Approximate Wasserstein Metric
and its Application to Image Processing

Julien Rabin ♢, with Gabriel Peyré ♠, Julie Delon ♡ and Marc Bernot ♣

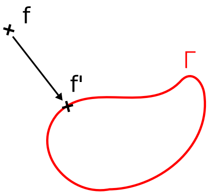
♢	GREYC, Université de Caen	*julien.rabin@unicaen.fr*
♠	CEREMADE, CNRS, Paris Dauphine	*gabriel.peyre@ceremade.dauphine.fr*
♡	CNRS, LTCI, Télécom ParisTech	*julie.delon@telecom-paristech.fr*
♣	Thalès Alenia Space	*marc.bernot@thalesaleniaspace.com*

Journées Traitement de l'image, 24–25 November 2011, Marseille

Problem Statement

Let f be a given signal and $S[f]$ the statistical distribution of some of its characteristics.

Problem 1: match the considered statistics of f to some desired statistical distribution S_0.



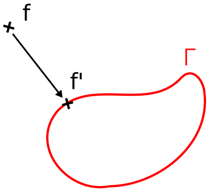
$$f' = \textbf{Proj}_{\Gamma}(f) \quad \textbf{where} \;\; \Gamma = \{u | S[u] = S_0\}$$

Example: Contrast Enhancement

Problem Statement

Let f be a given signal and $S[f]$ the statistical distribution of some of its characteristics.

Problem 1: match the considered statistics of f to some desired statistical distribution S_0.



$$f' = \textbf{Proj}_{\Gamma}(f) \quad \textbf{where} \;\; \Gamma = \{u | S[u] = S_0\}$$

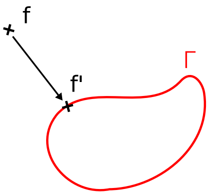
Example: Contrast Enhancement

1-D Histogram equalization

Problem Statement

Let f be a given signal and $S[f]$ the statistical distribution of some of its characteristics.

Problem 1: match the considered statistics of f to some desired statistical distribution S_0.



$$f' = \textbf{Proj}_{\Gamma}(f) \quad \textbf{where} \quad \Gamma = \{u | S[u] = S_0\}$$

Example: Contrast Enhancement

1-D Histogram equalization

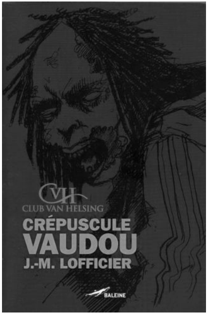
Extension to **multi-dimensional** statistics ?

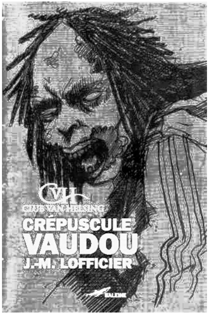
Problem Statement

Modifying statistics yields many **artifacts**: noise enhancement, detail loss, "bloquing effect" (JPEG compression), color inconsistencies . . .

Problem 2: **restore** the texture and the geometry of the original image.

Example: **Noise reduction for equalization**





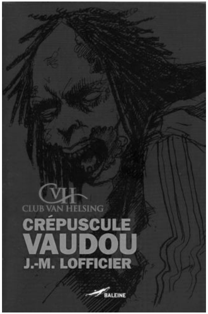
Example with histogram equalization

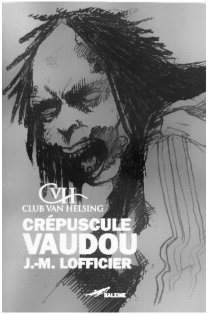
Problem Statement

Modifying statistics yields many **artifacts**: noise enhancement, detail loss, “bloquing effect” (JPEG compression), color inconsistencies . . .

Problem 2: **restore** the texture and the geometry of the original image.

Example: **Noise reduction for equalization** **Filtering technique**





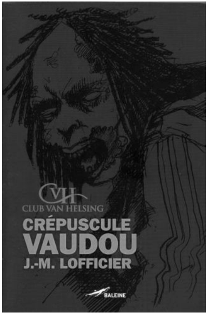
Regularization using non-local transfer [Rabin et al., 2011]

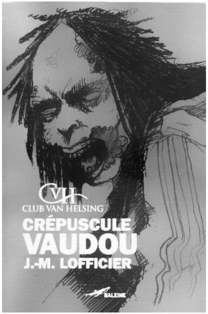
Problem Statement

Modifying statistics yields many **artifacts**: noise enhancement, detail loss, "bloquing effect" (JPEG compression), color inconsistencies . . .

Problem 2: restore the texture and the geometry of the original image.

Example: Noise reduction for equalization **Filtering technique**





Statistical **matching** and **regularization within the same framework** ?

Problem Statement

Let $\{f_i\}_{i=1,\dots,K}$ be a set of K signals and $\{S[f_i]\}_i$ their statistics.

Problem 3: Compute **average statistics** of $\{f_i\}_i$.

Simple example: Flicker correction of old movies

Problem Statement

Let $\{f_i\}_{i=1,\ldots,K}$ be a set of K signals and $\{S[f_i]\}_i$ their statistics.

Problem 3: Compute **average statistics** of $\{f_i\}_i$.

Simple example: Flicker correction of old movies

Midway-histogram [Delon, 2004]

Problem Statement

Let $\{f_i\}_{i=1,\dots,K}$ be a set of K signals and $\{S[f_i]\}_i$ their statistics.

Problem 3: Compute **average statistics** of $\{f_i\}_i$.

Simple example: Flicker correction of old movies

Extension to **multi-dimensional** statistics ?

Outline

General problem: **regularize** and **average** images under **multi-dimensional** statistical constraints

Part I. Multi-dimensional statistic **specification** (“Wasserstein projection”)

Part II. Variational **regularization** under statistical constraints (“Wasserstein regularization”)

Part III. Multi-dimensional statistics **averaging** (“Wasserstein Barycenter”)

Applications to many image processing and computer vision tasks:

- **Color transfer**;
- Non-rigid **shape matching**;
- **Texture synthesis** and mixing from exemplar images;

Methodology: **Optimal mass transportation problem** framework.

Part I

Wasserstein Projection

Formulation in general case

Let f and g be two probability distributions in $\mathbb{R}^d$ ($f, g > 0$ and $\int f = \int g = 1$).

Monge-Kantorovich optimal mass transportation problem
Let $c : \mathbb{R}^d \times \mathbb{R}^d \mapsto \mathbb{R}^+$ be a nonnegative cost function ("**ground cost**").
Optimal transport theory defines a **cost** and a **transportation flow** between two measures [Villani, 2008]

$$\mathrm{MK}(f,g) := \inf_{\pi \in \Pi_{\{f,g\}}} \iint_{x,y} c(x,y)\, d\pi(x,y) \,, \qquad (1)$$

where $\Pi_{\{f,g\}}$ is the set of probability measures on $\mathbb{R}^d \times \mathbb{R}^d$ with marginals f and g ("***transport plans***").

Wasserstein distance of order p when using L^2 metric for the ground cost

$$W_p(f,g) = \left(\inf_{\pi \in \Pi_{\{f,g\}}} \iint_{x,y \in \mathbb{R}^d} \|x-y\|^p \pi(dx,dy) \right)^{\frac{1}{p}} \,, \qquad (2)$$

where $\|.\|$ is the Euclidean norm.

Remark: Earth Mover's Distance (EMD) [Rubner et al., 2000]

Formulation for point clouds

Definition: Wasserstein Distance Given two point clouds $X, Y \subset \mathbb{R}^{d\times N}$ of N elements in $\mathbb{R}^d$ with **equal masses**, the **quadratic Wasserstein distance** is defined as

$$W_2(X, Y)^2 = \min_{\sigma \in \Sigma_N} \sum_{i \in I} |X_i - Y_{\sigma(i)}|^2 \qquad (3)$$

where Σ_N is the set of all permutations of N elements, and $I = \{1, \ldots, N\}$.

$\Leftrightarrow$ **Optimal Assignment problem**

The corresponding **Wasserstein projection** of X on Y is therefore

$$\forall i \in I \quad \Big(W_2\text{-Proj}_{[Y]}(X)\Big)_i = X_i^* = Y_{\sigma^*(i)} \ , \qquad (4)$$

where σ^* is the optimal permutation of (3).

Exact solution in unidimensional case ($d = 1$)

Computing the L^2-Wasserstein projection in the **one-dimensional** case is simple.

Algorithm: If one denotes by σ_X and σ_Y the permutations that order the points

$$\forall\, 0 \leqslant i < N-1, \quad X_{\sigma_X(i)} \leqslant X_{\sigma_X(i+1)} \quad \text{and} \quad Y_{\sigma_Y(i)} \leqslant Y_{\sigma_Y(i+1)} \tag{5}$$

the optimal permutation $\sigma^\star$ that minimizes (3) is

$$\sigma^* = \sigma_Y \circ \sigma_X^{-1}, \tag{6}$$

so that point $X_{\sigma_X(i)}$ **is assigned to the point** $Y_{\sigma_Y(i)}$.

Time complexity: $O(N\log(N))$ operations using a fast sorting algorithm.

Application: Histogram equalization and specification (see *e.g.* [Nikolova *et al.* 2011]).

Exact solution in general case (d>1)

It is possible to recast the optimal assignment problem as **a linear programming** one

$$W_2(X,Y)^2 = \min_{P \in \mathcal{P}_N} \sum_{i,j \in I^2} P_{i,j} |X_i - Y_j|^2 \quad (7)$$

where $\mathcal{P}_N$ is the set of **bistochastic matrices**.

The **relaxed** problem (7) can be solved with **standard linear programming algorithms** (*e.g.* simplex and interior point method).

Remark 1: optimal transport matrices are assignment matrices (*i.e.* $P_{i,j} \in \{0,1\}$)

Remark 2: some dedicated algorithms are more efficient for optimal assignment problem (*e.g* **Hungarian and Auction algorithms** in $O(N^3)$)

Remark 3: computation can be accelerated when using other ground costs than L^2 (*e.g.* L_1 [Ling and Okada, 2007], Truncated L_1 [Pele and Werman, 2008])

Exact solution in general case (d>1)

It is possible to recast the optimal assignment problem as **a linear programming** one

$$W_2(X,Y)^2 = \min_{P \in \mathcal{P}_N} \sum_{i,j \in I^2} P_{i,j} |X_i - Y_j|^2 \quad (7)$$

where $\mathcal{P}_N$ is the set of **bistochastic matrices**.

The **relaxed** problem (7) can be solved with **standard linear programming algorithms** (*e.g.* simplex and interior point method).

Remark 1: optimal transport matrices are assignment matrices (*i.e.* $P_{i,j} \in \{0,1\}$)

Remark 2: some dedicated algorithms are more efficient for optimal assignment problem (*e.g* **Hungarian and Auction algorithms** in $O(N^3)$)

Remark 3: computation can be accelerated when using other ground costs than L^2 (*e.g.* L_1 [Ling and Okada, 2007], Truncated L_1 [Pele and Werman, 2008])

Limitation

Intractable for signal processing applications where $N \gg 10^3$ (time complexity & memory limitation)

Approximate solution: previous works

Solution

Approximation of optimal transportation problem

Previous works:

- Lower bounds on EMD [Guibas, 1997]
- EMD-**embedding** [Indyk and Thaper, 2003, Grauman and Darrell, 2004, Grauman and Darrell, 2005]
- approximation of EMD with **wavelet decomposition** (WEMD [Shirdhonkar and Jacobs, 2008])
- approximation of optimal transport with 1D-projections [Pitié et al., 2007]

Sliced Wasserstein projection

Sliced-Wasserstein Approximation

Let SW_2 be the Sliced Wasserstein Energy, defined for a given distribution Y at point Z as

$$\mathrm{SW}_2(Z,Y) = \int_{\theta\in\mathbb{S}^{d-1}} W_2(\langle Z,\theta\rangle,\langle Y,\theta\rangle)^2\, d\theta = \int_{\theta\in\mathbb{S}^{d-1}} E_\theta(Z,Y)\, d\theta \ , \tag{8}$$

where

$$E_\theta(Z,Y) = \min_{\sigma_\theta\in\Sigma_N} \sum_{i\in I} \langle Z_i - Y_{\sigma_\theta(i)},\theta\rangle^2$$

Sliced-Wasserstein Projection

To approximate the optimal transport of a point cloud X on a given discrete distribution Y, **evolve progressively X towards Y** in such a way that the **Sliced Wasserstein energy is decreasing**.

$\Rightarrow$ **gradient descent** algorithm

Sliced Wasserstein projection gradient descent

Batch Gradient Descent algorithm

- **Initialization:** Set $X^{(0)} := X$. Define a set of orientations $\Psi := \{\theta \in \mathbb{S}^{d-1}\}$ (s.t. $|\Psi| > d$)
- **Iteration:**
  - ▷ **Step 1:** For each $\theta \in \Psi$ compute the minimizer σ_θ^* of

$$E_\theta(X^{(k)}, Y) = \min_{\sigma_\theta \in \Sigma_N} \sum_{i \in I} \langle X_i^{(k)} - Y_{\sigma_\theta(i)}, \theta \rangle^2 ;$$

  - ▷ **Step 2:** For a given gradient step parameter $\lambda \leqslant 1$, $\forall\, i \in I$

$$X_i^{(k+1)} = X_i^{(k)} - \lambda \cdot H_\Psi^{-1}(X_i^{(k)}) \times \sum_{\theta \in \Psi} \left(\nabla E_\theta(X^{(k)}, Y) \right)_i ,$$

  where H_Ψ is the Hessian matrix.

- **Output:** The sliced Wasserstein projection of X onto Y is defined as $X^{(\infty)}$.

Sliced Wasserstein projection gradient descent (II)

$\forall i \in I$, and for a given set $\{\sigma_\theta^*\}_{\theta\in\Psi}$, **Gradient** and **Hessian** can be expressed as

$$\begin{aligned}\sum_{\theta\in\Psi} \nabla E_\theta(X_i) &= \sum_{\theta\in\Psi} \langle X_i - Y_{\sigma_\theta^*(i)}, \theta\rangle . \theta \\ &= \Big(\sum_{\theta\in\Psi} \theta . \theta^T\Big) \cdot X_i - \sum_{\theta\in\Psi} \theta . \theta^T \cdot Y_{\sigma_\theta^*(i)}\end{aligned}$$

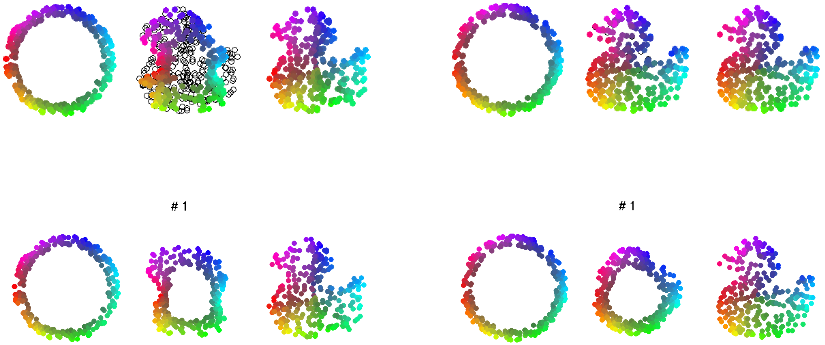
$$H_\Psi = \sum_{\theta\in\Psi} \nabla^2 E_\theta(X_i) = \sum_{\theta\in\Psi} \theta . \theta^T = \Theta$$

Note: H_Ψ^{-1} is precomputed.

Convergence: the energy $\mathrm{SW}_2(X^{(k)}, Y)$ is strictly decreasing w.r.t. k and $X^{(k)}$ converges towards a local minimum of the energy.

Results with gradient descent

Projection results with respectively $|\Psi| = 2d$ and $|\Psi| = 100d$



Remark: An interesting variant when using ***stochastic gradient descent***

Go

Application to Color Transfer

Source image (X)

Style image (Y)

Application to Color Transfer

Source image (X)

Style image (Y)

Sliced Wasserstein projection of X to style image color statistics Y

Source image after color transfer

Application to Color Transfer

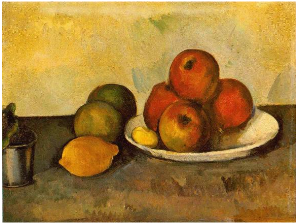
Source image (X)

Style image (Y)

Sliced Wasserstein projection of X to style image color statistics Y

Source image after color transfer

Application to Color Transfer



Source image (X)

Style image (Y)

Sliced Wasserstein projection of X to style image color statistics Y

Source image after color transfer

Application to Color Transfer

Source image (X)

Style image (Y)

Sliced Wasserstein projection of X to style image color statistics Y

Source image after color transfer

Application to Color Transfer

Source image (X)

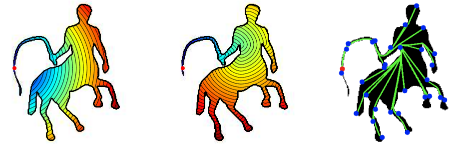
$X \mapsto Y$

Style image (Y)

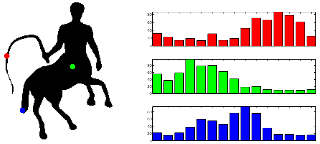
$Y \mapsto X$

Bending invariant shape comparison

Goal: articulated shapes comparison



Euclidean dist. *Geodesic dist.* *Geodesic paths*



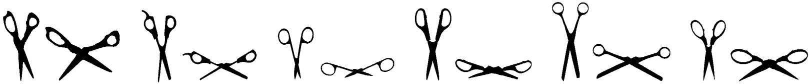
1-D geodesic distances distributions for three different starting points.

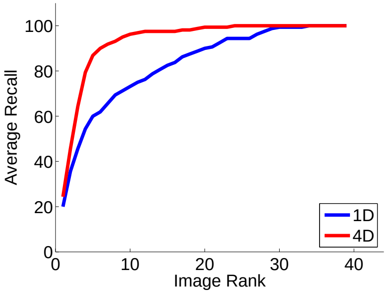
Idea: Use **multi-dimensional** geodesic statistics (extension of [Ion et al., 2007])

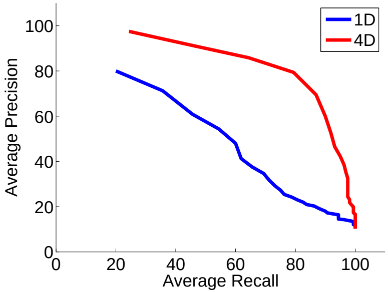
Example of quantile distributions (Min, Median and Max) inside a planar shape, and the corresponding joint-distribution.

Bending invariant shape comparison

Idea: Given two point-clouds X and Y, use $||X - X^{(\infty)}||$ as a similarity measure for descriptor comparison, where $X^{(\infty)}$ is the Sliced-Wasserstein projection of X onto Y.



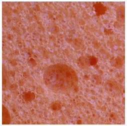




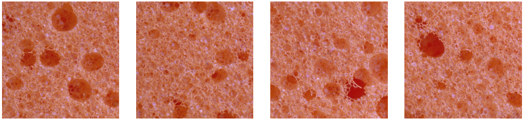
Texture synthesis with Heeger and Bergen algorithm

Let be Y a color texture exemplar $Y : x \in \Omega \mapsto Y(x) \in \mathbb{R}^3$ of N pixels.

Objective: Generate a new **random** texture with the **same visual aspect**.



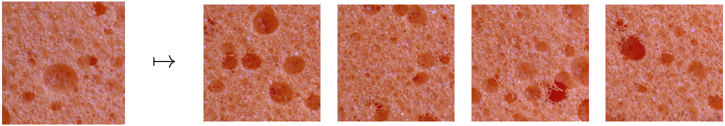
$\mapsto$



Texture synthesis with Heeger and Bergen algorithm

Let be Y a color texture exemplar $Y : x \in \Omega \mapsto Y(x) \in \mathbb{R}^3$ of N pixels.

Objective: Generate a new **random** texture with the **same visual aspect**.



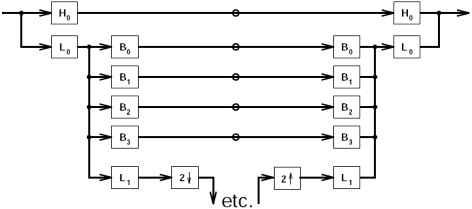
Principle of Heeger and Bergen algorithm: Texture synthesis through **iterated projections on statistical sets**.

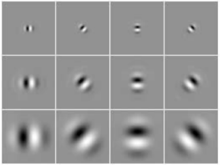
Sketch of the algorithm:

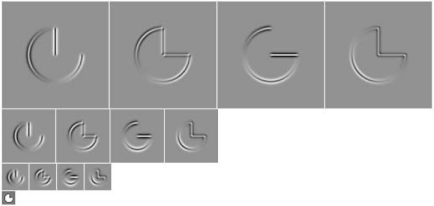
1– The texture exemplar image is analyzed *via* its projection on a set of atoms (**distribution of wavelet coefficients**).

2– A random image is generated and analyzed, and its statistics are modified to match the desired one (**1-D Wasserstein projection**).

3– The texture is synthesized by reconstruction (**tight frame**).


H0
H0
L0
B0
B0
L0
B1
B1
B2
B2
B3
B3
L1
2↓
etc.
2↑
L1

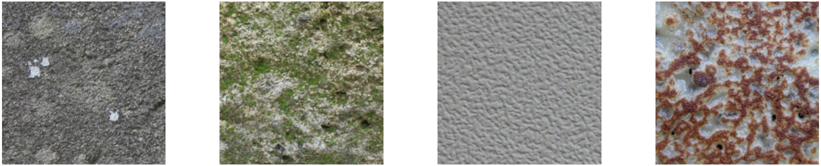




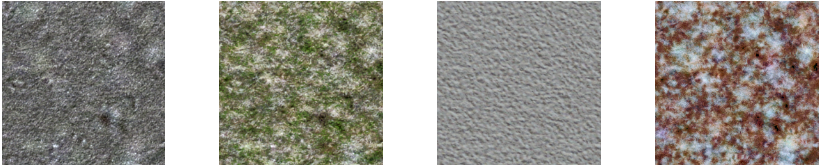
Examples of Texture synthesis with Heeger and Bergen algorithm

HB algorithm succeeds to synthesize "**micro-textures**"

Exemplar textures



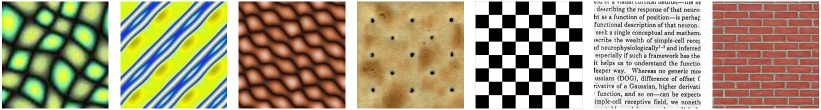
Synthesis



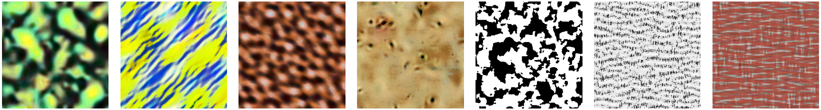
Examples of Texture synthesis with Heeger and Bergen algorithm

Strong limitation of HB approach: restriction to **1st order statistics**

Exemplar textures



Synthesis

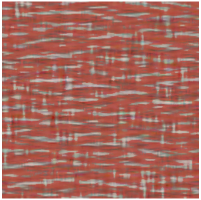


Portilla and Simoncelli extension

[Portilla and Simoncelli, 2000]: use of **2nd order statistics** (correlation of wavelet coefficients)

Original

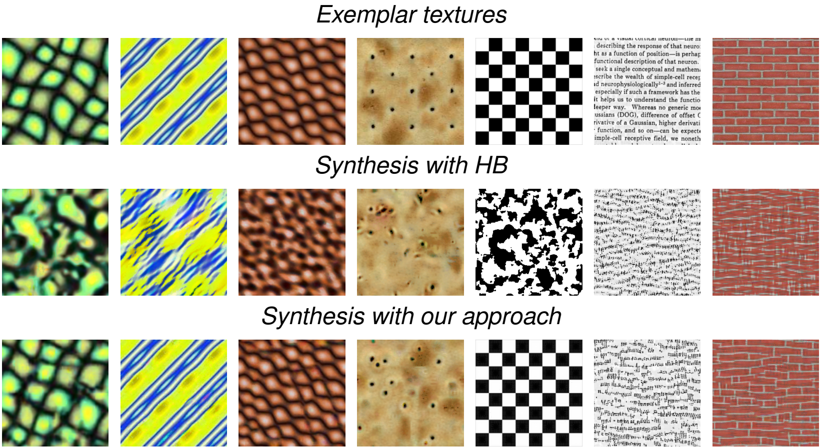
H-B.



P-S.

Examples of Texture synthesis with extended Heeger and Bergen algorithm

Extension of HB approach to **multi-dimensional statistics**



Part II

Wasserstein Regularization

General Formulation

Limitation of Wasserstein projection for color transfer: strong visual **artifacts**

Idea: define a variational framework to **regularize** under **statistical constraints**

Let u be the source image and v the style image, and let denote by $[u]$ and $[v]$ their respective color distribution.

Find the minimizer of

$$\min_{w \in \mathbb{R}^{d \times N}} \left\{ \mathcal{E}(w) = \mathbf{F}(\mathbf{w}, \mathbf{u}) + \lambda_{\mathrm{R}} \, \mathbf{R}(\mathbf{w}) + \lambda_{\mathrm{S}} \, \mathbf{S}([\mathbf{w}], [\mathbf{v}]) \right\} \; (*)$$

where

- **F** is the fidelity term (to preserve texture and geometry)
- **R** is the regularization penalty (denoising)
- **S** is the statistical constraint (here for color transfer)

Definition of penalty terms for color transfer

- The choice **F** and **R** strongly depends on the considered application. Here :
  - F defined as the sum of the quadratic loss and a level set consistency term [Ballester et al., 2006, Papadakis et al., 2010]

$$\mathbf{F}(\mathbf{w},\mathbf{u}) = \sum_{i\in\Omega} \left\{ \frac{\lambda_L}{2} \|w_i - u_i\|^2 - \lambda_{LS} \langle \nabla w_i \, , \, \frac{\nabla u_i}{\|\nabla u_i\|} \rangle \right\}$$

  - R defined as the color Total Variation [Rudin et al., 1992] penalty (TV):

$$\mathbf{R}(\mathbf{w}) = \|w\|_{\mathrm{TV}} = \sum_{i\in\Omega} \|\nabla w_i\| \, ,$$

- A general method to constraint statistics is to use $S = W_2$:

  Limitation: Using Wasserstein projection is computationally **prohibitive** in the multi-dimensional case !

  Solution: use the **Sliced Wasserstein** energy SW_2, which is differentiable

Variational formulation for color transfer problem:

$$\min_{w\in\mathbb{R}^{d\times N}} \left\{ \mathcal{E}(w) = F(w,u) + \lambda_{\mathrm{R}} \, \mathbf{R}(\mathbf{w}) + \lambda_{\mathrm{S}} \, \mathrm{SW}_\mathbf{2}([\mathbf{w}],[\mathbf{v}]) \right\} \ (\star)$$

Forward-backward algorithm

Problem ($\star$) is a **non-convex minimization** problem: we use a **forward-backward proximal** scheme to find a fixed point of energy $\mathcal{E}$.

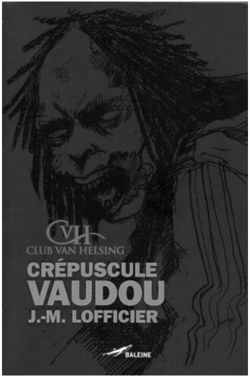
Starting from $w^{(0)} := u$, the update of the image $w^{(k)}$ at iteration k and point of coordinate $i \in \Omega$ depends on the two following Forward (F) and Backward (B) steps:

$$\begin{cases} w_i^{(k+\frac{1}{2})} = & w_i^{(k)} - \tau \left(\mathbf{F}'(\mathbf{w^{(k)}}, \mathbf{u})(\mathbf{i}) + \lambda_S \frac{\partial \mathrm{SW}_\mathbf{2}(\mathbf{w^{(k)}}, [\mathbf{v}])}{\partial w_i^{(k)}} \right) & \text{(F)} \\ w_i^{(k+1)} = & \operatorname{prox}_{\tau \cdot \lambda_\mathbf{R} \mathbf{R}} \left(w^{(k+\frac{1}{2})} \right)(i) & \text{(B)} \end{cases}$$

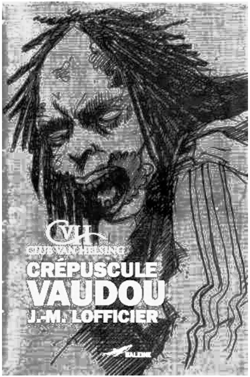
where

$$\mathbf{F}'(\mathbf{w^{(k)}}, \mathbf{u})(\mathbf{i}) = \lambda_L (w_i^{(k)} - u_i) + \lambda_{LS} \operatorname{div} \frac{\nabla u_i}{\|\nabla u_i\|} \ ,$$

Application to Contrast Enhancement (equalization)



Original image



W_2 Projection

W_2 Regularization

Regularized Color transfer

Source Image X

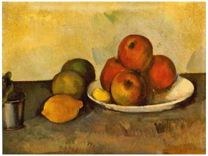
Style Image Y

SW_2 Projection

SW_2 Regularization

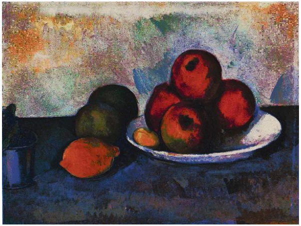
Zoom

Regularized Color transfer



Source Image X

Style Image Y



SW_2 Projection

SW_2 Regularization

Zoom

Regularized Color transfer

Source Image X

Style Image Y

SW_2 Projection

SW_2 Regularization

Zoom

Regularized Color transfer

Source Image X

Style Image Y

SW_2 Projection

SW_2 Regularization

Zoom

Regularized Color transfer

Source Image X

Style Image Y

SW_2 Projection

SW_2 Regularization

Part III

Approximate Wasserstein Barycenter

Wasserstein Barycenter definition for point clouds

Wasserstein Barycenter Given a family $\{Y^j\}_{j\in J}$ of point clouds, compute a **weighted average point cloud** $X^\star$, that is defined, by analogy to the Euclidean setting as the minimizer

$$X^\star := \mathrm{Bar}(\rho_j, Y^j)_{j\in J} \in \underset{X}{\operatorname{argmin}} \sum_{j\in J} \rho_j W_2(X, Y^j)^2, \tag{9}$$

where $\rho_j \geqslant 0$, is a set of weights, that is constrained to satisfy $\sum_j \rho_j = 1$.

Remark: See [Agueh and Carlier, 2011, Gangbo and Święch, 1998] for theoretical analysis

Wasserstein Barycenter definition for point clouds

Wasserstein Barycenter Given a family $\{Y^j\}_{j\in J}$ of point clouds, compute a **weighted average point cloud** $X^\star$, that is defined, by analogy to the Euclidean setting as the minimizer

$$X^\star := \mathrm{Bar}(\rho_j, Y^j)_{j\in J} \in \operatorname*{argmin}_X \sum_{j\in J} \rho_j W_2(X, Y^j)^2, \tag{9}$$

where $\rho_j \geqslant 0$, is a set of weights, that is constrained to satisfy $\sum_j \rho_j = 1$.

Remark: See [Agueh and Carlier, 2011, Gangbo and Święch, 1998] for theoretical analysis

Alternative formulation This barycenter can equivalently been computed in two steps, by first finding **a set of permutations** $\{\sigma_j^\star\}_{j\in J}$ maximizing

$$\{\sigma_j^\star\}_{j\in J} \in \max_{\{\sigma_j\}_{j\in(\Sigma_N)^{|J|}}} \sum_{i\in I} C(\sigma_1(i), \dots, \sigma_{|J|}(i)) \tag{10}$$

where the weights are defined as

$$C(i_1, \dots, i_{|J|}) = \sum_{k,\ell\neq k\in J} \rho_k \, \rho_\ell \, \langle Y^k_{i_k}, Y^\ell_{i_\ell} \rangle \ , \tag{11}$$

and then **averaging the assignments**

$$X_i^\star = \sum_{j\in J} \rho_j \, Y^j_{\sigma_j^\star(i)} \ \forall i \in I \, . \tag{12}$$

Computing the Wasserstein Barycenter for $d = 1$

In the **1-D case**, with points clouds, the Wasserstein barycenter can be computed again in $O(N\log(N))$ operations using each permutation $\sigma_j^\star$ that orders the set of values $Y^j \subset \mathbb{R}, \forall\, j \in J$.

The **Wasserstein barycenter** then reads

$$\forall\, i \in I, \quad \left(\mathrm{Bar}(\rho_j, Y^j)_{j\in J}\right)_i = \sum_{j\in J} \rho_j Y^j_{\sigma_j^\star(i)}\,. \tag{13}$$

Computing the Wasserstein Barycenter in general case ($d > 1$)

In the **multi-dimensional case**, the **relaxed** problem can be cast as a **linear program**

$$\max_{P \in \mathcal{P}_N^{|J|}} \sum_{(i_1,\ldots,i_{|J|}) \in I^{|J|}} P_{i_1,\ldots,i_j} C(i_1,\ldots,i_{|J|}) \,, \qquad (14)$$

where $\mathcal{P}_N^{|J|} \subset \mathbb{R}^{N \times \ldots \times N}$ is a multi-dimensional stochastic matrix and C is the cost matrix defined in Eq. (11).

Note: Now, the matrix P has $N^{|J|}$ elements.

Computing the Wasserstein Barycenter in general case ($d > 1$)

In the **multi-dimensional case**, the **relaxed** problem can be cast as a **linear program**

$$\max_{P \in \mathcal{P}_N^{|J|}} \sum_{(i_1,\dots,i_{|J|}) \in I^{|J|}} P_{i_1,\dots,i_j} C(i_1,\dots,i_{|J|}) \ , \qquad (14)$$

where $\mathcal{P}_N^{|J|} \subset \mathbb{R}^{N \times \dots \times N}$ is a multi-dimensional stochastic matrix and C is the cost matrix defined in Eq. (11).

Note: Now, the matrix P has $N^{|J|}$ elements.

Limitations

- **Intractable** for signal processing applications where $N \gg 10^3$
- Solution of (14) **is not a point cloud** anymore !

Remark: The optimal multi-assignment is a **NP-hard** problem [Burkard et al., 2009] . . .

Sliced Wasserstein Barycenter definition

Using the Sliced Wasserstein energy SW_2, we define the **Sliced Wasserstein Barycenter** of several point clouds $\{Y^j\}_{j\in J}$ for a given set of orientations Ψ as the **point cloud**

$$S\mathrm{Bar}(\rho_j, Y^j)_{j\in J} \in \arg\min_X \sum_{j\in J} \rho_j \,\mathrm{SW}_2(X, Y^j)^2 \,. \tag{15}$$

Sliced Wasserstein Barycenter gradient descent

A similar gradient descent algorithm can be defined for Sliced Wasserstein Barycenter.

- **Initialization:** $X^{(0)} := Y^q$, where $q = \text{argmax}_{j\in J}\rho_j$. Define a set Ψ of chosen orientations on $\mathbb{S}^{d-1}$.
- **Iteration:**
  - ▷ **Step 1:** For each $\theta \in \Psi$ and $j \in J$ compute the minimizer $\sigma^\star_{j,\theta}$ of

$$E_\theta(X^{(k)}, Y^j) = \min_{\sigma\in\Sigma_N} \sum_{i\in I} \langle X_i^{(k)} - Y^j_{\sigma(i)}, \theta\rangle^2 \,;$$

  - ▷ **Step 2:** For a given gradient step parameter $\lambda \leqslant 1$, $\forall\, i \in I$

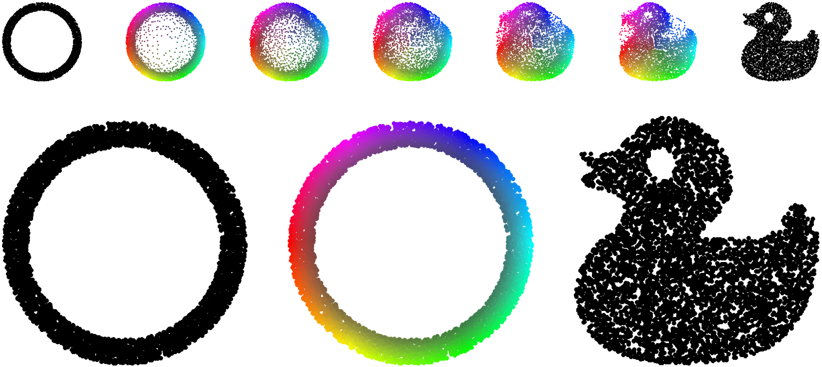
$$X_i^{(k+1)} \quad = \quad X_i^{(k)} - \lambda \cdot H_\Psi^\dagger(X_i^{(k)}) \times \sum_{\theta\in\Psi}\sum_{j\in J} \rho_j \left(\nabla E_\theta(X^{(k)}, Y^j)\right)_i \;,$$

where $H_\Psi^\dagger$ is the pseudo-inverse Hessian matrix.

- **Output:** The **Sliced Wasserstein Barycenter** of $\{Y^j\}_j$ is defined as $S\text{Bar}(\rho_j, Y^j)_{j\in J} := X^{(\infty)}$.

Example with $|J = 2|$

Interpolation of 2 distributions.

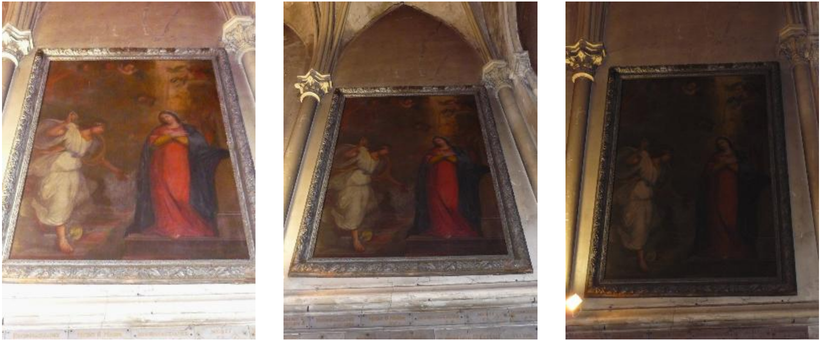


Example with $|J = 3|$

Color transfer

Color harmonization of several images

▷ **Step 1:** compute Sliced-Wasserstein Barycenter of color statistics;
▷ **Step 2:** compute Sliced-Wasserstein projection of each image onto the Barycenter;



Raw image sequence

Color transfer

Color harmonization of several images

▷ **Step 1:** compute Sliced-Wasserstein Barycenter of color statistics;
▷ **Step 2:** compute Sliced-Wasserstein projection of each image onto the Barycenter;

Sliced Wasserstein Projection *on the barycenter*

Color transfer

Color harmonization of several images

▷ **Step 1:** compute Sliced-Wasserstein Barycenter of color statistics;
▷ **Step 2:** compute Sliced-Wasserstein projection of each image onto the Barycenter;
▷ **Step 3:** Sliced-Wasserstein regularization.

Combination with the **Sliced Wasserstein regularization** *framework*

Extension of Heeger and Bergen algorithm for texture mixing

Idea:

use the Sliced Wasserstein Barycenter to **mix color textures**
within the Heeger&Bergen texture synthesis framework.


Original f^1


$\rho = 0.1$


$\rho = 0.2$


$\rho = 0.3$


$\rho = 0.4$


$\rho = 0.5$


$\rho = 0.6$

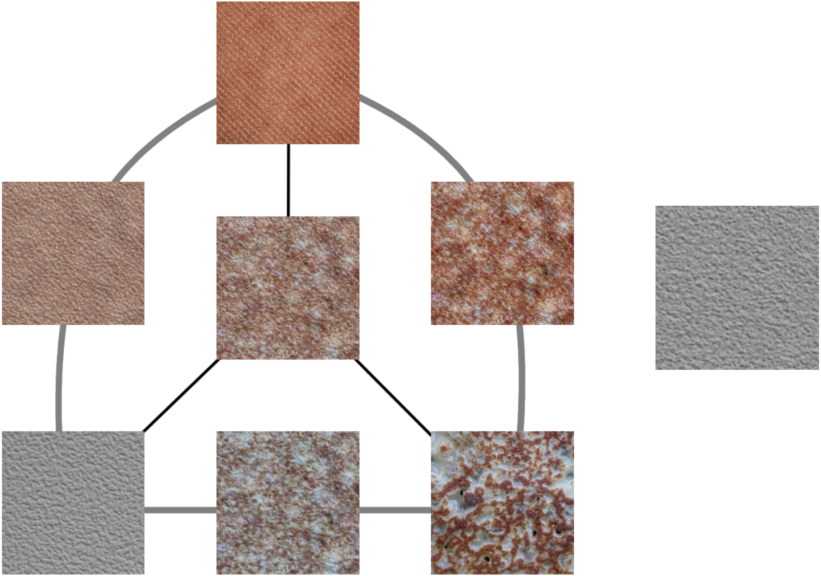

$\rho = 0.7$

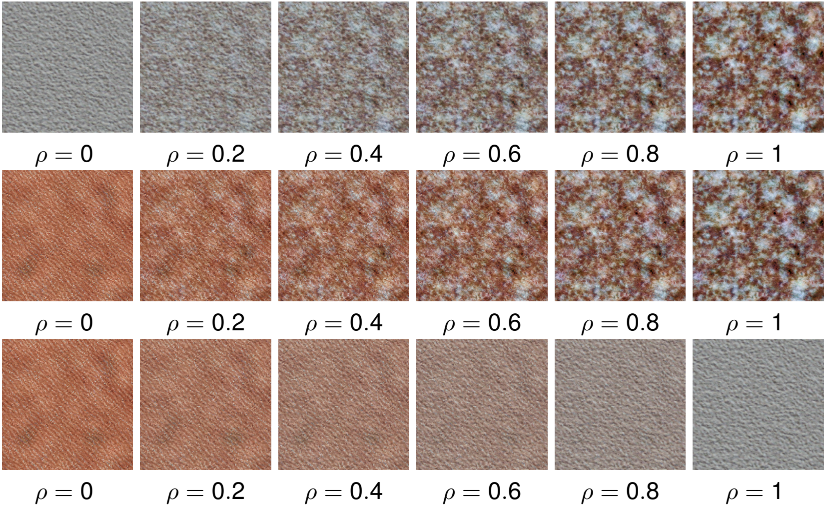

$\rho = 0.8$

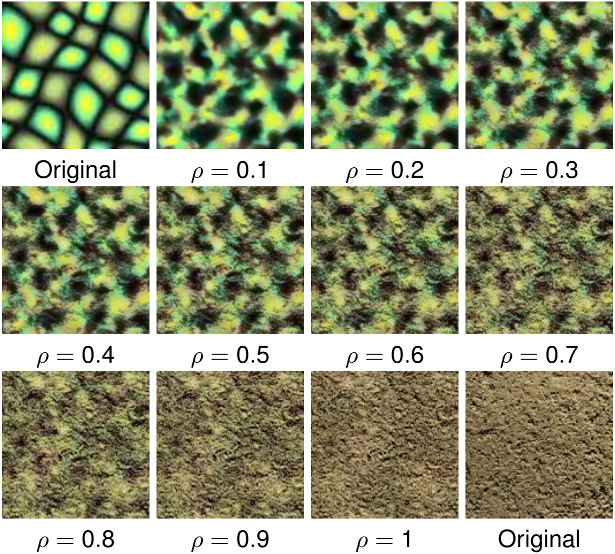

$\rho = 0.9$


$\rho = 1$

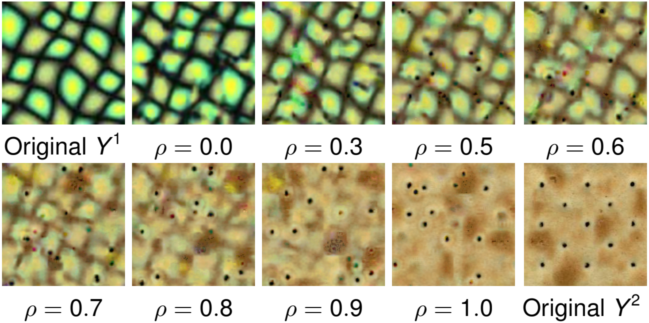

Original f^2



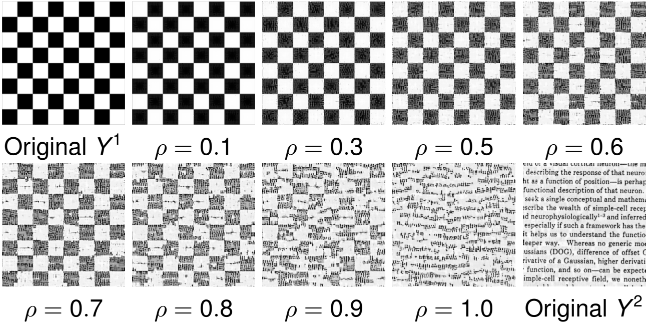

$\rho = 0$
$\rho = 0.2$
$\rho = 0.4$
$\rho = 0.6$
$\rho = 0.8$
$\rho = 1$
$\rho = 0$
$\rho = 0.2$
$\rho = 0.4$
$\rho = 0.6$
$\rho = 0.8$
$\rho = 1$
$\rho = 0$
$\rho = 0.2$
$\rho = 0.4$
$\rho = 0.6$
$\rho = 0.8$
$\rho = 1$


Original
$\rho = 0.1$
$\rho = 0.2$
$\rho = 0.3$
$\rho = 0.4$
$\rho = 0.5$
$\rho = 0.6$
$\rho = 0.7$
$\rho = 0.8$
$\rho = 0.9$
$\rho = 1$
Original

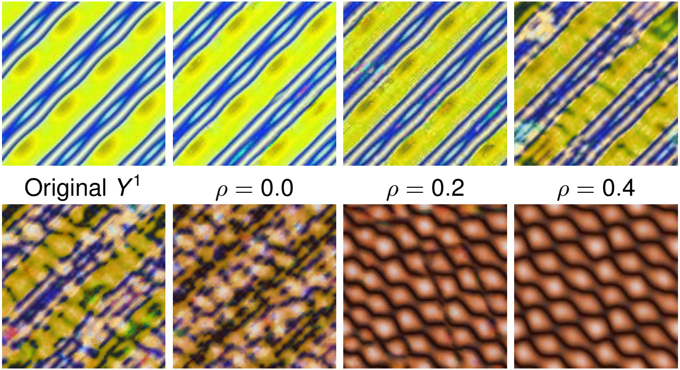
Examples of Texture mixing with extended Heeger and Bergen algorithm



Examples of Texture mixing with extended Heeger and Bergen algorithm



Examples of Texture mixing with extended Heeger and Bergen algorithm



Original Y^1	$\rho = 0.0$	$\rho = 0.2$	$\rho = 0.4$
$\rho = 0.6$	$\rho = 0.8$	$\rho = 1.0$	Original Y^2

Conclusion

- A fast algorithm to **approximate optimal transport** between **several** point clouds;
- A new and **generic** variational framework for regularization under statistical constraints.

Future works:

- Extension to other ground cost functions and other statistics;
- Use data structure to speed-up the algorithm;
- Some artifacts are not removed (diffusion):
  - Use more appropriate fidelity and regularity terms ? Example
  - Define a penalty term depending on the **transport plan regularity** ? Example

Question time

Thank you for your attention !

Bibliography

Agueh, M. and Carlier, G. (2011).
Barycenters in the Wasserstein space.
To appear in SIAM Journal on Mathematical Analysis.

Ballester, C., Caselles, V., Igual, L., Verdera, J., and Rougé, B. (2006).
A variational model for p+xs image fusion.
Int. J. Comput. Vision, 69(1):43–58.

Bottou, L. (1998).
Online algorithms and stochastic approximations.
In Saad, D., editor, *Online Learning and Neural Networks*. Cambridge University Press, Cambridge, UK.

Burkard, R., Dell'Amico, M., and Martello, S. (2009).
Assignment Problems.
SIAM.

Delon, J. (2004).
Midway image equalization.
J. Math. Imaging Vis., 21(2):119–134.

Gangbo, W. and Święch, A. (1998).
Optimal maps for the multidimensional monge-kantorovich problem.
Commun. on Pure and Appl. Math., 51:23–45.

Grauman, K. and Darrell, T. (2004).
Fast contour matching using approximate earth mover's distance.
In *Proc. of IEEE CVPR'04*, pages I: 220–227.

Grauman, K. and Darrell, T. (2005).
Efficient image matching with distributions of local invariant features.
In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 627–634, Washington, DC, USA. IEEE Computer Society.

Indyk, P. and Thaper, N. (2003).
Fast image retrieval via embeddings.
In *3rd International Workshop on Statistical and Computational Theories of Vision*, Nice, France.

Ion, A., Peyré, G., Haxhimusa, Y., Peltier, S., Kropatsch, W., and Cohen, L. (2007).
Shape matching using the geodesic eccentricity transform - a study.
In *Proc. Workshop of the Austrian Association for Pattern Recognition*, pages 97–104. OCG.

Ling, H. and Okada, K. (2007).
An Efficient Earth Mover's Distance Algorithm for Robust Histogram Comparison.
IEEE Trans. PAMI, 29(5):840–853.

Papadakis, N., Provenzi, E., and Caselles, V. (2010).
A variational model for histogram transfer of color images.
IEEE Transaction on Image Processing, 19(12).

Pele, O. and Werman, M. (2008).
A Linear Time Histogram Metric for Improved SIFT Matching.
In *ECCV08*.

Pitié, F., Kokaram, A., and Dahyot, R. (2007).

Automated colour grading using colour distribution transfer.
Computer Vision and Image Understanding.

Portilla, J. and Simoncelli, E. P. (2000).
A parametric texture model based on joint statistics of complex wavelet coefficients.
Int. Journal of Computer Vision, 40(1):49–70.

Rabin, J., Delon, J., and Gousseau, Y. (2011).
Removing artefacts from color and contrast modification.
IEEE Trans. Image Proc.
http://hal.archives-ouvertes.fr/hal-00505966/en/.

Rubner, Y., Tomasi, C., and Guibas, L. J. (2000).
The earth mover's distance as a metric for image retrieval.
International Journal of Computer Vision, 40(2):99–121.

Rudin, L., Osher, S., and Fatemi, E. (1992).
Nonlinear total variation based noise removal algorithms.
60:259–268.

Shirdhonkar, S. and Jacobs, D. (2008).
Approximate Earth Mover's Distance in linear time.
In *Computer Vision and Pattern Recognition (CVPR)*, pages 1–8.

Villani, C. (2008).
Optimal Transport. old and new.
Springer-Verlag.
Grundlehren Der Mathematischen Wissenschaften.

Appendix

Stochastic gradient descent for Sliced Wasserstein projection

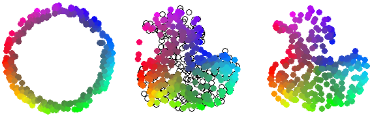
Alternative method: a **Stochastic gradient descent** scheme can be implemented in a such way that [Bottou, 1998]:

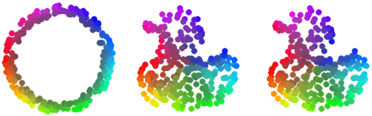
- The set of orientations $\Psi^{(k)}$ used at each iteration k is **random**.
- The gradient steps $\{\lambda_k \leqslant 1\}_k$ are s.t. $\sum_k \lambda_k = \infty$ and $\sum_k \lambda_k^2 < \infty$ (*e.g* $\lambda_k = \frac{1}{k}$) (optimal under some hypotheses which are not verified here).

Convergence: There is no proof of convergence in such settings, nevertheless we have always observed that $X^{(k)}$ converges to a local minimum which is a **(non-optimal) permutation** of the distribution Y.

Results with stochastic gradient descent

Projection results with $|\Psi| = 10d$, respectively **with** and **without** fixed direction set $|\Psi|$





1

1

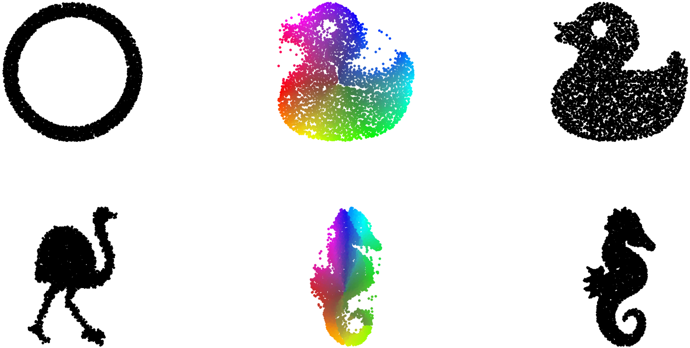
Back

Results with regularized Slice-Wasserstein projection

Given two point clouds X and $Y \in \mathbb{R}^{d \times N}$, compute a **regularized Sliced - Wasserstein Projection** $W^\star \in \mathbb{R}^{d \times N}$ of X onto Y

$$W^\star \in \min_{W \in \mathbb{R}^{d \times N}} \|W - X\|^2 + \lambda \, \|\nabla \mathbf{T}\|^2 + \mu \, \mathrm{SW}_2(W, Y)$$

where $T = Y - W$ is the approximate transport plan



Results with $\mu = 10^3$ and $\frac{\lambda}{\mu} \in \{0, .1, .2, .5, 1, 2, 5, 10, 20, 50, 100, 200, 500\}$ Back

Back to part II Back to conclusion

Back to part II Back to conclusion

Back

Back

Back

Back

Back

Back