Convolutional Wasserstein Distances Report

Nurlanov Zhakshylyk

Informatics - Technische Universität München

Abstract

Convolutional Wasserstein Distance (CWD) is the approximation of Wasserstein Distance – the measure between probability distributions taking into account the in-domain distance. The method was proposed by Justin Solomon, Fernando de Goes, Gabriel Peyre, Marco Cuturi, Adrian Butscher, Andy Nguyen, Tao Du, Leonidas Guibas in 2015. Unlike traditional methods for optimal transportation, such as Linear Programming, the resulting method CWD can be applied on large domains used in graphics, such as images and triangle meshes, improving performance by orders of magnitude. For this, the authors used entropic regularization of the optimal transportation problem and the heat kernel approximation of the kernel-based geodesic distances. The generalisability and efficiency of the proposed approach were shown on tasks including optimization over distances, such as Shape Interpolation, Wasserstein Barycenters, and Propagation.

1 Introduction

The probability distributions are widely used in Computer Graphics for different tasks. For example, manipulating images by histogram matching, relaxation of correspondence maps, shapes interpolation, and more. Whenever some geometrical features can be described by a non-negative integrable function defined on the geometrical domain, the necessity for the ability to manipulate distributions arises.

For this aim, we will introduce the Wasserstein Distance and discuss its advantages compared to other ways of measuring the distance between distributions in the Introductory part. Then in the Main part we will present the efficient approach of computing approximated Wasserstein Distance, called Convolutional Wasserstein Distance, by using entropy regularization of optimal transportation problem, and approximating kernel-based geodesic distances by convolution against heat kernel. The resulting method leads to simple iterative numerical schemes with linear convergence, in which each iteration only requires Gaussian convolution or the solution of a sparse, pre-factored linear system, i.e. the heat equation. In this part, we will also describe the ways of solving problems involving optimization over Wasserstein Distances, and show the commonality of algorithms. In Experimental part we will compare timings of our final solution with baseline approach (Linear Programming), and the previous solution [2], which is the intermediate step of our method. Also, there are applications of the resulted method presented in this section. And at the end, we will conclude the work, and discuss the drawbacks of the proposed solution.

1.1 Wasserstein distance

The main tool for manipulating discrete probability distributions over the geometric domain is the so-called Earth Mover's Distance (EMD).

Def 1. *Earth Mover's Distance (EMD):*

$$\begin{aligned} EMD(p,q) = & \min_{T\geq 0} \sum_{i,j} T_{i,j} d(x_i, x_j), \\ & s.t. \sum_{j} T_{i,j} = p_i, \quad \sum_{i} T_{i,j} = q_j \end{aligned} \tag{1}$$

Which measures the minimum amount of "work" to be done for matching source distribution p with target distribution q. The "work" here is the mass $T_{i,j}$ taken from the source distribution at point x_i and moved to point x_j to finally get the target distribution q multiplied by the geodesic distance d between points x_i and x_j, and summed over all possible pairs of points. The joint distribution T, constrained by source and target, is called Transportation Plan. And the problem of minimizing such a functional is called the Optimal Transportation problem. The geodesic distance $d(\cdot,\cdot)$ is the shortest path between points over the geometric domain M, where the points and this distance itself are defined.

A continuous and more general formulation of the above problem is the so-called Wasserstein Distance.

Def 2. *p-Wasserstein Distance:*

$$\begin{gathered} W_p = \inf_{T\in\Pi}\left[\iint\limits_{M\times M} T(x,y)d(x,y)^p dxdy\right]^{1/p} \\ \Pi = \Pi(\mu_0,\mu_1) := \left\{ f \in Prob(M\times M)\ \mid \int\limits_M f(x,y)dy = \mu_0(x), \int\limits_M f(x,y)dx = \mu_1(y)\right\} \end{gathered} \tag{2}$$

In this work, we will analyze the specific case, 2-Wasserstein Distance, and everywhere if it is not stated explicitly, we will understand this particular case under Wasserstein Distance.

Def 3. *2-Wasserstein Distance := Wasserstein Distance:*

$$\begin{gathered} W_2 = \inf_{T\in\Pi}\left[\iint\limits_{M\times M} T(x,y)d(x,y)^2 dxdy\right]^{1/2} \\ \Pi = \Pi(\mu_0,\mu_1) := \left\{ f \in Prob(M\times M)\ \mid \int\limits_M f(x,y)dy = \mu_0(x), \int\limits_M f(x,y)dx = \mu_1(y)\right\} \end{gathered} \tag{3}$$

1.2 Comparison with other distances

We would like to explicitly point out the advantages of the Wasserstein Distance compared to other well-known ways of measuring the distance between probability distributions. It turns out that in the general case of non-overlapping domains, i.e. areas where a function is non-zero, the Wasserstein Distance gives geometrically feasible results, while other distances are either not computable or geometrically infeasible, i.e. they do not depend on geometric changes.

For comparison, we have chosen the following distance functions: L_p-norm, Kullback-Leibler (KL), and Jensen-Shannon (JS) divergences.

Def 4. *L_p-norm of function from L_p Lebesque space M:*

$$L_p(f) = \int_M ||f(x)||_p dx \tag{4}$$

In our case of two functions μ_0 and μ_1, we can measerue the distance between them as the norm of their pointwise difference:

$$L_p(\mu_0,\mu_1) = \int_M ||\mu_0(x)-\mu_1(x)||_p dx \tag{5}$$

Def 5. *KL divergence between probability distributions μ_0 and μ_1:*

$$D_{KL}(\mu_0 \parallel \mu_1) = \int_M \mu_0(x) log\frac{\mu_0(x)}{\mu_1(x)}dx \tag{6}$$

Def 6. *JS divergence between probability distributions μ_0 and μ_1:*

$$D_{JS}(\mu_0 \parallel \mu_1) = \frac{1}{2}D_{KL}(\mu_0 \parallel \frac{\mu_0+\mu_1}{2}) + \frac{1}{2}D_{KL}(\mu_1 \parallel \frac{\mu_0+\mu_1}{2}) \tag{7}$$

Let us consider the following problem: there are two 2-dimensional distributions $\mu_0(x,y)$ and $\mu_1(x,y)$, such that the first is located at $x=0$ on the x-axis, $\mu_0(x=0,y)=\mu_0(\cdot,y)=U([0,1])$, i.e. $\mu_0(x\neq 0,y)=0,\forall y$, and the second disctribution is located at point $x=\theta$ on x-axis, $\mu_1(x=\theta,y)=\mu_1(\cdot,y)=U([0,1])$, i.e. $\mu_1(x\neq\theta,y)=0,\forall y$.

Now we can compute the all defined above distances between these two distributions. In the first case, let's assume that $\theta\neq 0$, i.e. two functions with non-overlapping domains. Then

$$\begin{aligned} L_1(\mu_0,\mu_1) &= \int ||\mu_0(x,y)-\mu_1(x,y)||dxdy = \int ||\mu_0(0,y)||dy + \int ||\mu_1(\theta,y)||dy = 2 \\ D_{KL}(\mu_0 \parallel \mu_1) &= +\infty = D_{KL}(\mu_1 \parallel \mu_0) \\ D_{JS}(\mu_0 \parallel \mu_1) &= \log 2 \\ W_2^2(\mu_0,\mu_1) &= \inf_{T\in\Pi}\left[\int T(x_0,x_1)d(x_0,x_1)^2dx_0dx_1\right] = \left[\int U(y)\theta^2dy\right] = \theta^2 \end{aligned} \tag{8}$$

But in case $\theta=0$, when the domains of the functions are the same, we have:

$$L_1(\mu_0,\mu_1) = D_{KL}(\mu_0 \parallel \mu_1) = D_{JS}(\mu_0 \parallel \mu_1) = W_2^2(\mu_0,\mu_1) = 0 \tag{9}$$

So it means that only Wasserstein distance provides smooth and geometrically coherent results in both cases.

Another example of the advantage of Wasserstein Distance can be shown at a simple averaging distributions problem. Let us consider two delta Dirac functions, centered at x and $y\in\mathbb{R}$: $\delta(x)$, $\delta(y)$. The euclidean average $\frac{\delta(x)+\delta(y)}{2}$ is a bi-modal function with peaks at both x and y. However, the result of Wasserstein averaging $\mu^* = \arg\min_{\mu}\left\{\frac{1}{2}W_2(\mu,\delta(x))+\frac{1}{2}W_2(\mu,\delta(y))\right\}$ is equal to $\delta(\frac{x+y}{2})$ – delta Dirac centered at midpoint.

2 Main approach

2.1 Entropy regularization

To solve the transportation problem 3, we propose first to solve the regularized problem, and then leading the regularization term close to zero we assume to get the approximate solution of the original problem. As a regularization term, we use the entropy function, which shows the level of uncertainty of the distribution.

Def 7. *Entropy $H(T)$:*

$$H(T) = -\iint\limits_{M\times M} T(x,y)lnT(x,y)dxdy \tag{10}$$

The entropy regularization gives not only understandable intuition, making the solution less strict, but also a new mathematical formulation of the problem.

Def 8. *Regularized Wasserstein Distance $W_{2,\gamma}$:*

$$\begin{aligned} W_{2,\gamma}^2(\mu_0,\mu_1) &:= \inf_{T\in\Pi}\left[\iint\limits_{M\times M} T(x,y)d(x,y)^2dxdy - \gamma H(T)\right] = \\ &= \inf_{T\in\Pi}\left[\iint\limits_{M\times M} T(x,y)d(x,y)^2dxdy + \gamma\iint\limits_{M\times M} T(x,y)lnT(x,y)dxdy\right] \end{aligned} \tag{11}$$

Here, it is useful to define distance-based kernel function, parametrized by the same parameter γ, for further simplification of the problem.

Def 9. *Distance-based kernel function $\mathcal{K}_\gamma$:*

$$\mathcal{K}_\gamma(x,y) := e^{-\frac{d(x,y)^2}{\gamma}} \tag{12}$$

And now replacing the squared distance by the kernel, i.e. $d(x,y)^2 = -\gamma \ln \mathcal{K}_\gamma(x,y)$, we have:

$$\begin{aligned} W_{2,\gamma}^2(\mu_0,\mu_1) :=& \gamma \inf_{T\in\Pi} \iint\limits_{M\times M} T(x,y) \ln \frac{T(x,y)}{\mathcal{K}_\gamma(x,y)} dxdy = \\ =& \gamma \left(\inf_{T\in\Pi} D_{KL}(T \parallel \mathcal{K}_\gamma) + const \right) \end{aligned} \tag{13}$$

It turns out that the new regularized problem has a nice interpretation and useful properties induced by properties of KL-divergence. Firstly, now the optimal transportation problem can be interpreted as a projection of kernel $\mathcal{K}_\gamma$ with respect to KL-divergence on space Π of joint distributions constrained by source and target distributions. Moreover, as KL-divergence is a convex function in both arguments, and it is strictly convex in the first argument with a fixed second one, we derived the strictly convex problem with convex space of constraints.

2.2 Heat kernel approximation

There is still a need in computing all pairwise geodesic distances $d(\cdot,\cdot)$ to construct a kernel $\mathcal{K}_\gamma$. In an arbitrary domain, the naive precomputing takes $O(n^2)$ space and time, where n- is the number of points. It is notable that for solving the regularized problem 13 it is necessary to be able to compute the convolution of arbitrary function f against kernel $\mathcal{K}_\gamma$. With this aim, we propose to approximate the distance-based kernel $\mathcal{K}_\gamma$ with the Heat kernel $\mathcal{H}_t$ as in [1].

Def 10. *Heat Equation:*

$$\begin{cases} \frac{\partial f(t,x)}{\partial t} = \Delta f(t,x), x \in M \\ f(0,x) = g(x), x \in M \end{cases} \tag{14}$$

We know that the solution of heat equation 10, where the Laplacian operator is defined on the domain M, is of the form:

Def 11. *Solution of the Heat Equation 10:*

$$f(t,x) = \int\limits_M g(y)\mathcal{H}_t(x,y)dy \tag{15}$$

So here we defined the Heat kernel $\mathcal{H}_t$ utilizing the convolution of an arbitrary initial function g against it.

Physically the Heat kernel $\mathcal{H}_t(x,y)$ determines the diffusion between points $x, y \in M$ after time t. We can intuitively derive the relationship between heat and distance. The heat diffusion can be modeled as a large set of hot particles randomly walking with starting point x. Any particle that reaches a distant point y after a small time t has had little time to deviate from the shortest path. With this idea Varadhan in 1967 [5] stated that the geodesic distance $d(\cdot,\cdot)$ between any pair of points x, y on a Riemannian manifold M can be recovered via a simple pointwise transformation of the heat kernel:

Def 12. *Varadhan's formula:*

$$d(x,y)^2 = \lim_{t\to 0} \left[-2t \cdot ln\mathcal{H}_t(x,y)\right] \tag{16}$$

Setting $t := \frac{\gamma}{2}$, we approximate distance-based kernel by the Heat kernel:

$$\mathcal{K}_\gamma(x,y) \approx \mathcal{H}_{\frac{\gamma}{2}}(x,y) \tag{17}$$

The advantage of such an approximation is that we can now compute the convolution of an arbitrary function f against kernel $\mathcal{K}_\gamma$ by solving the heat equation for a time $t = \gamma/2$ with f as an initial condition. After inserting a heat kernel approximation, we derive a diffusion approximation $W_{2,\mathcal{H}}$ of regularized $W_{2,\gamma}$ Wasserstein Distance W_2.

Def 13. *Convolutional Wasserstein Distance:*

$$W^2_{2,\mathcal{H}_{\frac{\gamma}{2}}}(\mu_0,\mu_1) := \gamma \inf_{T\in\Pi} KL\left(T \parallel \mathcal{H}_{\frac{\gamma}{2}}\right)$$
$$\Pi = \Pi(\mu_0,\mu_1) := \left\{ f \in Prob(M\times M) \mid \int\limits_M f(x,y)dy = \mu_0(x), \int\limits_M f(x,y)dx = \mu_1(y) \right\} \tag{18}$$

2.3 Discrete setting

Now it is time to go into details of numerical implementation of the proposed approach for discretized domains.

- Domain M is descretized into n parts;
- Functions on domain M are represented as vectors $\vec{f} \in \mathbb{R}^n$;
- The integration over domain M is computed with means of "area vector" $\vec{a} \in \mathbb{R}^n_+$: $\int\limits_M f(x)dx \approx \vec{a}^T\vec{f}$, such that $\vec{a}^T\vec{1} = 1$
- Distributions $\mu \in Prob(M)$ are represented as unit vectors: $\vec{\mu} \in \mathbb{R}^n_+$, such that $\vec{\mu}^T\vec{a} = 1$
- Joint distributions $T \in Prob(M\times M)$ are represented as positive matrices $T \in \mathbb{R}^{n\times n}_+$, normed to unit length: $\vec{a}^T T\vec{a} = 1$.
- The Heat kernel $\mathcal{H}_t$ is represented as a symmetric matrix $H_t \in \mathbb{R}^{n\times n}_+$
- The convolution against Heat kernel is represented by means of "area vector" $\vec{a}$ and elementwise multiplication $\otimes$ (elemetwise division is $\oslash$):

$$\int\limits_M f(y)\mathcal{H}_t(\cdot,y)dy \approx H_t(\vec{a}\otimes f)$$

- The Laplacian operator Δ defined on a domain M is a linear operator, that is why in case of the discrete domain it becomes a matrix $L_M \in \mathbb{R}^{n\times n}$. In the case of the mesh representation of the surface M, respective Laplacian L_M becomes the cotangent Laplacian, and area vector $\vec{a}$ is proportional to the sum of triangle areas adjacent to a given vertex.

So in descrete setting, the Heat equation with forward parametrization is of the form:

$$\begin{cases} \vec{f}_{t+1} - \vec{f}_t = L\vec{f}_t \\ \vec{f}_0 = \vec{g} \end{cases} \tag{19}$$

Where its solution written in descrete terms is the following:

$$\vec{f}_t = H_t(\vec{a}\otimes\vec{g}) \tag{20}$$

For this moment, it becomes clear that convolution against Heat kernel is the same as finding the solution of the Heat equation with respective initial function. And in a discrete setting in most of the cases, the matrix in the Heat equation is sparse and symmetric, so it can be pre-factorized and efficiently solved.

At the end of this subsection, let us write the formulation of the diffusion approximation of Wasserstein distance in a discrete setting.

Def 14. *Discrete formulation of Convolutional Wasserstein Distance:*

$$W^2_{2,H_{\frac{\gamma}{2}}}(\vec{\mu_0},\vec{\mu_1}) = \gamma \inf_{T\in\Pi} KL\left(T \parallel H_{\frac{\gamma}{2}}\right) = \gamma \inf_{T\in\Pi} \sum_{i,j} T_{i,j}a_i a_j \ln\frac{T_{i,j}}{H_{i,j}}$$
$$\Pi = \Pi(\vec{\mu_0},\vec{\mu_1}) := \left\{ T \in \mathbb{R}^{n\times n}_+ \ |T\vec{a} = \vec{\mu_0}, T^T\vec{a} = \vec{\mu_1} \right\} \tag{21}$$

2.4 Alternating projections algorithm

The optimization problem 21 is strictly convex with linear constraints on Π. But the complexity of the problem is tied in quadratic number of unknowns in matrix T. To overcome this issue there has been proved a following proposition:

Proposition 1. *The transportation plan $T \in \Pi(\vec{\mu}_0, \vec{\mu}_1)$ minimizing 21 is of the form*

$$T = D_{\vec{v}} H_t D_{\vec{w}} \tag{22}$$

with unique vectors $\vec{v}, \vec{w} \in \mathbb{R}^n$ satisfying

$$\begin{cases} D_{\vec{v}} H_t D_{\vec{w}} \vec{a} = \vec{\mu}_0, \\ D_{\vec{w}} H_t D_{\vec{v}} \vec{a} = \vec{\mu}_1 \end{cases} \tag{23}$$

Here $D_{\vec{v}}$ denotes the diagonal matrix with vector $\vec{v}$ in the diagonal. As a consequence of this proposition 1, we have now linear number of unknowns. Moreover, the unknown unique vectors $\vec{v}, \vec{w}$ can be found by alternating projections onto linear constraints 21. This approach gives rise to an area-weighted version of *Sinkhorn's algorithm* [3] with linear convergence rate.

Algorithm 1. *Alternating Projections for Convolutional Wasserstein Distance:*

```
• Input: $\vec{\mu}_0, \vec{\mu}_1; H_t, \vec{a}$
• Initialize: $\vec{v} = \vec{w} = \vec{1}$
• Do num_iter iterations:
    – $\vec{v} = \vec{\mu}_0 \oslash H_t(\vec{a} \otimes \vec{w})$
    – $\vec{w} = \vec{\mu}_1 \oslash H_t(\vec{a} \otimes \vec{v})$
• Output: $\gamma \vec{a}^T \left[ (\vec{\mu}_0 \otimes \ln \vec{v}) + (\vec{\mu}_1 \otimes \ln \vec{w}) \right]$
```

2.5 Optimization over Wasserstein Distance

In practice, it is more useful not to evaluate the Wasserstein Distance, but to minimize functionals constructed out of them. An advantage of the proposed method is in the common structure of algorithms for all of these problems. Let us show this fact for the problem of computing Wasserstein Barycenters, which is a generalized form of weighted interpolation between N distributions.

Def 15. *Wasserstein Barycenter:*

$$\mu^* = arg \min_{\mu} \sum_{i=1}^{N} \alpha_i W_2^2(\mu, \mu_i) \tag{24}$$

Rewriting the above problem 24 in descrete terms of optimal transportation, and defining $\vec{\mu}^*$ as $\vec{\mu}^* = T_i \vec{a}, \forall i \in \{1, 2, \dots, n\}$, where T_i is the transportation plan between $\vec{\mu}^*$ and $\vec{\mu}_i$, we have:

Def 16. *Wasserstein Barycenter in transportation terms:*

$$\begin{aligned} &\min_{\{T_i\}} \sum_{i=1}^{N} \alpha_i KL(T_i \parallel H_t) \\ &s.t. \{T_i^T \vec{a} = \vec{\mu}_i, \forall i\} = C_1 \\ &\quad\;\; \{T_i \vec{a} = T_j \vec{a}, \forall i, j\} = C_2 \end{aligned} \tag{25}$$

The optimal transportation again can be viewed as a projection with respect to KL divergence from H_t (repeated N times) onto the constraint set $C_1 \cap C_2$. Problems of this form can be minimized using Bregman projection, which initializes all T_i to H_t and then cyclically projects the current iterate onto C_i at a time. Moreover, the projections onto C_1 and C_2 can be written in closed form.

As it has been shown, the optimization over Wasserstein Distance has the common structure, which can be written in general form:

Algorithm 2. *Common structure of algorithms for Optimization over Wasserstein Distance:*

- *Input:* $\vec{\mu}_i$, *for* $i \in V_0; H_t, \vec{a}$
- *Initialize all transportation plans as* H_t
- *Do num_iter iterations:*
  - *Projections onto* C_j *induced by constraints of source and target distributions*
    - *Convolutions against Heat kernel*
    - *Elementwise vector operations*
- *Output: final transportation plans or final distributions*

For detailed implementations, proofs of the propositions, and other formulations inducing optimal transportation go to the original paper [4].

3 Experiments

To evaluate efficiency, we compare three approaches approximating W_2: a linear program discretizing the original formulation 3, regularized distances with a full distance-based kernel $W^2_{2,\gamma}$ [2], and convolutional Wasserstein distances W^2_{2,H_t}.

3.1 Timing comparison

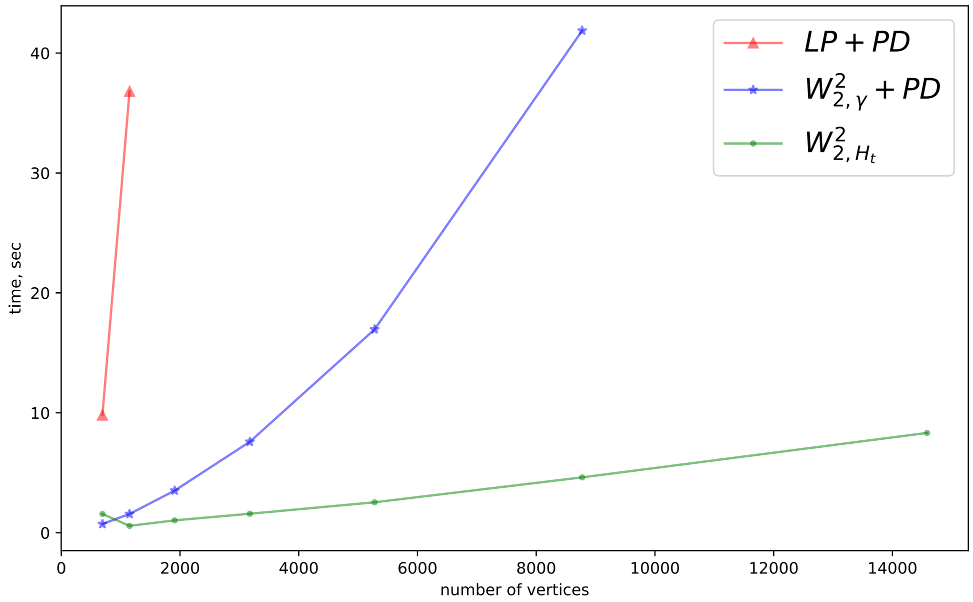

Figure 1: Timing (in sec.) for approximating W_2 between random distributions on triangle meshes, averaged over 10 trials. After one minute time out the result was not captured. Pairwise distance (PD) computation is needed for the linear program (LP) and $W_{2,\gamma}$. Cholesky pre-factorization (PF) is needed for convolutional distance W_{2,H_t}.

Graph 1 shows the results of the experiment on meshes of the same shape with varying density. Both regularized approximations of Wasserstein Distance outperform the linear program by a significant margin. The final approach outperforms the intermediate one notably on large meshes, for which the kernel computation takes a large amount of time and space.

3.2 Applications

There are a variety of applications of the proposed solution. The examples are shape interpolation, BRDF design, color histogram manipulation, skeleton layout, soft maps. Here, we will present the most demonstrative application, namely shapes interpolation. Other more impressive examples can be found in the original paper [4].

In order to compute an intermediate shape, we represent k shapes $(S_i)_{i=1}^k$ using normalized indicator functions μ_i. Given weights $(\alpha_i)_{i=1}^k$, we compute the distribution function of an averaged shape as:

$$\mu^* = arg\min_{\mu} \sum_{i=1}^{k} \alpha_i W^2_{2,H-t}(\mu, \mu_i)$$

The indicator function can easily be sharpened from this distribution if a true binary function is desired.

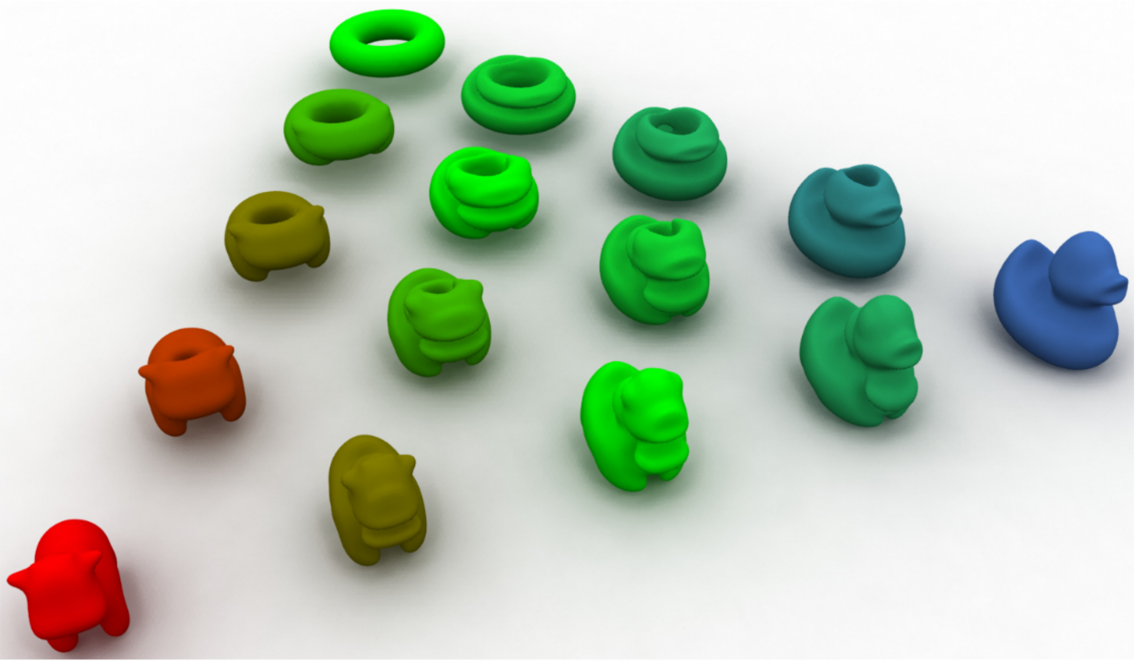

Figure 2: Shape interpolation in 3D between a cow, a duck, and a torus via convolutional Wasserstein barycenters on a 100 × 100 × 100 grid.

4 Conclusion

The proposed method of efficiently computing the Convolutional Wasserstein Distance allows us to make the optimal transportation a common tool in Computer Graphics. Moreover, the applications of the tool can appear in different fields where manipulations of the probability distributions over some domain are desired. Although the result of this work outperforms the previous approaches, it has some drawbacks. First of all, the approximated $W^2_{2,H_{\gamma/2}}$ is not a distance in common sense. It is never equal to exact zero, and it satisfies the triangle inequality approximately only for small γ. And it brings us to the second, but more important issue, $W^2_{2,H_{\gamma/2}}$ is unstable when γ is close to zero. The numerics degrade, and it is a true art to find the compromise between the sharpness of the result and the computational time, which highly increases when γ becomes less than the resolution of the discretized domain. The conditions for convergence of $W^2_{2,H_{\gamma/2}}$ as $\gamma \to 0$ were not developed in this work.

References

[1] Keenan Crane, Clarisse Weischedel, and Max Wardetzky. Geodesics in heat: A new approach to computing distance based on heat flow. *ACM Trans. Graph.*, 32(5), October 2013.

[2] Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26*, pages 2292–2300. Curran Associates, Inc., 2013.

[3] Richard Sinkhorn. A relationship between arbitrary positive matrices and doubly stochastic matrices. *Ann. Math. Statist.*, 35(2):876–879, 06 1964.

[4] Justin Solomon, Fernando de Goes, Gabriel Peyré, Marco Cuturi, Adrian Butscher, Andy Nguyen, Tao Du, and Leonidas Guibas. Convolutional wasserstein distances: Efficient optimal transportation on geometric domains. *ACM Trans. Graph.*, 34(4), July 2015.

[5] S. R. S. Varadhan. On the behavior of the fundamental solution of the heat equation with variable coefficients. *Communications on Pure and Applied Mathematics*, 20(2):431–455, 1967.