Optimal Transport for Super Resolution Applied to Astronomy Imaging

Michael Rawson
Department of Mathematics
University of Maryland at College Park
Maryland, USA
rawson@umd.edu

Jakob Hultgren
Department of Mathematics
University of Maryland at College Park
Maryland, USA
hultgren@umd.edu

***Abstract*—Super resolution is an essential tool in optics, especially on interstellar scales, due to physical laws restricting possible imaging resolution. We propose using optimal transport and entropy for super resolution applications. We prove that the reconstruction is accurate when sparsity is known and noise or distortion is small enough. We prove that the optimizer is stable and robust to noise and perturbations. We compare this method to a state of the art convolutional neural network and get similar results for much less computational cost and greater methodological flexibility.**

***Index Terms*—optimal transport, Wasserstein distance, super resolution, compressed sensing, sparse imaging, sparse regularization, sparsity, maximum entropy, convolutional neural network**

I. Introduction and Background

A. Super Resolution

Super resolution seeks to improve image resolution without further data collection. This is useful when important features or pixels are missing. Improving the measurement device, such as a camera or telescope, will improve the image resolution but only up to limits governed by physical laws, for example the diffraction limit. Super resolution can increase image resolution beyond this point given constraints that give a well-posed inverse problem. The most common constraint is that the true image is either sparse or smooth in some basis. For example, Gaussian noise is a common input that blurs important features. Removing additive Gaussian noise can be done, imperfectly, by solving an inverse problem that constrains total variation and hence enforces smoothness [14].

A more general solution is to minimize with respect to a regularizing term that maximizes sparsity. Compressed sensing methods often minimize an objective function involving the L^1 norm of the solution [3]. Minimizing L_0 maximizes sparsity but L^1 is usually used instead for its convex properties. Another regularizer that maximizes sparsity is entropy [4], [10]. Neural networks or deep learning has more recently been used for inverse problems, especially on images [15].

B. Optimal Transport

Optimal transport has a long history in pure and applied math, and has been used recently in the field of imaging [6]. One starts with a source distribution, $\mu \in \mathbb{P}(X)$, target distribution, $\nu \in \mathbb{P}(Y)$, and a cost $C : X \times Y \to \mathbb{R}$ on spaces X and Y. In the discrete setting relevant to scientific computing (where X and Y are finite sets), distributions are represented as finite dimensional vectors. Given two positive vectors of L^1-norm equal 1, $\mu \in \mathbb{R}^n_{>0}, \nu \in \mathbb{R}^m_{>0}$, together with a cost C represented as an $n \times m$ matrix (C_{ij}), the optimal transport plan between μ and ν is

$$\arg \min_{P \in \Pi(\mu,\nu)} \sum_{i=1}^{n} \sum_{j=1}^{m} C_{ij} P_{ij} \tag{1}$$

where $\Pi(\mu, \nu)$ is the set of transport plans from μ to ν:

$$\Pi(\mu,\nu) = \{P \in \mathbb{R}^{n \times m} : \sum_{i=1}^{n} P_{ij} = \nu_j \ \forall j, \sum_{j=1}^{m} P_{ij} = \mu_i \ \forall i\}.$$

When $m = n$ and C defines a metric on $\{1, \ldots, n\}$ (i.e. $d(i,j) := C_{ij}$ is symmetric, non-degenerate, and satisfies the triangle inequality) then the minimum value

$$d_W(\mu,\nu) := \min_{P \in \Pi(\mu,\nu)} \sum_{i=1}^{n} \sum_{j=1}^{m} C_{ij} P_{ij} \tag{2}$$

defines a metric on $\mathbb{P}(\{1, \ldots, n\})$ often referred to as the 1st Wasserstein metric or just the Wasserstein metric [20].

C. Entropy

For a probability mass function $p : \mathcal{J} \to \mathbb{R}$ (non-negative and sums to 1), we will use $H(p)$ to denote its entropy,

$$H(p) = -\sum_{\iota \in \mathcal{J}} p(\iota) \ln(p(\iota)). \tag{3}$$

As we represent probability densities with vectors and matrices, we will consider the indices to be in the domain $\mathcal{J}$. In (3), we use the convention that $0 \cdot \ln(0) = 0$. With this convention, H defines a continuous non-negative function on the probability simplex, differentiable in the interior of the probability simplex and with a derivative unbounded at the boundary. The key idea is that sparse arrays have low entropy.

D. The Sinkhorn algorithm

The distance d_W in Equation (2) can be computed exactly using methods from linear programming. However, for large n the quickly growing computation times excludes this from many applications [6]. In applications involving large data sets,

d_W is often replaced by its *entropic regularization*, which is more feasible from a computational perspective [17]. Given a small constant $\epsilon > 0$, the ϵ-regularized distance between $\mu, \nu \in \mathbb{P}(\{1, \dots, n\})$ is

$$d_W^\epsilon(\mu, \nu) := \sum_{i=1}^{n} \sum_{j=1}^{m} C_{ij} P_{ij}^* \tag{4}$$

where

$$P^* = \arg \min_{P \in \Pi(\mu, \nu)} \sum_{i=1}^{n} \sum_{j=1}^{m} C_{ij} P_{ij} - \epsilon H(P). \tag{5}$$

The regularized objective function in (5) is strictly convex and proper, hence always admits a unique minimizer. While the minimizer in the true optimal transport problem is usually very sparse, the entropy term in (5) pushes the minimizer away from the boundary of the unit simplex, producing a less sparse minimizer. As $\epsilon \to 0$, this minimizer converges to a minimizer of (2) (the minimizer with highest entropy if there are more than one), and d_W^ϵ converges to d_W [17].

A simple application of Lagrange multipliers show that the minimizer of (5) is the unique element in $\Pi(\mu, \nu)$ on the form

$$P_{ij} = \sum_{i=1}^{n} \sum_{j=1}^{m} f_i e^{-C_{ij}/\epsilon} g_j \tag{6}$$

for some unknown positive multipliers $f = (f_1, \dots, f_n),\ g = (g_1, \dots g_n)$ [17]. Determining f and g from μ, ν and the matrix $(e^{-C_{ij}/\epsilon})$ is known as the matrix scaling problem, and a standard algorithm to find approximate solutions is the iterative proportional fitting procedure, also known as the Sinkhorn Algorithm [17]. The matrix (6) lies in $\Pi(\mu, \nu)$ if $\sum_i P_{ij} = \mu_i$ for all i and $\sum_j P_{ij} = \nu_j$ for all j. The Sinkhorn algorithm proceeds iteratively, alternating between updating f so that the first of these conditions is satisfied and updating g so that the second of these conditions is satisfied (see Algorithm 1).

E. Wasserstein Distance Gradient

The multipliers $f \in \mathbb{R}^n$ and $g \in \mathbb{R}^m$ in the previous section can be thought of as dual variables for the optimization problem (5). More precisely,

$$F = \epsilon \ln(f),\ G = \epsilon \ln(g)$$

are the Lagrange multipliers for (4) [6]. When considering the regularized distance between μ and ν as a function of μ (keeping ν fixed) this can, at least for positive μ and ν, be exploited to approximate its gradient (see for example [8], [12]). The Sinkhorn Algorithm, used to approximate $d_\mu^\epsilon(\mu, \nu)$ and its gradient with respect to μ is summarized in Algorithm 1. Note that the output F and G of Algorithm 1 needs to be projected onto the tangent space of the probability simplex to yield an approximation of the true gradients.

Remark. *In Algorithm 1, if one is only interested the regularized distance d_W^ϵ, the assumption of positivity of μ and ν can be relaxed to non-negativity. However, reflecting the fact that the entropy term in* (4) *pushes the minimizer away from the boundary, any entry F_i in F will return as $+\infty$ if $\mu_i = 0$.*

Algorithm 1: The Sinkhorn Algorithm for Regularized Optimal Transport Distances

```
Input:
  $\mu, \nu \in \mathbb{R}^n$ : positive probability vectors
  $C \in \mathbb{R}^{n \times n}$ : cost matrix
  $\epsilon$ : positive regularization parameter
Output:
  $d_W^\epsilon \in \mathbb{R}$ : regularized distance between $\mu$ and $\nu$
  $F \in \mathbb{R}^n$ : gradient of $d_W^\epsilon(\mu, \nu)$ with resp. to $\mu$ at $\mu$
  $G \in \mathbb{R}^n$ : gradient of $d_W^\epsilon(\mu, \nu)$ with resp. to $\nu$ at $\nu$
Begin:
$f = (1, \dots, 1) \in \mathbb{R}^n$
$g = (1, \dots, 1) \in \mathbb{R}^n$
while $f$ and $g$ have not converged do
    for $1 \leq i \leq n$ do
        $f_i = \mu_i / \left( \sum_j \exp(-C_{ij}/\epsilon) g_j \right)$
    end
    for $1 \leq j \leq n$ do
        $g_j = \nu_j / \left( \sum_i \exp(-C_{ij}/\epsilon) f_i \right)$
    end
end
$d_W^\epsilon = \sum_{i=1}^n \sum_{j=1}^m f_i g_j \exp(-C_{ij}/\epsilon) C_{ij}$
$F = -\epsilon \ln(f)$
$G = -\epsilon \ln(g)$
```

II. Wasserstein Inverse Problem for Super Resolution

We propose two super resolution inverse problems that produce sparse solutions which are near to the measurement in Wasserstein distance. For a measurement ν and positive regularization parameters λ and λ', we define the *sparse approximation* of ν as a minimizer

$$\mu_* = \arg \min_{\mu \in \mathbb{P}(X)} d_W^\epsilon(\mu, \nu) + \lambda H(\mu). \tag{7}$$

and the *sparse retrieval* of ν as a minimizer

$$\mu_* = \arg \min_{\mu : d_W(\mu, \nu) < \lambda'} H(\mu). \tag{8}$$

Problem (7) and (8) are essentially dual, and for generic data ν there is a mapping $\lambda \mapsto \lambda'(\lambda)$ such that μ is a solution to (7) if and only if μ is a solution to (8). We will approach (8) from a theoretical perspective in Section III but use (7) in our application since it fits well into a gradient descent method.

At least one minimizer exists by compactness of the finite dimensional probability simplex. The entropy term in (7) favors sparse solutions. Naturally, there is a trade-off between sparsity of the solution and proximity to the measurement. How these two objectives are prioritized is governed by λ. For $\lambda = 0$, no priority is given to the goal of sparsity and $\mu_* = \nu$. As λ increases, μ_* turns into an increasingly sparse approximation of ν and when $\lambda \to \infty$, $\|\mu_*\|_0 \to 1$.

This inverse problem is useful whenever there is a natural distance, or cost function, on the index set of ν. If, for example, ν is given in Fourier space and each entry μ_i corresponds to a frequency σ_i, then two natural choices for the cost C_{ij} are $C_{ij} = |\sigma_i - \sigma_j|$ and $C_{ij} = |\ln(\sigma_i/\sigma_j)|$. In the application we describe below, each entry in ν describes the intensity of a pixel in a 32×32 image and C_{ij} is chosen as the L^2-distance between the i^{th} pixel and j^{th} pixel.

Remark. *This method can be contrasted to maximum entropy methods in statistical physics, where the probability distribution with highest entropy (under constraints dictated by observations) is chosen as the best representative of the current state of knowledge about a system. In our context, we work with the crucial assumption of sparsity, which motivates minimizing the entropy instead of maximizing it.*

III. Main Results

We will let $\nu \in P(\{1, \dots, n\})$ be a sparse signal (i.e. $\|\nu\|_0 < n$ is small), and use $\bar{\nu}$ to denote this signal with noise and distortion. In our application, we are interested in determining the support of ν (i.e. the indices of all non-zero entries in ν) from $\bar{\nu}$. For two probability vectors μ and ν we will say that μ identifies the structure of ν if they have the same support, i.e. if $\mu_i > 0$ if and only if $\nu_i > 0$ for all i. Our main theorem (Theorem 2 below) shows that the minimizer of (7) identifies the structure of ν under the assumptions that ν is sparse and the noisy signal $\bar{\nu}$ is close to ν in optimal transport distance. As is indicated by Theorem 1 below, the latter assumption is natural when dealing with Gaussian noise since the optimal transport distance, unlike total variation and L^p distances, take the geometry of the space into account.

A. Optimal Transport Bound on Gaussian Noise

Let the probability distribution $\nu := \frac{1}{k}\sum_{i=1}^{k} \delta_{p_i}$ be a sparse signal in $\mathbb{P}(\mathbb{R}^d)$ where δ is the Dirac delta. Assume the noisy signal $\tilde{\nu}$ is produced in the following way: For each p_i in the sum above, we sample n points $x_i^1, \dots, x_i^n$ in $\mathbb{R}^d$ according to a normal distribution centered at p_i with independent components of variance σ^2. Let $N = kn$ be the number of points sampled and $\tilde{\nu} = \frac{1}{N}\sum_{i=1}^{n}\sum_{j=1}^{m} \delta_{x_i^j}$ be the noisy signal.

Theorem 1. *Given a sparse signal $\nu \in \mathbb{P}(\mathbb{R}^d)$ giving rise to a noisy signal $\tilde{\nu}$ as described above, the optimal transport distance between ν and $\tilde{\nu}$ is bounded by $\frac{\sigma^2}{N} X_N$ where X_N is a random variable with distribution χ^2_{dN}. In particular, the expected value and variance of $\frac{\sigma^2}{N} X_N$ are $d\sigma^2$ and $2d\sigma^4/N$, respectively.*

Proof. The optimal transport cost is bounded from above by the cost of the transport plan sending each x_i^j to p_i. The cost of this plan is $\frac{1}{N}\sum |x_i^j - p_i|^2$. By assumption, each term in this sum is the squared sum of d normal distributed random variables with mean 0 and variance σ^2. □

B. Reconstructing the Support of a Sparse Signal

Theorem 2. *Assume ν is a sparse signal and $\bar{\nu}$ is a noisy signal such that $d_w(\nu, \bar{\nu}) < \delta$. Then the solution of*

$$\mu = \arg \min_{\mu : d_W(\bar{\nu}, \mu) \leq \delta} H(\mu) \qquad (9)$$

will identify the structure of μ, i.e. have the same support as μ, if $||\nu||_0 \leq ||\mu||_0$ for all μ such that $d_W(\mu, \bar{\nu}) < 2\delta$, with equality only if μ and ν has the same support.

Remark. *The conditions in Theorem 2 can be summarized as a low enough noise level δ and enough sparsity of the true signal ν (making it a local minimizer of the L^0-norm). It is interesting to note that these conditions are essentially necessary: if the inequality in Theorem 2 is violated by some μ closer than δ to $\bar{\nu}$, then the solution of* (9) *does not identify the structure of ν.*

Remark. *Noise is high entropy, hence it is expected that the noise can be removed by minimizing the entropy. However, if the signal-to-noise ratio is too low, this reconstruction is underdetermined.*

Proof of Theorem 2. By the triangle inequality, the feasible set in (9) is contained in the ball centered at $\bar{\nu}$ of radius 2δ. As the feasible set in (9) contains ν, this means any solution of (9) has to be ν or have the same support as ν. □

Theorem 3. *Fix a positive probability vector $\nu \in \mathbb{R}^d_{>0}$ such that all elements of ν are distinct. Then the sparse recovery is continuous to perturbations around ν for small λ, i.e. for every $\epsilon' > 0$ there exists $\delta > 0$, such that if $d_W(\nu, \nu') < \delta$,*
$\mu_* = \arg\min_{\mu \in \mathbb{P}(X) : d_W(\mu,\nu) < \lambda} H(\mu)$, and
$\mu'_* = \arg\min_{\mu \in \mathbb{P}(X) : d_W(\mu,\nu') < \lambda} H(\mu)$ then $\|\mu_* - \mu'_*\| < \epsilon'$.

Proof. The assumption on ν guarantees that minimizers are unique for small λ. Continuity of the minimizer then follows from smoothness of H. □

IV. Minimizing the objective

We solve (7) using a gradient descent method with variable step size. More precisely, letting $J(\mu) := d_W^{\epsilon}(\mu, \nu) + \lambda H(\mu)$ be the objective we set the step size to $\alpha_* := \sup\{\alpha > 0 : J(\mu - \alpha \nabla J|_{\mu}) < J(\mu)\}$. As mentioned in Section I-C, the entropy is not differentiable on the boundary of the probability simplex. An effect of this is that the output F in Algorithm 1 is infinity in all indices where μ is zero. We circumvent this problem by defining the i'th entry in the gradient of J to be 0 whenever $\mu_i = 0$. Geometrically, this means that whenever the algorithm reaches a sub-simplex of the probability simplex, it ignores the component of the gradient orthogonal to this sub-simplex, thus remaining in this sub-simplex for the rest of the algorithm. Gradient descent will converge to a local minimum on the compact probability simplex since the objective is smooth when restricted to the local simplex face. Algorithm 2 contains the pseudocode and also makes the star cluster classification described in Section VI.

V. Simulation

We first show this method's results on a low dimensional example. For example, let the measurement be
$\nu = (0.2, 0.15, 0, 0, 0, 0.1, 0.15, 0.2, 0.15, 0.1)^T$. With sparsity

Algorithm 2: Optimal Transport Star Cluster Prediction

```
Input:
    $X \in \mathbb{R}^{N\times m\times m}$ : $N$ images size $m \times m$
    $\lambda \in \mathbb{R}$ : positive noise level
    $0 < \epsilon < 1$ : optimal transportation regularization
    $C \in \mathbb{R}^{m^2\times m^2}$ : cost matrix
    $J_{\lambda,\epsilon}(x, v) := d_W^\epsilon(x, v) + \lambda H(v)$
Output:
    $K \in \mathbb{R}^N$ : star cluster classification
Begin:
$K = 0$
for $i = 1, 2, ..., N$ do
    $v = X_n$;  $w = 1$
    while  $v$ has not converged do
        $w = \nabla d_W^\epsilon(X_n, \cdot)|_v + \lambda\nabla H|_v$
        $w = w - \langle w, \frac{1}{m}\mathbb{1}\rangle \cdot \frac{1}{m}\mathbb{1}$
        $\alpha = \sup\{\alpha \in \mathbb{R} : J_{\lambda,\epsilon}(v) > J_{\lambda,\epsilon}(v - \alpha w)\}$
        $\alpha = \min\{0.01, \alpha\}$
        $v = v - \alpha w$; $v = diag(\mathbb{1}_{v>0})\ v$; $v = v/\|v\|_1$
    end
    $V_i = v$; $\delta = \max V_i$
    if  $rank(H_0(V_i^{-1}([0.75\delta,\ \delta]))) == 1$ then
        $K_i = 1$
    end
end
```

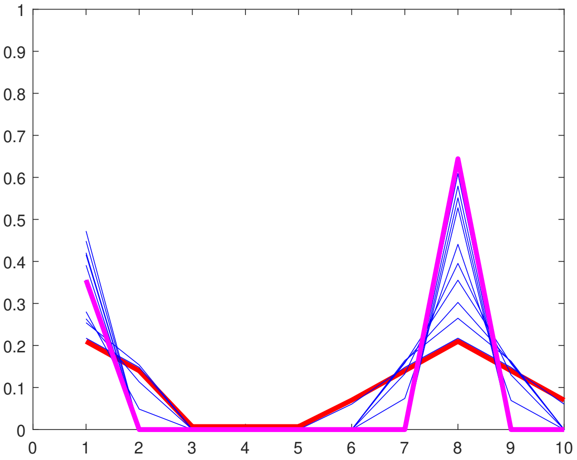

Fig. 1. Plot of super resolution O.T. method Algorithm 2. Red line is initial distribution. Blue lines are steps along gradient of Equation (7). Pink line is final, converged distribution. $\lambda = 10$. epsilon = 0.1. Max Sinkhorn iterations = 5000. Gradient step size = 0.01. Gradient steps=50.

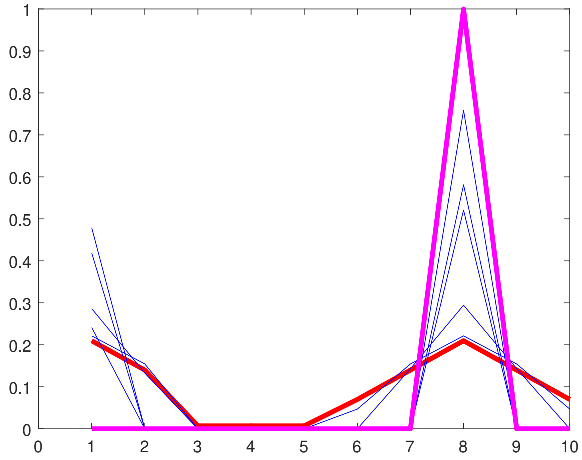

Fig. 2. Plot of super resolution O.T. method Algorithm 2. Red line is initial distribution. Blue lines are steps along gradient of Equation (7). Pink line is final, converged distribution. $\lambda = 100$. epsilon = 0.1. Max Sinkhorn iterations = 5000. Gradient step size = 0.01. Gradient steps=50.

parameter $\lambda = 10$ the method produces the sparse approximation $(0.35, 0, 0, 0, 0, 0, 0, 0.65, 0, 0)$. This reflects the fact that ν has two peaks, one peak centered at position 1 and one peak centered at position 8, and that 35% of the mass of ν is situated close to position 1 and 65% of the mass of ν is situated close to position 8. Figure 1 plots ν, the gradient descent steps, and the final result. With sparsity parameter $\lambda = 100$, the method produce the sparse approximation $(0, 0, 0, 0, 0, 0, 0, 1, 0, 0)$, reflecting the fact that most of the mass of ν is part of a peak centered at position 8. Figure 2 plots ν, the gradient descent steps, and the final result.

VI. Star Clustering Application

TABLE I
CONFUSION MATRIX OF O.T. METOD ALGORITHM 2 ON LEGUS DATA COMPARED TO STARCNET [16]. COLUMN GIVES STARCNET CLASSIFICATION AND ROW GIVES ALGORITHM 2 CLASSIFICATION.

	StarcNet Cluster	StarcNet Not Cluster
O.T. Cluster	25% (32)	13.3% (17)
O.T. Not Cluster	12.5% (16)	49.2% (63)

The formation and evolution of star clusters provide insight into the processes governing the birth of stars as well as the dynamical evolution of galaxies [16]. In order to save human hours and get reproducible results, it is of interest to algorithmically detecting star clusters in images of sky patches. Many methods have been proposed to algortihmically detect star clusters, including CLEAN [11], Multiscale CLEAN [5], IUWT-based CS [13], decision trees [9] and optimal sheaves [19]. The state of the art method trains a convolutional neural network (CNN) to classify each sky patch or region in an image as containing a star cluster or not [16]. These neural networks are notoriously computationally expensive, sensitive to noise, and inflexible to appending or removing data variables.

We propose using the Wasserstein inverse problem (7) to detect star cluster locations. Our dataset consists of measurements from the Hubble Space Telescope in the survey Treasury Project LEGUS (Legacy ExtraGalactic Ultraviolet Survey) [2].

This data set consists of 32×32 pixel images of star patches. Each image comes in 5 frequency bands (NUV, U, B, V, and I) [2]. We encode each of these in a probability vector ν where each entry correspond to the intensity of a pixel, normalized to sum to 1. Algorithm 2 produces a sparse approximation each image which is classified as a star cluster if it contains just one 'peak'. Specifically, the sparse image is made binary (1 and 0) by thresholding at 75% of the max of the image. Then the number of 'peaks' is the number of connected components in the binary image. This is the rank of the 0^{th} homology of the binary image, denoted $rank(H_0(V_i^{-1}([0.75\delta,\ \delta])))$ in Algorithm 2. We perform this calculation for each of the 5 frequency bands that were measured. Then these 5 predictions vote to produce the final prediction for the image in question.

Algorithm 2 can be compared to the method of producing a binary image directly from the source image, without first producing a sparse approximation, and counting the number of connected components in this. The accuracy rate of this naive approach is 46% with respect to the CNN. Algorithm 2 increases in accuracy to 74% with respect to the CNN. The CNN accuracy rate is 86% with respect to experts, but even experts agree with each other only around 70%-75% [1], [9], [21]. Given that experts are the baseline, it is impossible, without overfitting, for a computational model to do better than that. Therefore our method provides a very high performance given that no neural network training, which often takes weeks of compute time, is required. Additionally, the O.T. method is less sensitive to noise than a CNN, see [22], which we describe and bound in Theorems 1, 2, and 3. Finally, with our method, variables can be simply added and removed where Equation (7) is quickly recalculated.

We give the confusion matrix in Table I from our calculations. We test on 128 random samples. The maximum Sinkhorn iterations is 500. The cost C_{ij} is chosen as the L^2-distance between the i^{th} pixel and j^{th} pixel. The H_0 threshold is 0.75. The initial gradient descent step size is 0.001. Wasserstein parameter $\epsilon = 0.001$. Sparsity parameter $\lambda = 1$. When specifying the accuracy rates in the previous paragraph we use the classification results of the CNN in [16] as the definition of the correct classification.

VII. Conclusion

Optimal transportation is more efficient, robust, and flexible than CNNs. We proved that optimal transportation will reconstruct sparse sources and is robust to noise. This is relevant for correcting distortions and noise in imaging which we showed for star cluster detection. Another benefit of a predictive model for star clusters is that it can produce a *policy* that informs where future surveys should look for star clusters [7], [18].

Acknowledgment

The last author thanks the Knut and Alice Wallenberg Foundation (Grant 2018-0357), BSF (Grant 2020329) and NSF (Grant DMS-1906370).

References

[1] A. Adamo, J. Ryon, M. Messa, H. Kim, K. Grasha, D. Cook, D. Calzetti, et al., "Legacy ExtraGalactic UV Survey with The Hubble Space Telescope: Stellar Cluster Catalogs and First Insights Into Cluster Formation and Evolution in NGC 628," *The Astrophysical Journal*, vol. 841, no. 2, p. 131, 2017, doi: 10.3847/1538-4357/aa7132.

[2] D. Calzetti, J. C. Lee, E. Sabbi, A. Adamo, L. J. Smith, J. E. Andrews, L. Ubeda, et al., "LEGACY EXTRAGALACTIC UV SURVEY (LEGUS) WITH THE HUBBLE SPACE TELESCOPE. I. SURVEY DESCRIPTION," *The Astronomical Journal*, vol. 149, no. 2, p. 51, Jan. 2015, doi: 10.1088/0004-6256/149/2/51.

[3] E. J. Candes and M. B. Wakin, "An Introduction To Compressive Sampling," *IEEE Signal Processing Magazine*, vol. 25, no. 2, pp. 21–30, 2008, doi: 10.1109/MSP.2007.914731.

[4] T. J. Cornwell and K. Evans, "A simple maximum entropy deconvolution algorithm," *Astronomy and Astrophysics*, vol. 143, pp. 77–83, 1985.

[5] T. J. Cornwell, "Multiscale CLEAN deconvolution of radio synthesis images," *IEEE Journal of selected topics in signal processing*, vol. 2, no. 5, pp. 793–801, 2008.

[6] M. Cuturi, "Sinkhorn distances: Lightspeed computation of optimal transport," *Advances in neural information processing systems*, vol. 26, pp. 2292–2300, 2013.

[7] J. Freeman and M. Rawson, "Top-K Ranking Deep Contextual Bandits for Information Selection Systems," in *2021 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, Melbourne, Australia, Oct. 2021, pp. 2209–2214. doi: 10.1109/SMC52423.2021.9658912.

[8] C. Frogner, C. Zhang, H. Mobahi, M. Araya-Polo, and T. A. Poggio, "Learning with a Wasserstein Loss," in *Advances in Neural Information Processing Systems*, 2015, vol. 28. [Online].

[9] K. Grasha, D. Calzetti, A. Adamo, R. Kennicutt, B. Elmegreen, M. Messa, D. Dale, et al., "The spatial relation between young star clusters and molecular clouds in M51 with LEGUS," *Monthly Notices of the Royal Astronomical Society*, vol. 483, no. 4, pp. 4707–4723, 2019, doi: 10.1093/mnras/sty3424.

[10] S. F. Gull and G. J. Daniell, "Image reconstruction from incomplete and noisy data," *Nature*, vol. 272, no. 5655, pp. 686–690, 1978.

[11] J. Högbom, "Aperture synthesis with a non-regular distribution of interferometer baselines," *Astronomy and Astrophysics Supplement Series*, vol. 15, p. 417, 1974.

[12] J. Lellmann, D. A. Lorenz, C. Schönlieb, and T. Valkonen, "Imaging with Kantorovich-Rubinstein discrepancy," *SIAM J. Imaging Sci.*, vol. 7, no. 4, pp. 2833–2859, Jan. 2014, doi: 10.1137/140975528.

[13] F. Li, T. J. Cornwell, and F. de Hoog, "The application of compressive sampling to radio astronomy: I. Deconvolution," *Astronomy & Astrophysics*, vol. 528, p. A31, Apr. 2011, doi: 10.1051/0004-6361/201015045.

[14] M. K. Ng, H. Shen, E. Y. Lam, and L. Zhang, "A total variation regularization based super-resolution reconstruction algorithm for digital video," *EURASIP Journal on Advances in Signal Processing*, vol. 2007, pp. 1–16, 2007.

[15] G. Ongie, A. Jalal, C. A. Metzler, R. G. Baraniuk, A. G. Dimakis, and R. Willett, "Deep Learning Techniques for Inverse Problems in Imaging," *IEEE Journal on Selected Areas in Information Theory*, vol. 1, no. 1, pp. 39–56, 2020, doi: 10.1109/JSAIT.2020.2991563.

[16] G. Pérez, M. Messa, D. Calzetti, S. Maji, D. E. Jung, A. Adamo, and M. Sirressi, "StarcNet: Machine Learning for Star Cluster Identification," *The Astrophysical Journal*, vol. 907, no. 2, p. 100, Feb. 2021.

[17] G. Peyré and M. Cuturi, "Computational Optimal Transport: With Applications to Data Science," *Foundations and Trends in Machine Learning*, vol. 11, no. 5–6, pp. 355–607, 2019, doi: 10.1561/2200000073.

[18] M. Rawson and R. Balan, "Convergence Guarantees for Deep Epsilon Greedy Policy Learning," arXiv:2112.03376, Dec. 2021.

[19] M. Robinson and C. Capraro, "Super-resolving star clusters with sheaves," arXiv:2106.08123, Jun. 2021.

[20] C. Villani, *Topics in optimal transportation*, vol. 58. Providence, RI: American Mathematical Society, 2003.

[21] W. Wei, E. Huerta, B. C. Whitmore, J. C. Lee, S. Hannon, R. Chandar, D. A. Dale, et al., "Deep transfer learning for star cluster classification: I. application to the PHANGS–HST survey," *Monthly Notices of the Royal Astronomical Society*, vol. 493, no. 3, pp. 3178–3193, 2020.

[22] D. Zou, R. Balan, and M. Singh, "On Lipschitz Bounds of General Convolutional Neural Networks," *IEEE Transactions on Information Theory*, vol. 66, no. 3, pp. 1738–1759, 2019.