SLICED PARTIAL OPTIMAL TRANSPORT

Nicolas Bonneel*, David Coeurjolly*

*CNRS, Univ. Lyon.

© 2019 SIGGRAPH. ALL RIGHTS RESERVED.
Matching points with optimal transport

- Monge (Linear Assignment Problem)

\[
\min_{T \text{ bijective}} \sum_{i} c(x_i, y_{T(i)})
\]

- Kantorovich

\[
W(f, g) = \min \sum_{i,j} c_{i,j} \pi_{i,j}
\]
\[
\text{s.t.} \quad \sum_{j} \pi_{i,j} = 1
\]
\[
\sum_{i} \pi_{i,j} = 1
\]
\[
\pi_{i,j} \geq 0
\]
1-d Linear Assignment Problem is trivial*

*assuming the cost $c$ is a convex function of $|x-y|$
Partial optimal assignment?

\[ W(f, g) = \min \sum_{i,j} c_{i,j} \pi_{i,j} \quad \text{s.t.} \quad \sum_j \pi_{i,j} = 1, \]
\[ \sum_i \pi_{i,j} \leq 1, \]
\[ \pi_{i,j} \geq 0 \]

\[ \min_{T \text{ injective}} \sum_i c(x_i, y_{T(i)}) \]
Similar problems

• DNA sequence alignment
• Text alignment
• Music synchronization
• ...
Existing solutions

- Dynamic Time Warping
  - Solves a dynamic programming problem
- Smith–Waterman algorithm, Needleman–Wunsch algorithm \( O(N^2) \) space and time
- Hirschberg's algorithm \( O(N^2) \) time, \( O(N) \) space
- All end up doing variants of
  - \( A_{i,j} = \min(A_{i-1,j-1} + \text{cost}, A_{i-1,j} + \text{cost}', A_{i,j-1} + \text{cost}'') \)
Quadratic time complexity algorithm (linear space)
Quadratic time complexity algorithm (linear space)

\[ X \]

\[ Y \]
Quadratic time complexity algorithm (linear space)

Euclidean Nearest Neighbor assignment
Quadratic time complexity algorithm (linear space)
Quadratic time complexity algorithm (linear space)

Euclidean Nearest Neighbor assignment

Optimal Transport assignment

Intervals of bijective assignments
Quadratic time complexity algorithm (linear space)
Quadratic time complexity algorithm (linear space)
Euclidean Nearest Neighbor assignment

Optimal Transport assignment

Intervals of bijective assignments

Quadratic time complexity algorithm (linear space)
Quadratic time complexity algorithm (linear space)

Euclidean Nearest Neighbor assignment

Optimal Transport assignment

Intervals of bijective assignments
Quadratic time complexity algorithm (linear space)
Quadratic time complexity algorithm (linear space)
Quadratic time complexity algorithm (linear space)

Euclidean Nearest Neighbor assignment

Optimal Transport assignment

Intervals of bijective assignments
Quadratic time complexity algorithm (linear space)

Euclidean Nearest Neighbor assignment

Optimal Transport assignment

Intervals of bijective assignments
Quadratic time complexity algorithm (linear space)
Euclidean Nearest Neighbor assignment
Optimal Transport assignment
Intervals of bijective assignments

Quadratic time complexity algorithm (linear space)
Quadratic time complexity algorithm (linear space)

Euclidean Nearest Neighbor assignment

Optimal Transport assignment

Intervals of bijective assignments
Quadratic time complexity algorithm (linear space)

- Euclidean Nearest Neighbor assignment
- Optimal Transport assignment

Intervals of bijective assignments
Quadratic time complexity algorithm (linear space)
Quadratic time complexity algorithm (linear space)

Euclidean Nearest Neighbor assignment

Optimal Transport assignment

Intervals of bijective assignments
Quadratic time complexity algorithm (linear space)
Quadratic time complexity algorithm (linear space)
Quadratic time complexity algorithm (linear space)
Quadratic time complexity algorithm (linear space)
Quadratic time complexity algorithm (linear space)

Euclidean Nearest Neighbor assignment

Optimal Transport assignment

Intervals of bijective assignments
Quadratic time complexity algorithm (linear space)
Quadratic time complexity algorithm (linear space)

Euclidean Nearest Neighbor assignment
Optimal Transport assignment
Intervals of bijective assignments
Euclidean Nearest Neighbor assignment
Optimal Transport assignment
Intervals of bijective assignments

Quadratic time complexity algorithm (linear space)
Euclidean Nearest Neighbor assignment
Optimal Transport assignment

Intervals of bijective assignments

Quadratic time complexity algorithm (linear space)
Quadratic time complexity algorithm (linear space)

Euclidean Nearest Neighbor assignment

Intervals of bijective assignments

Optimal Transport assignment
Linear time problem decomposition
Problem decomposition

\[ X \]

\[ Y \]
Problem decomposition
Problem decomposition

- Computed in quasi-linear time via Union-Find
  - 1 M points in a fraction of a second

- Yields independent subproblems
  - Solvable in parallel
  - That can be further simplified (see paper)
Sliced Partial Optimal Transport (SPOT)
Extension to \(d\) dimensions

- Sliced optimal transport

\[
E = \int_{S^{d-1}} W(P_\omega X, P_\omega Y) d\omega = \int_{S^{d-1}} \min_T \sum_i \left( P_\omega x_i - P_\omega y_{T(i)} \right)^2 d\omega
\]
Gradient flow

- Sliced optimal transport

\[ X^{n+1} = X^n - \nabla E \]

Stochastic descent: 
\[ X^{n+1} = X^n - \nabla W(P_{\omega^n X}, P_{\omega^n Y}). \omega^n \]
Gradient flow

- Sliced optimal transport

\[ X^{n+1} = X^n - \nabla E \]

Stochastic descent: \( X^{n+1} = X^n - \nabla W(P_{\omega^n X}, P_{\omega^n Y}) \cdot \omega^n \)
Color transfer application

Full Transfer

Target 20% larger

Target 40% larger
Color transfer application

Full Transfer  Target 20% larger  Target 40% larger
Fast Iterative Sliced Transport (FIST)
Source: 8k samples
Target: 10k samples

ICP
(0.005 s / iteration)

Iterative Transport with network simplex
(40 s / iteration)

Our FIST algorithm
(0.04 s / iteration)
Source: 90k samples
Target: 100k samples

ICP
(0.05 s / iteration)

Our FIST algorithm
(0.66 s / iteration)
Source: 90k samples
Target: 100k samples

ICP
(0.05 s / iteration)

Our FIST algorithm
(0.69 s / iteration)

(input too large for iterative transport with network simplex)
Source: 150k samples
Target: 200k samples

ICP
(0.09 s / iteration)

Our FIST algorithm
(2.18 s / iteration)

(input too large for iterative transport with network simplex)
Failure case: the transport is optimal only on projections
Conclusions

• Fast partial optimal transport in 1d
  • Quadratic-time algorithm (worst case)
  • Quasi-linear time decomposition

• Sliced Partial Optimal Transport

• Fast Iterative Sliced Transport

• Applications: point cloud registration, color matching

• Code available: https://perso.liris.cnrs.fr/nicolas.bonneel/spot/