{\mathbf{T}} \mathbf{1_{n_s}} = \boldsymbol{\mu_t} \right\}$$

For classical OT, the regularization term is $\Omega(\cdot) = 0$.

Regularized optimal transport



Optimization problem

$$\boldsymbol{\gamma}_0 = \underset{\boldsymbol{\gamma}\in\mathcal{P}}{\arg\min} \quad \langle\boldsymbol{\gamma},\mathbf{C}\rangle_F + \lambda\Omega(\boldsymbol{\gamma}) \tag{2}$$

where $\mathbf{C}$ is a transportation cost matrix, $\Omega(\cdot)$ is a regularization term and

$$\mathcal{P} = \left\{\boldsymbol{\gamma} \in (\mathbb{R}^+)^{\mathbf{n_s}\times\mathbf{n_t}} \,|\, \boldsymbol{\gamma}\mathbf{1_{n_t}} = \boldsymbol{\mu_s}, \boldsymbol{\gamma}^{\mathbf{T}}\mathbf{1_{n_s}} = \boldsymbol{\mu_t}\right\}$$

For classical OT, the regularization term is $\Omega(\cdot) = 0$.

Choice of the regularization

OT$_{sinkhorn}$: Entropy based regularization (Cuturi [1])

- Information theory based regularization:

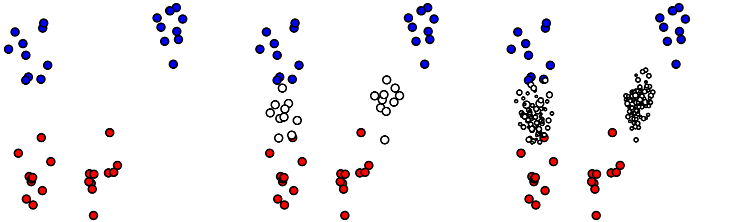
$$\Omega(\boldsymbol{\gamma}) = \sum_{i,j} \gamma_{i,j} \log(\gamma_{i,j})$$

- Shrinkage effect on the transported samples for large regularization.
- Efficient solver but sometimes numerical problems.

LOT : Laplacian regularization (Ferradans et al. [2])

- Encode graph based knowledge in the optimization problem.
- Regularization focus on the displacement of the samples during interpolation.
- Can be expressed as a Linear Program (LP) or as a Quadratic Program (QP).
- Use interior point Frank-Wolfe algorithm to solve the QP.

Transporting the discrete samples (1)



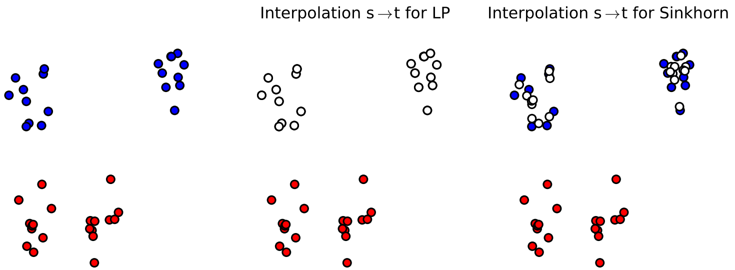
Interpolation between discrete distributions

- γ_0 defines the distribution of the mass of each source sample onto the target samples.
- Symmetric interpolation between source $(t = 0)$ and target $(t = 1)$:

$$\mu(t) = \sum_{i=1,j=1}^{n_s,n_t} \gamma_{i,j} \delta_{(1-t)\mathbf{x}_i^s + t\mathbf{x}_j^t} \tag{3}$$

The number of dirac in the intermediate $0 < t < 1$ interpolation is the number of nonzero coefficients in γ.

Transporting the discrete samples (2)



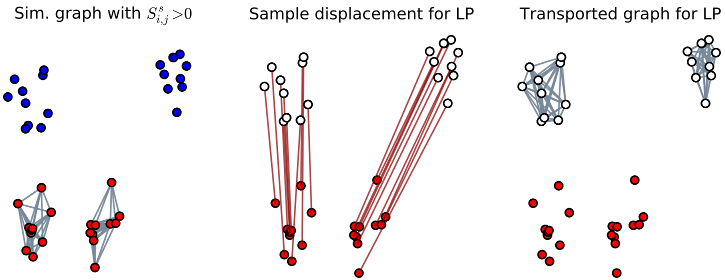
Interpolation samples between distributions

- The original mass of each source sample is spread onto the target samples as defined by $\boldsymbol{\gamma}_0$.
- Position of the transported samples:

$$\hat{\mathbf{X}}_s = \text{diag}(\boldsymbol{\gamma}_0 \mathbf{1}_{n_t})^{-1} \boldsymbol{\gamma}_0 \mathbf{X}_t \quad \text{and} \quad \hat{\mathbf{X}}_t = \text{diag}(\boldsymbol{\gamma}_0^\top \mathbf{1}_{n_s})^{-1} \boldsymbol{\gamma}_0^\top \mathbf{X}_s. \tag{4}$$

- Transported sample at the center of mass of its transported distribution.
- For uniform distributions $p_i^s = \frac{1}{n_s}$ we have $\hat{\mathbf{X}}_s = n_s \boldsymbol{\gamma}_0 \mathbf{X}_t$.

Two flavors of Laplacian regularization



Similarity matrix $\mathbf{S}^s$ defines a graph of similarity between source samples.

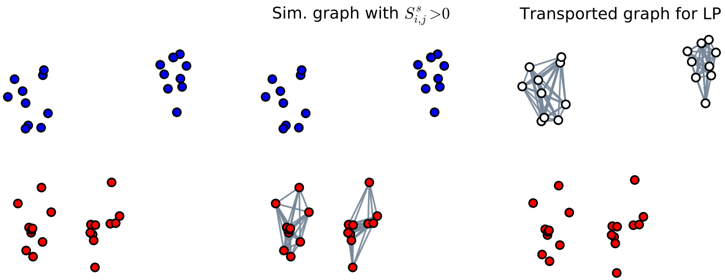
Regularizing the sample displacement [2]

- Similar samples should have similar displacement.
- Rigid displacements in clusters for large regularization.
- LOT$_{disp}$.

Regularizing the sample position

- Similar samples should be transported to similar positions.
- Shrinkage to the center of mass of clusters for large regularization.
- **Our contribution**.
- LOT$_{pos}$.

Laplacian regularization for sample position



Graph regularization for the sample position

- We want similar samples to have similar positions after transport:

$$\Omega_{pos}(\boldsymbol{\gamma}) = \frac{1}{N_s^2}\sum_{i,j} S^s_{i,j}\|\hat{\mathbf{x}}^s_i - \hat{\mathbf{x}}^s_j\|^2$$

- With uniform distributions the transported sample $\hat{\mathbf{x}}^s_i$ is linear *w.r.t.* $\boldsymbol{\gamma}$.
- Regularization term is quadratic *w.r.t.* $\boldsymbol{\gamma}$.

Laplacian regularization for sample position (2)

Reformulation of the regularization

- The regularization term can be expressed in matrix form as:

$$\Omega_{pos}(\boldsymbol{\gamma}) = \frac{1}{N_s^2} \sum_{i,j} S_{i,j}^s \|\hat{\mathbf{x}}_i^s - \hat{\mathbf{x}}_j^s\|^2 = \mathrm{Tr}(\mathbf{X}_t^\top \boldsymbol{\gamma}^\top \mathbf{L}_s \boldsymbol{\gamma} \mathbf{X}_t)$$

  where $\mathbf{L}_s = \mathrm{diag}(\mathbf{S}_s \mathbf{1}) - \mathbf{S}_s$ is the Laplacian of the graph $\mathbf{S}_s$.
- Gradient of the trace is easy to obtain during the optimization.

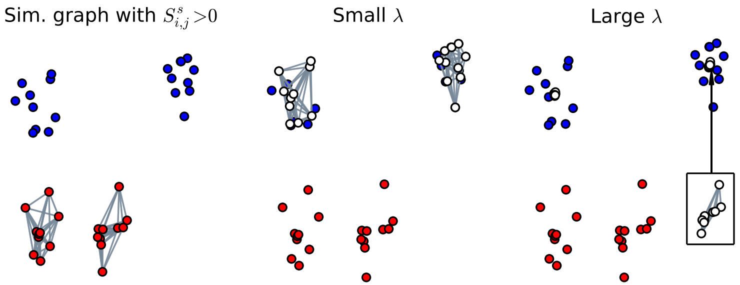
Symmetric regularization

- Regularization $\Omega_{pos}(\cdot)$ promotes only similarity in the transported source samples.
- We can also use a symmetric regularization of the form:

$$\Omega_{pos}(\boldsymbol{\gamma}) = \frac{1-\alpha}{N_s^2} \sum_{i,j} S_{i,j}^s \|\hat{\mathbf{x}}_i^s - \hat{\mathbf{x}}_j^s\|^2 + \frac{\alpha}{N_t^2} \sum_{i,j} S_{i,j}^t \|\hat{\mathbf{x}}_i^t - \hat{\mathbf{x}}_j^t\|^2$$

  where $0 \leq \alpha \leq 1$ is a term that weighs the importance of the source/target regularizations.

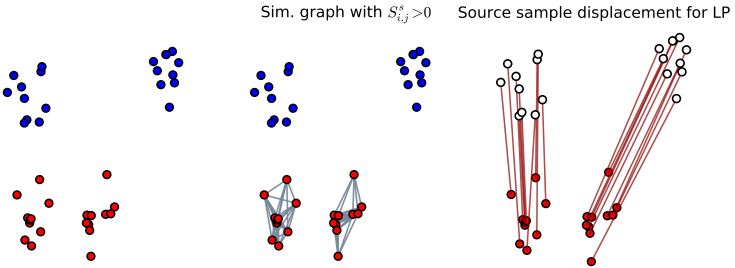
Example of LOT$_{pos}$



Discussion

- Similarity graph $\mathbf{S}^s$ obtained by a Gaussian kernel with threshold (left).
- The transported samples are illustrated in white for a small regularization (center) and a large regularization (right).
- Clusters and local structures are promoted.
- Per-cluster shrinkage for large regularization.

Laplacian regularization for sample displacement



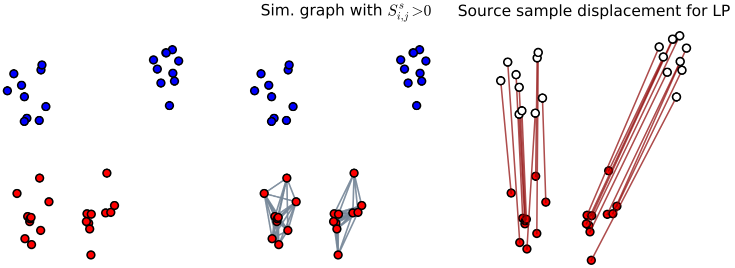
Graph regularization for the sample displacement

- Proposed in [2] for color transfer in images.
- $\hat{\mathbf{x}}^s_i - \mathbf{x}^s_i$ is the displacement of source sample $\mathbf{x}^s_i$ during transport.
- We want similar samples to have similar displacements:

$$\Omega(\boldsymbol{\gamma}) = \frac{1}{N_s^2} \sum_{i,j} S^s_{i,j} \|(\hat{\mathbf{x}}^s_i - \mathbf{x}^s_i) - (\hat{\mathbf{x}}^s_j - \mathbf{x}^s_j)\|^2$$

- Quadratic regularization term.

Laplacian regularization for sample displacement (2)



Reformulation of the regularization term

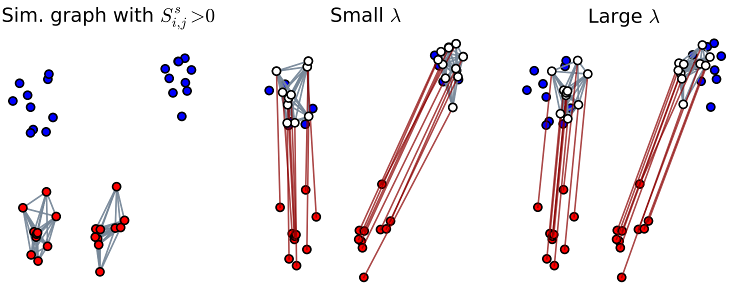
- The regularization term can be expressed in matrix form as:

$$\Omega_{disp}(\boldsymbol{\gamma}) = \mathrm{Tr}(\mathbf{X}_t^\top \boldsymbol{\gamma}^\top \mathbf{L}_s \boldsymbol{\gamma} \mathbf{X}_t) + \left\langle \boldsymbol{\gamma}, -N_s(\mathbf{L}_s + \mathbf{L}_s^\top)\mathbf{X}_s\mathbf{X}_t^\top) \right\rangle_F + c_s$$

  with $c_s = \mathrm{Tr}(\mathbf{X}_s^\top \mathbf{L}_s \mathbf{X}_s)$ a constant *w.r.t.* $\boldsymbol{\gamma}$.

- Similarly to $\Omega_{pos}(\cdot)$, one can use a symmetric regularization for the displacement.

Example of LOT$_{disp}$



Discussion

- Similarity graph $\mathbf{S}^s$ obtained by a Gaussian kernel with threshold (left).
- The transported samples displacements are illustrated in red for a small regularization (center) and a large regularization (right).
- Clusters and local structures are promoted.
- Per-cluster rigid translation for large regularization.

Optimization with Frank-Wolfe Algorithm

- Resulting optimization problem for both Laplacian regularization is a quadratic Program (QP).
- [2] proposed to use a Frank-Wolfe Algorithm to solve the problem.

Algorithm for symmetric regularization of the sample position

0. Initialize $k = 0$ and $\boldsymbol{\gamma}^0 \in \mathcal{P}$.

1. Compute the solution of the linear problem $\boldsymbol{\gamma}^* = \arg\min_{\boldsymbol{\gamma}\in\mathcal{P}} \langle \boldsymbol{\gamma}, \mathbf{C}_k \rangle_F$ with

$$\mathbf{C}_k = \mathbf{C} + (1-\alpha)(\mathbf{L} + \mathbf{L}^\top)\boldsymbol{\gamma}\mathbf{X}_t\mathbf{X}_t^\top + \alpha\mathbf{X}_s\mathbf{X}_s^\top\boldsymbol{\gamma}(\tilde{\mathbf{L}} + \tilde{\mathbf{L}}^\top)$$

2. Find the optimal step $0 \leq \alpha^k \leq 1$ with descent direction $\Delta\boldsymbol{\gamma} = \boldsymbol{\gamma}^* - \boldsymbol{\gamma}^k$ such that

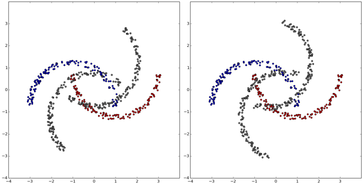
$$\alpha^k = -\frac{1}{2}\frac{\langle \Delta\boldsymbol{\gamma}, \mathbf{C}\rangle_F + \lambda_s \mathrm{Tr}(\mathbf{X}_t^\top \Delta\boldsymbol{\gamma}^\top(\mathbf{L}_s + \mathbf{L}_s^\top)\boldsymbol{\gamma}^k\mathbf{X}_t) + \lambda_t \mathrm{Tr}(\mathbf{X}_s^\top \Delta\boldsymbol{\gamma}(\mathbf{L}_t + \mathbf{L}_t^\top)\boldsymbol{\gamma}^{k\top}\mathbf{X}_s)}{\lambda_s \mathrm{Tr}(\mathbf{X}_t^\top \Delta\boldsymbol{\gamma}^\top \mathbf{L}_s \Delta\boldsymbol{\gamma}\mathbf{X}_t) + \lambda_t \mathrm{Tr}(\mathbf{X}_s^\top \Delta\boldsymbol{\gamma}\mathbf{L}_t\Delta\boldsymbol{\gamma}^\top\mathbf{X}_s)}$$

3. $\boldsymbol{\gamma}^{k+1} \leftarrow \boldsymbol{\gamma}^k + \alpha^k\Delta\boldsymbol{\gamma}$, set $k \leftarrow k+1$ and go to step 1.

Numerical experiments

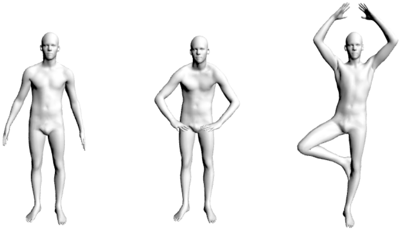
Domain adaptation (DA)

- Classification problem (blue VS red).
- Classical simulated *two-moons* problem.
- Increasing adaptation difficulty.
- Comparison with state of the art DA.



Non-rigid shape matching

- Register 3D shapes.
- Use the FAUST dataset[3].
- Preliminary matching results.
- Mean average error for transported vertices.



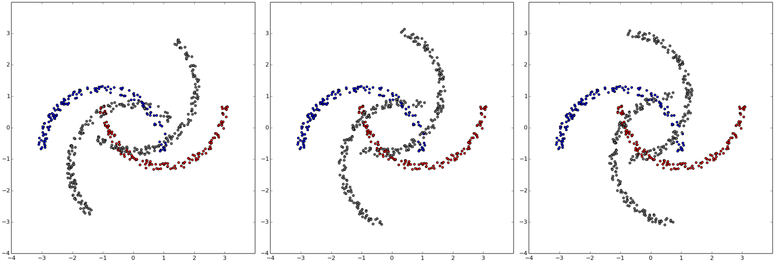
Domain adaptation problem

	10°	20°	30°	40°	50°	70°	90°
SVM (no adapt)	0.000	0.104	0.24	0.312	0.4	0.764	0.828
DASVM [4]	0.000	0.000	0.259	0.284	0.334	0.747	0.82
PBDA [5]	0.000	0.094	0.103	0.225	0.412	0.626	0.687
OT LP	0.000	0.000	0.031	0.102	0.166	0.292	0.441
OT$_{sinkhorn}$	0.000	0.000	0.000	0.000	0.013	0.202	0.386
LOT$_{pos}$	0.000	0.000	0.000	0.000	0.000	0.022	0.152
LOT$_{disp}$	0.000	0.000	0.000	0.000	0.000	0.067	0.384

Discussion

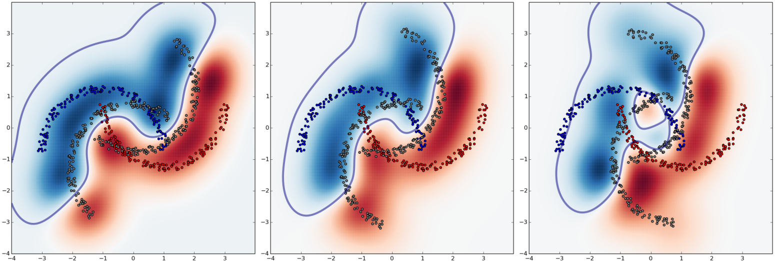
- Non-linear classification problem handled by a SVM with Gaussian kernel with parameters set by k-fold validation.
- Regularization parameters set empirically.
- Mean error rate over 10 samplings reported in the table.
- Good performances of Optimal transport for domain adaptation.
- Better performance of LOT$_{pos}$ for this problem.

Domain adaptation problem (2)



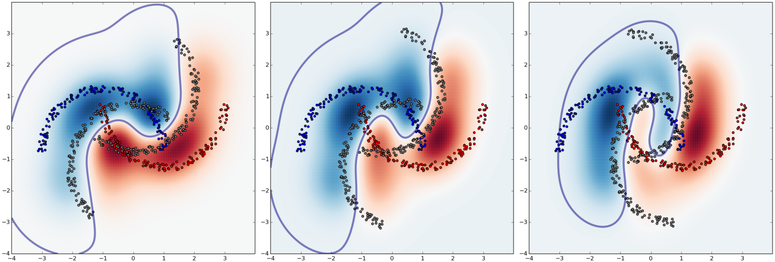
- **Domain adaptation problem for different rotations.**
- Decision function for LP transport (no regularization).
- Decision function for entropy based regularization ($\mathrm{OT}_{sinkhorn}$).
- Decision function for position based laplacian regularization (LOT_{pos}).

Domain adaptation problem (2)



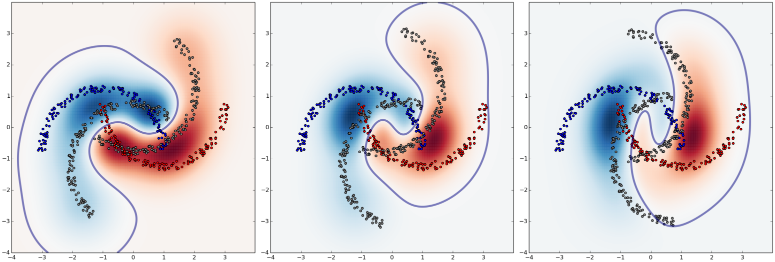
- Domain adaptation problem for different rotations.
- **Decision function for LP transport (no regularization).**
- Decision function for entropy based regularization ($\mathrm{OT}_{sinkhorn}$).
- Decision function for position based laplacian regularization (LOT_{pos}).

Domain adaptation problem (2)



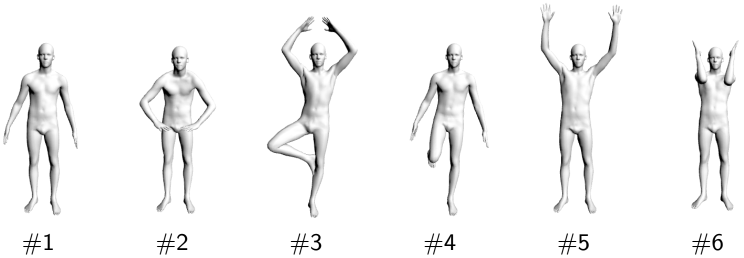
- Domain adaptation problem for different rotations.
- Decision function for LP transport (no regularization).
- **Decision function for entropy based regularization ($\mathrm{OT}_{sinkhorn}$).**
- Decision function for position based laplacian regularization (LOT_{pos}).

Domain adaptation problem (2)



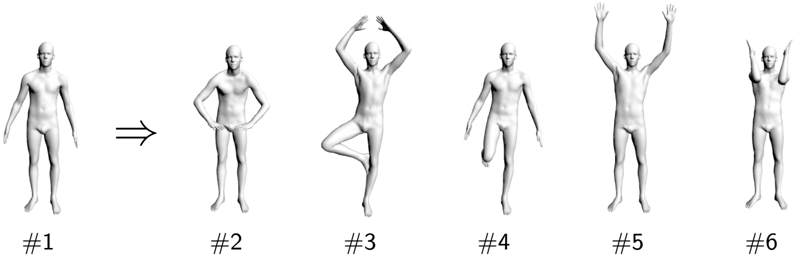
- Domain adaptation problem for different rotations.
- Decision function for LP transport (no regularization).
- Decision function for entropy based regularization ($\mathrm{OT}_{sinkhorn}$).
- **Decision function for position based laplacian regularization ($\mathbf{LOT}_{pos}$).**

Shape matching problem



- We want to match different shapes from the FAUST dataset.
- Ground truth available (exact matching between watertight meshes).
- Performance measure: distance of the transported vertices to the true target vertices (mean and max).
- 3D plot of the vertices assignments.

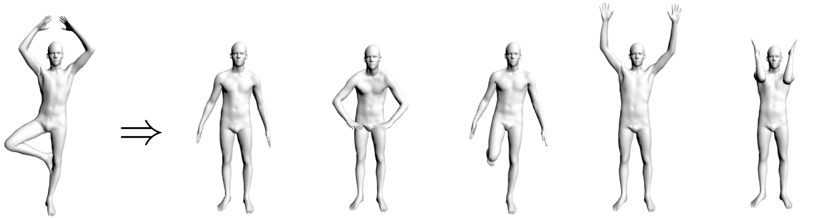
Shape matching problem (2)



	method	1	2	3	4	5	6
	OT LP	–	8.0 (54.5)	46.3 (141.2)	3.4 (57.6)	50.1 (160.9)	34.0 (125.3)
1	LOT$_{pos}$	–	7.5 (37.2)	44.0 (133.2)	3.7 (41.1)	46.5 (139.1)	31.1 (102.5)
	LOT$_{disp}$	–	7.1 (38.9)	44.3 (132.0)	3.5 (42.6)	47.9 (142.5)	31.9 (108.0)

- Mean error in cm (max errors in cm).
- Laplace regularization slightly better than classic OT.
- LOT$_{pos}$ and LOT$_{disp}$ have similar performances in average.
- Encouraging preliminary results.

Shape matching problem (2)

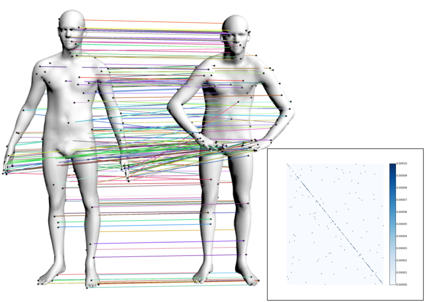


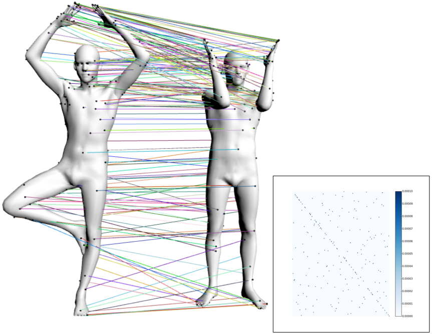
#3 ⇒ #1 #2 #4 #5 #6

	method	1	2	3	4	5	6
3	OT LP	51.3 (113.7)	41.3 (85.2)	–	48.5 (113.2)	6.1 (50.3)	11.4 (49.4)
	LOT_{pos}	49.2 (108.7)	39.1 (78.7)	–	46.4 (107.7)	6.0 (47.5)	11.0 (46.9)
	LOT_{disp}	51.4 (109.7)	40.8 (80.0)	–	48.7 (108.8)	5.7 (48.3)	10.8 (46.0)

- Mean error in cm (max errors in cm).
- Laplace regularization slightly better than classic OT.
- LOT_{pos} and LOT_{disp} have similar performances in average.
- Encouraging preliminary results.

Shape matching problem (3)





Two matching examples

- Lines show vertex displacement.
- Vertices are sorted : perfect assignment means a diagonal matrix.
- Left: $\#1 \Rightarrow \#2$, Error $=7.5$ (37.2)
- Right: $\#3 \Rightarrow \#6$, Error $= 11.0$ (46.9).

Conclusion

Optimal transport with Laplacian regularization

- When data has a graph structure, regularize to keep it during the transport.
- Two flavors of Laplacian regularization (depends on the problem).
- Use Frank-Wolfe to solve the problem (efficient LP solvers, early stopping).
- Encouraging results on two applications.

Next steps

- More numerical experiments on real life datasets.
- Use label during graph computation for domain adaptation.
- Large scale optimization procedure.
- Find the regularization parameters automatically.

References

M. Cuturi.
Sinkhorn distances: Lightspeed computation of optimal transportation.
In *NIPS*, pages 2292–2300. 2013.

S. Ferradans, N. Papadakis, G. Peyré, and J-F. Aujol.
Regularized discrete optimal transport.
SIAM Journal on Imaging Sciences, 7(3), 2014.

Federica Bogo, Javier Romero, Matthew Loper, and Michael J. Black.
FAUST: Dataset and evaluation for 3D mesh registration.
In *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, Piscataway, NJ, USA, June 2014. IEEE.

L. Bruzzone and M. Marconcini.
Domain adaptation problems: A dasvm classification technique and a circular validation strategy.
Pattern Analysis and Machine Intelligence, IEEE Transactions on, 32(5):770–787, May 2010.

P. Germain, A. Habrard, F. Laviolette, and E. Morvant.
A PAC-Bayesian Approach for Domain Adaptation with Specialization to Linear Classifiers.
In *ICML*, pages 738–746, Atlanta, USA, 2013.