

# GradNetOT: Learning Optimal Transport Maps with GradNets

Shreyas Chaudhari\* Srinivasa Pranav\* José M. F. Moura

Electrical and Computer Engineering, Carnegie Mellon University

{shreyasc@andrew.cmu.edu, spranav@cmu.edu, moura@andrew.cmu.edu }

**Abstract**—Monotone gradient functions play a central role in solving the Monge formulation of the optimal transport (OT) problem, which arises in modern applications ranging from fluid dynamics to robot swarm control. When the transport cost is the squared Euclidean distance, Brenier’s theorem guarantees that the unique optimal transport map satisfies a Monge–Ampère equation and is the gradient of a convex function. In [1], [2], we proposed Monotone Gradient Networks (mGradNets), neural networks that directly parameterize the space of monotone gradient maps. In this work, we leverage mGradNets to directly learn the optimal transport mapping by minimizing a training loss function defined using the Monge–Ampère equation. We empirically show that the structural bias of mGradNets facilitates the learning of optimal transport maps across both image morphing tasks and high-dimensional OT problems.

**Index Terms**—Optimal Transport, Gradient Networks, Monge–Ampère, Brenier’s Theorem, Physics Informed Neural Networks (PINNs)

## I. INTRODUCTION

The optimal transport (OT) problem originated with the work of Gaspard Monge in 1781, who was interested in a fundamental problem that arises when constructing fortifications: how can one transfer molecules of soil from piles to holes while minimizing the cost of transfer [3]? Since then, Monge’s optimal transport problem and Leonid Kantorovich’s relaxation [4], [5] have been phrased in terms of probability densities and finding an optimal transport map: how can a source probability distribution be mapped to a target probability distribution in a way that minimizes the transport cost?

These formulations of optimal transport, along with several newer variations, have found applications spanning not only statistics and logistics, but also signal and image processing, robotics, machine learning, fluid dynamics, economics, geology, astronomy, and beyond. In our simulations, we illustrate an example of optimal transport for image morphing applications. Robotics applications of OT include the control of large swarms of robots or unmanned aerial vehicles (UAVs) [6], which have applications ranging from critical natural disaster rescue missions to intricate and artistic light shows. By modeling movements of large robot swarms as time-evolving spatial probability distributions, an optimal transport map can guide the swarm from an initial distribution to a desired target distribution while minimizing travel cost [7]. Furthermore, OT applications to economics include works showing the equivalence between optimal transport, stable matching, and modeling hedonic price equilibria, which capture relationships between prices and product characteristics [8]. In geology, developments concerning OT on Riemannian manifolds have been used to model continental drift by describing ground mass distributions and geodesic transport paths over time [9].

\* Authors contributed equally. Authors partially supported by NSF Graduate Research Fellowships (DGE-1745016, DGE-2140739), NSF Grant CCF-2327905, and an ARCS Fellowship. Authors thank Kawisorn Kamtue for helpful discussions. Code available at <https://github.com/cShreyas/GradNetOT>.

© 2025 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

In astrophysics, recent works on Monge–Ampère gravity employ optimal transport maps to model the distribution of mass throughout the universe and the formation of cosmological structures [10].

On the theory side, in 1991, Yann Brenier proved a key result that became a catalyst for optimal transport flourishing into an active research field in the modern era [11]. Brenier showed the equivalence between the Monge optimal transportation problem and the elliptic Monge–Ampère equation, leading to the following result: when the source and target probability distributions are well-defined and the transport cost is the squared Euclidean distance, then the optimal transport map is the gradient of a convex potential function that solves the corresponding Monge–Ampère equation.

Exploiting Brenier’s link between optimal transport and the Monge–Ampère equation is nontrivial because the Monge–Ampère equation is highly non-linear and generally difficult to solve using standard techniques like calculus of variations [12]. While some works, such as [10], rely on additional assumptions like spherical symmetry to recover an analytic solution, the majority of the literature uses numerical approaches like discrete-discrete entropic regularization and semi-discrete optimal transport [13]. In this paper, we focus on the intersection of Brenier’s result and advancements in neural networks to solve Monge optimal transport problems.

We propose GradNetOT: a method that leverages monotone gradient networks (mGradNets) [2] to solve Monge optimal transport problems. Using an mGradNet to parameterize the optimal transport map, we search through the space of all gradients of convex potential functions. To guide our search, we explicitly use the Monge–Ampère equation in our optimization objective.

## II. BACKGROUND

We briefly review probability concepts relevant to optimal transport (OT). A *probability measure*  $\mu$  assigns mass to events, which are measurable subsets of a sample space  $\mathcal{X}$ . The probability measure satisfies the Kolmogorov axioms of non-negativity, countable additivity, and normalization ( $\mu(\mathcal{X}) = 1$ ). The probability of an event  $A \subseteq \mathcal{X}$  is expressed as  $\mu(A) = \int_A d\mu(\mathbf{x})$ . For example, if  $\mu$  is discrete, it assigns probability mass to a finite set of points and  $\mu(A) = \sum_{x \in A} \mu(x)$ . Alternatively, if  $\mathcal{X} = \mathbb{R}^N$ , then the Lebesgue measure yields the  $n$ -dimensional volume of  $A \subset \mathbb{R}^N$ . If  $\mu$  is absolutely continuous with respect to the Lebesgue measure, then it admits a probability density function  $p(\mathbf{x})$  and  $\mu(A) = \int_A p(\mathbf{x}) dx$ .

Probability measures can be transformed by functions. Given a function  $f : \mathcal{X} \rightarrow \mathcal{Y}$ , where  $\mathcal{X}$  is equipped with probability measure  $\mu$ , the *pushforward measure*  $f_\# \mu$  is the resulting probability measure on  $\mathcal{Y}$  induced by  $f$ . For  $A \subseteq \mathcal{Y}$ , the pushforward is defined by:

$$f_\# \mu(A) = \mu(f^{-1}(A)) = \mu(\{\mathbf{x} \in \mathcal{X} : f(\mathbf{x}) \in A\}) \quad (1)$$

where  $f^{-1}(A)$  corresponds to the preimage of  $A$  under  $f$ . Intuitively, (1) computes the probability of set  $A$  in the target space  $\mathcal{Y}$  by measuring the probability of the set in the source space  $\mathcal{X}$  whose image under  $f$  is  $A$ . Additionally,  $f$  is called a diffeomorphism if  $f$  is a bijection and both  $f$  and  $f^{-1}$  are differentiable.

### III. PROBLEM STATEMENT

Optimal transport involves finding the most efficient way to transform one probability distribution into another. The distributions can be thought of as descriptions of how mass (representing probability) is spread over a space. Efficiency here is measured by a predefined cost function.

Formally, let  $\mathcal{X}$  and  $\mathcal{Y}$  be spaces equipped with respective probability measures  $\mu$  and  $\nu$ . Let  $c : \mathcal{X} \times \mathcal{Y} \rightarrow [0, \infty)$  be a cost function, where  $c(\mathbf{x}, \mathbf{y})$  denotes the cost of moving a unit of mass (probability) from  $\mathbf{x} \in \mathcal{X}$  to  $\mathbf{y} \in \mathcal{Y}$ . The optimal transport problem is then to find a map  $T : \mathcal{X} \rightarrow \mathcal{Y}$  that pushes forward  $\mu$  to  $\nu$  (i.e.,  $T_\# \mu = \nu$ ), and minimizes the total transport cost:

$$\inf_{T: T_\# \mu = \nu} \int_{\mathcal{X}} c(\mathbf{x}, T(\mathbf{x})) d\mu(\mathbf{x}) \quad (2)$$

The formulation in (2) is known as the Monge problem and applies to settings where the probability measures  $\mu$  and  $\nu$  are either discrete or continuous [12]. In general, establishing the uniqueness, and structure of the optimal transport map  $T$  that solves (2) is challenging [14]. In fact, a map satisfying  $T_\# \mu = \nu$  may not exist at all since the Monge problem does not permit mass splitting. For example, consider discrete measures where  $\mu$  assigns all its mass to a single point, while  $\nu$  distributes the mass evenly between two distinct points. Since  $T$  is a deterministic map sending each point  $\mathbf{x} \in \mathcal{X}$  to a single point  $\mathbf{y} \in \mathcal{Y}$ , no such  $T$  can push  $\mu$  forward to  $\nu$  [12].

The existence and uniqueness of the solution to the Monge problem in (2) can be established with additional assumptions on the probability measures and cost function [14]. In this work, we specify  $\mathcal{X} = \mathcal{Y} = \mathbb{R}^N$  and assume  $T$  is a diffeomorphism.<sup>1</sup> Brenier's theorem [11], [15] guarantees that if the probability measures  $\mu$  and  $\nu$  are absolutely continuous with respect to the Lebesgue measure, i.e., they admit respective probability densities  $p(\mathbf{x})$  and  $q(\mathbf{y})$ , and the cost function is squared Euclidean distance, the solution to (6) is unique and can be expressed as

$$T = \nabla \phi \quad (3)$$

where  $\phi : \mathbb{R}^N \rightarrow \mathbb{R}$  is a convex potential function and  $\nabla \phi : \mathbb{R}^N \rightarrow \mathbb{R}^N$  denotes its monotone gradient. Furthermore, as proved in [16],  $T$  being the monotone gradient of  $\phi$  means  $T$  is cyclically monotone. This corresponds to the intuition that the transport routes taken by units of mass (probability) do not cross or intersect, as detailed in Sec. IX-A of [10]. Moreover, in this setting,  $T$  is smooth almost everywhere [14]. As described in [14], [17],  $T$  being smooth and one-to-one means that the constraint  $T_\# \mu = \nu$  can be expressed in terms of the corresponding densities as

$$\int_{T^{-1}(A)} p(\mathbf{x}) d\mathbf{x} = \int_A q(\mathbf{y}) d\mathbf{y} = \int_{T^{-1}(A)} q(T(\mathbf{x})) |\det \mathbf{J}_T(\mathbf{x})| d\mathbf{x}$$

where  $\mathbf{J}_T(\mathbf{x})$  denotes the  $N \times N$  Jacobian of  $T$  evaluated at  $\mathbf{x}$  and where the second equality follows from the change of variables formula applied to the substitution  $\mathbf{y} = T(\mathbf{x})$ . As described in [10], [11], this yields the following standard pushforward expression:

$$p(\mathbf{x}) = q(T(\mathbf{x})) |\det \mathbf{J}_T(\mathbf{x})| \quad (4)$$

Substituting for  $T = \nabla \phi$  reveals that the convex potential  $\phi$  satisfies a type of Monge-Ampère equation [10], [11], [17]

$$p(\mathbf{x}) = q(\nabla \phi(\mathbf{x})) \det(\mathbf{H}_\phi(\mathbf{x})) \quad (5)$$

<sup>1</sup>See Theorem 3.6 in [14] for generalizations of this setting.

where  $\mathbf{H}_\phi = \mathbf{J}_T$  is the  $N \times N$  positive semidefinite Hessian of  $\phi$ .

In summary, for Monge optimal transport problems with squared Euclidean cost (satisfying appropriate regularity conditions), the optimal transport map  $T$  exists and is unique. In this work, we specifically leverage (3) and (4) to learn  $T$ .

### IV. RELATED WORK

Brenier's theorem has inspired several approaches for training neural networks to learn the optimal transport mapping between continuous measures with squared Euclidean cost. For example, [18], [19] first parameterize a convex potential  $\phi$  using an input convex neural network (ICNN) [20] and train the gradient of the ICNN with respect to the input to approximate the optimal transport map. Unfortunately, training the gradient maps of ICNNs often leads to numerical optimization challenges [21]–[23].

Other works have considered parameterizing the monotone gradient map without first characterizing the underlying potential. For example, [22] parameterize a Gram factor of  $\mathbf{H}_\phi$ , the positive semidefinite Hessian of the potential  $\phi$ , and obtain the gradient map by numerically integrating the learned Hessian. The only known architecture suitable for parameterizing the Gram factor is a single-layer network of the form  $\sigma(\mathbf{W}\mathbf{x} + \mathbf{b})$ , which does not require numerical integration and serves primarily as a proof of concept. Moreover, the reliance on numerical integration poses significant challenges for optimal transport in high-dimensional settings.

In contrast, we proposed neural network architectures that are guaranteed to directly model monotone gradients of convex functions in [1]. Then, in [2], we proved that these neural networks are universal approximators of monotone gradient functions and introduced mGradNets that offer expanded parameterizations more amenable for optimization in practice. In this work, we leverage the Monge-Ampère equation to design a principled approach for training mGradNets to solve Monge optimal transport problems where both the source and target probability densities are known. This contrasts with our approach in [1], where the source probability density was unknown and we trained MGNs (predecessors of mGradNets) by maximizing the target likelihood of the mapped samples – the approach taken previously by [18].

### V. PROPOSED METHOD

We propose GradNetOT, a method for computing the optimal transport map in (2) with squared Euclidean cost and continuous probability measures  $\mu$  and  $\nu$ . Specifically, we are interested in finding  $T$  that minimizes the total transport cost:

$$\inf_{T: T_\# \mu = \nu} \int_{\mathcal{X}} \|\mathbf{x} - T(\mathbf{x})\|^2 d\mu(\mathbf{x}) \quad (6)$$

As a result of Brenier's theorem, the optimal transport map  $T$  solving (6) has the following properties: 1)  $T$  has a symmetric positive semidefinite Jacobian since it is the monotone gradient of a convex function; 2)  $T$  satisfies the pushforward expression in (4) that is closely related to the Monge-Ampère equation. Hence, we propose to parameterize  $T$  as a monotone gradient network (mGradNet) [1], [2], and train it to satisfy the constraint in (4). This leverages Brenier's theorem [11]: a function that is the gradient of a convex function and satisfies the constraint must be the unique optimal transport map.

mGradNets are a class of neural network architectures that are explicitly designed to correspond to monotone gradients of convex functions. The Jacobian of a mGradNet's output with respect to its input is always symmetric positive semidefinite – for all inputs  $\mathbf{x}$  and all parameter values  $\theta$  – thereby guaranteeing that for any mGradNet

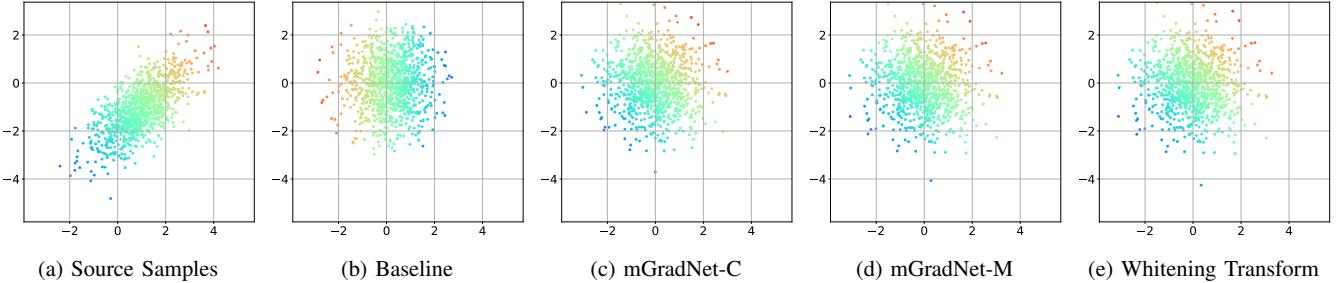


Fig. 1: OT from skewed to standard Gaussian. Points are colored based on their positions in the source distribution.

$T_\theta$ , there exists a convex potential  $\phi_\theta$  such that  $T_\theta = \nabla\phi_\theta$ . Moreover, mGradNets are *universal approximators* of gradients of convex functions and can be additionally tailored to correspond to subclasses of convex functions, such as convex functions expressible as the sum of convex ridge functions. We refer to our paper [2] for further details and proofs regarding mGradNets and GradNets (a generalization to gradients of  $L$ -smooth functions).

---

#### Algorithm 1 GradNetOT Training Procedure

---

**Require:** mGradNet  $T_\theta$ , source density  $p(\mathbf{x})$ , target density  $q(\mathbf{y})$ , learning rate  $\eta$ , batch size  $B$ , iterations  $I$

**for** iteration  $i = 1$  to  $I$  **do**

- Sample  $\{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_B\} \sim p(\mathbf{x})$
- Compute pushforward points:  $\mathbf{y}_j = T_\theta(\mathbf{x}_j)$ ,  $j = \{1, \dots, B\}$
- Compute Jacobians:  $\mathbf{J}_{T_\theta}(\mathbf{x}_j) = \frac{\partial}{\partial \mathbf{x}_j} T_\theta(\mathbf{x}_j)$
- Compute loss:

$$\mathcal{L}(\theta) = \frac{1}{B} \sum_{j=1}^B \ell(\log \det \mathbf{J}_{T_\theta}(\mathbf{x}_j), \log p(\mathbf{x}_j) - \log q(\mathbf{y}_j))$$

Update parameters:  $\theta \leftarrow \theta - \eta \nabla_\theta \mathcal{L}(\theta)$

**end for**

**return**  $\theta$

---

The training procedure, as provided in Algorithm 1, is inspired by the training of physics informed neural networks [24] designed to satisfy differential equations. During training, points  $\{\mathbf{x}_j\}_{j=1}^B$  are sampled from the source distribution  $p$  and transformed by the mGradNet  $T_\theta$ . Using the target density  $q$  to evaluate the density of the transformed points leads to computing the training label as  $\log p(\mathbf{x}_j) - \log(q(T_\theta(\mathbf{x}_j)))$ . The log determinant of the Jacobian of  $T_\theta$  (with respect to the input) is evaluated at  $\mathbf{x}_j$  and trained to match this label via standard backpropagation. The training procedure presented in Algorithm 1 minimizes the empirical version of the following objective over the mGradNet parameters  $\theta$ , where  $\ell$  is a loss function like the squared Euclidean distance function:

$$\mathbb{E}_{\mathbf{x} \sim p(\mathbf{x})} \ell(\mathbf{J}_{T_\theta}(\mathbf{x}), \log p(\mathbf{x}) - \log(q(T_\theta(\mathbf{x}))))$$

In contrast to approaches that first characterize the underlying potential only to later extract the gradient, our method directly characterizes the gradient map itself. Parameterizing the underlying convex potential  $\phi$ , e.g., using an ICNN, and solving (5) using a generalization of the approach presented would require computing the Hessian of  $\phi$  using auto-differentiation, which is much more computationally expensive than computing the Jacobian. By forgoing direct characterization of the potential, our method enables us to consider the expression in (4) involving the Jacobian.

## VI. EXPERIMENTS

We demonstrate the efficacy of the proposed GradNetOT procedure with a series of optimal transport tasks. In all experiments, we use the Adam optimizer and a fixed batch size of 1000.

### A. Gaussians

We consider the OT problem in (6) with quadratic cost and Gaussian measures  $\mu$  and  $\nu$ . The OT map between multivariate Gaussians  $\mathcal{N}(\boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0)$  and  $\mathcal{N}(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)$  has an analytic solution. In particular, when the target density is the multivariate standard normal ( $\boldsymbol{\mu}_1 = \mathbf{0}$ ,  $\boldsymbol{\Sigma}_1 = \mathbf{I}$ ), the OT map is given by the whitening transform  $T(\mathbf{x}) = \boldsymbol{\Sigma}_0^{-1/2}(\mathbf{x} - \boldsymbol{\mu}_0)$ . We train the mGradNet-C and mGradNet-M architectures from [2] using Algorithm 1. Using the same method, we also train a baseline feedforward neural network that is not constrained to be the gradient of a convex function. We visualize the OT maps for the neural network baseline, mGradNet-C, mGradNet-M and the ground truth OT map whitening in Fig. 1. The points are colored in the figure according to their corresponding positions in the source distribution. As shown in the figure, the neural network baseline fails to learn the OT map as indicated by the fact that the ordering of the points changes drastically: the red source points are on the right in Fig. 1a, but they are mapped to the red points on the left in Fig. 1b. This indicates that the baseline neural network is not learning a monotone mapping. Meanwhile, the mGradNet-C and mGradNet-M can learn the OT map and their displacements closely match those of the known OT map, namely the whitening transform.

To test with higher-dimensional Gaussians, we randomly generate a  $d$ -dimensional source Gaussian distribution by sampling the mean from the standard Gaussian distribution and by sampling the covariance matrix from a Wishart distribution with  $d+1$  degrees of freedom and identity scale matrix. After using Algorithm 1 to train mGradNets to learn the OT map to a standard Gaussian, we evaluate the neural networks on a set of 1000 test points. The whitening transform applied to the test points again yields the ground truth target under the optimal transport map. We use our mGradNets to map the test points and compute the mean squared between them and the ground truth mapped points. Fig. 2 shows that, for all dimensions, both mGradNets closely approximate the optimal transport map and the mGradNet-M consistently outperforms the mGradNet-C.

### B. Image Morphing as Optimal Transport

We consider the problem of optimally transporting a distribution over pixel locations from their known initial locations to desired locations, thereby effectively morphing the image. We model image pixel intensities from the MNIST dataset as samples from a continuous density over the unit square  $[0, 1]^2$ . Each pixel is treated as a point mass located on a uniform grid and we fit a smooth density using kernel density estimation (KDE). Given an  $n \times n$  image with intensity

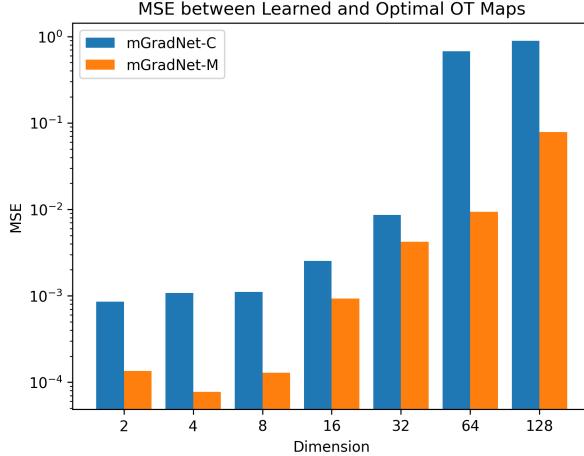


Fig. 2: MSE for 1000 samples from skewed multivariate Gaussians mapped to standard Gaussians by mGradNets.

values  $I(i, j)$ , we model the density over pixel locations  $\mathbf{x} \in [0, 1]^2$  as the following mixture of Gaussians:

$$p(\mathbf{x}) = \sum_{i=0}^{n-1} \sum_{j=0}^{n-1} I(i, j) \cdot \mathcal{N}\left(\mathbf{x}; \left(\frac{i}{n-1}, \frac{j}{n-1}\right), \sigma^2 \mathbf{I}\right)$$

with  $\sigma^2 = 10^{-4}$  and  $n = 28$  for MNIST. We train an mGradNet-M to learn the optimal transport map between these densities using Algorithm 1. During training, we exponentially decay the learning rate from  $1e-2$  to  $1e-4$  over 10000 iterations and also decay  $\sigma^2$  in the same manner to facilitate more stable training. In Fig. 3, we illustrate the learned OT displacement map

$$\mathbf{x}^{(t)} = (1-t)\mathbf{x}^{(0)} + tT_\theta(\mathbf{x}^{(0)})$$

for morphing between a 0 and a 4. The first column corresponds to the transport map learned by the mGradNet-M. The second column corresponds to the transport map from the barycentric projection. The barycentric projection converts a discrete optimal transport plan, from solving the Monge-Kantorovich relaxation, to an optimal transport map. Given an optimal transport plan  $\gamma \in \mathbb{R}^{N_x \times N_y}$ , which describes how much mass to move from source points  $\{\mathbf{x}_i\}_{i=1}^{N_x}$  to target points  $\{\mathbf{y}_j\}_{j=1}^{N_y}$ , the barycentric projection described in [13] is defined as:

$$T(\mathbf{x}_i) = \frac{\sum_{j=1}^{N_y} \gamma_{ij} \mathbf{y}_j}{\sum_{j=1}^{N_y} \gamma_{ij}}$$

Specifically, the barycentric projection approximates a Monge (not mass splitting) transport map using a potentially mass-splitting Kantorovich plan. When the solution to the Kantorovich problem  $\gamma$  does not split mass, and a Monge map exists, then the barycentric projection recovers the optimal Monge map in the limit as the number of source and target points increases [13], [25]. However, computing the optimal transport plan scales quadratically with the number of points, motivating the learning of a parametric OT map, which avoids the cost of solving and storing a full discrete transport plan.

In Table I we report the mean squared error (MSE) between the barycentric projection  $T(\mathbf{x}_i)$  and the learned mapping  $T_\theta(\mathbf{x}_i)$  over 1000 samples  $\mathbf{x}_i$  for various source and target image pairs. Each entry in the table reports the difference between the transport map learned by an mGradNet-M and the barycentric projection, when transporting mass from the source density (row) to the target density (column).

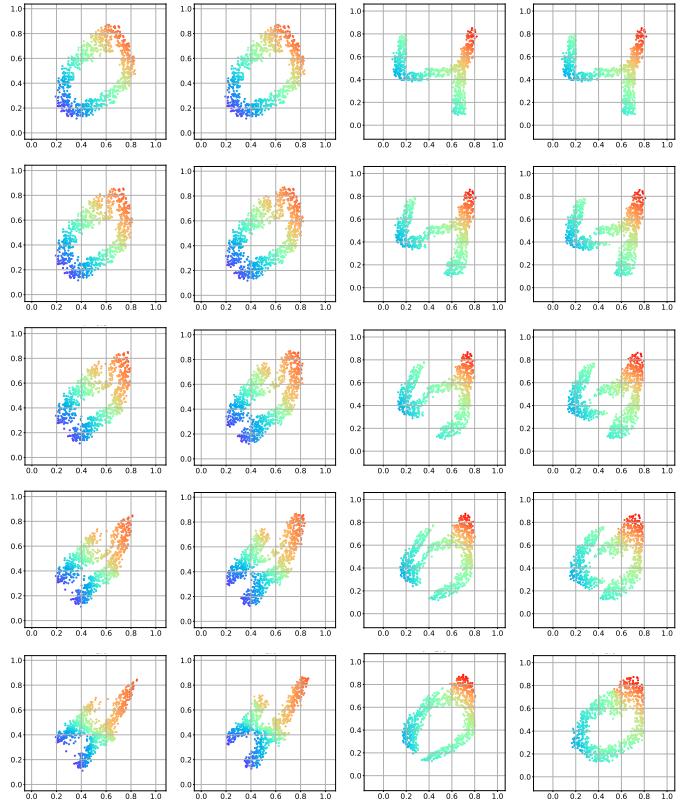


Fig. 3: Left two columns: OT and barycentric interpolation from 0 to 4. Right two columns: from 4 to 0. Rows correspond to different interpolation times  $t \in \{0.0, 0.25, 0.5, 0.75, 1.0\}$ .

TABLE I: MSE between learned OT map and barycentric projection for digits 0–4 from MNIST.

		Target				
		0	1	2	3	4
Source	0	0.00138	0.00031	0.00566	0.00036	0.00145
	1	0.00146	0.00024	0.00438	0.00178	0.00104
	2	0.00123	0.00038	0.00146	0.00055	0.00055
	3	0.00117	0.00026	0.00253	0.00051	0.00105
	4	0.00273	0.00034	0.00170	0.00159	0.00082

Lower MSE values in the table imply that the transport cost of the learned OT map is closer to that of the optimal transport map. Overall, we observe both numerically and qualitatively in Fig. 3, that our proposed method closely matches the performance of the barycentric projection.

## VII. CONCLUSION

We propose a method for learning optimal transport maps using mGradNets – neural networks guaranteed to correspond to gradients of convex functions. Inspired by Brenier’s theorem, our approach finds solutions to the Monge problem with quadratic Euclidean cost by directly searching over the space of gradients of convex potentials by optimizing a Monge-Ampère loss. We validate our method through a series of experiments demonstrating its effectiveness. Our framework opens avenues for further research in the design and training of theoretically-motivated neural networks for solving optimal transport problems.

## REFERENCES

- [1] Shreyas Chaudhari, Srinivasa Pranav, and José M. F. Moura, “Learning gradients of convex functions with monotone gradient networks,” in *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2023, pp. 1–5.
- [2] Shreyas Chaudhari, Srinivasa Pranav, and José M. F. Moura, “Gradient networks,” *IEEE Transactions on Signal Processing*, vol. 73, pp. 324–339, 2025.
- [3] Gaspard Monge, “Memoire sur la theorie des deblais et des remblais, histoire de l’academie royale des sciences annee 1781,” *Avec les Memoires de Mathematique & de Physique, pour la même Annee*(2e partie), pp. 34–38, 1784.
- [4] Leonid V. Kantorovich, “On the translocation of masses,” in *Dokl. Akad. Nauk. USSR (NS)*, 1942, vol. 37, pp. 199–201.
- [5] Leonid V. Kantorovich, “On a problem of Monge,” in *CR (Doklady) Acad. Sci. URSS (NS)*, 1948, vol. 3, pp. 225–226.
- [6] Yunes Alqudsi and Murat Makaraci, “UAV swarms: research, challenges, and future directions,” *Journal of Engineering and Applied Science*, vol. 72, no. 1, pp. 12, 2025.
- [7] Shaojun Ma, Mengxue Hou, Xiaojing Ye, and Haomin Zhou, “High-dimensional optimal density control with wasserstein metric matching,” in *2023 62nd IEEE Conference on Decision and Control (CDC)*. IEEE, 2023, pp. 6813–6818.
- [8] Pierre-André Chiappori, Robert J. McCann, and Lars P. Nesheim, “Hedonic price equilibria, stable matching, and optimal transport: equivalence, topology, and uniqueness,” *Economic Theory*, vol. 42, no. 2, pp. 317–354, 2010.
- [9] Samuel Cohen, Brandon Amos, and Yaron Lipman, “Riemannian convex potential maps,” in *International Conference on Machine Learning*. PMLR, 2021, pp. 2028–2038.
- [10] Albert Bonnefous, Yann Brenier, and Roya Mohayee, “Monge–Ampère gravity, optimal transport theory, and their link to the galileons,” *Physical Review D*, vol. 110, no. 12, pp. 123513, 2024.
- [11] Yann Brenier, “Polar factorization and monotone rearrangement of vector-valued functions,” *Communications on Pure and Applied Mathematics*, vol. 44, no. 4, pp. 375–417, 1991.
- [12] Matthew Thorpe, “Introduction to optimal transport,” *Notes of Course at University of Cambridge*, vol. 3, 2018.
- [13] Gabriel Peyré, Marco Cuturi, et al., “Computational optimal transport: With applications to data science,” *Foundations and Trends® in Machine Learning*, vol. 11, no. 5-6, pp. 355–607, 2019.
- [14] Guido De Philippis and Alessio Figalli, “The Monge–Ampère equation and its link to optimal transportation,” *Bulletin of the American Mathematical Society*, vol. 51, no. 4, pp. 527–580, 2014.
- [15] Filippo Santambrogio, “Optimal transport for applied mathematicians,” *Birkhäuser, NY*, vol. 55, no. 58–63, pp. 94, 2015.
- [16] R. Tyrrell Rockafellar, *Convex analysis*, vol. 18, Princeton University Press, 1970.
- [17] Jean-David Benamou and Yann Brenier, “A computational fluid mechanics solution to the Monge–Kantorovich mass transfer problem,” *Numerische Mathematik*, vol. 84, no. 3, pp. 375–393, 2000.
- [18] Chin-Wei Huang, Ricky TQ Chen, Christos Tsirigotis, and Aaron Courville, “Convex potential flows: Universal probability distributions with optimal transport and convex optimization,” in *International Conference on Learning Representations*, 2020.
- [19] Ashok Makkluva, Amirhossein Taghvaei, Sewoong Oh, and Jason Lee, “Optimal transport mapping via input convex neural networks,” in *International Conference on Machine Learning*. PMLR, 2020, pp. 6672–6681.
- [20] Brandon Amos, Lei Xu, and J. Zico Kolter, “Input convex neural networks,” in *International Conference on Machine Learning*. PMLR, 2017, pp. 146–155.
- [21] Charlotte Bunne, Andreas Krause, and Marco Cuturi, “Supervised training of conditional Monge maps,” *Advances in Neural Information Processing Systems*, vol. 35, pp. 6859–6872, 2022.
- [22] Jack Richter-Powell, Jonathan Lorraine, and Brandon Amos, “Input convex gradient networks,” *arXiv preprint arXiv:2111.12187*, 2021.
- [23] Alexander Korotin, Lingxiao Li, Aude Genevay, Justin M Solomon, Alexander Filippov, and Evgeny Burnaev, “Do neural optimal transport solvers work? a continuous Wasserstein-2 benchmark,” *Advances in Neural Information Processing Systems*, vol. 34, pp. 14593–14605, 2021.
- [24] Maziar Raissi, Paris Perdikaris, and George E Karniadakis, “Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations,” *Journal of Computational physics*, vol. 378, pp. 686–707, 2019.
- [25] Cédric Villani, *Optimal transport: old and new*, vol. 338, Springer, 2009.