Wasserstein Smoothing: Certified Robustness against Wasserstein Adversarial Attacks

Alexander Levine and Soheil Feizi
Department of Computer Science, University of Maryland, College Park
{alevine0, sfeizi}@cs.umd.edu

Abstract

In the last couple of years, several adversarial attack methods based on different threat models have been proposed for the image classification problem. Most existing defenses consider additive threat models in which sample perturbations have bounded L_p norms. These defenses, however, can be vulnerable against adversarial attacks under non-additive threat models. An example of an attack method based on a non-additive threat model is the Wasserstein adversarial attack proposed by Wong et al. (2019), where the distance between an image and its adversarial example is determined by the Wasserstein metric ("earth-mover distance") between their normalized pixel intensities. Until now, there has been no certifiable defense against this type of attack. In this work, we propose the first defense with certified robustness against Wasserstein adversarial attacks using randomized smoothing. We develop this certificate by considering the space of possible flows between images, and representing this space such that Wasserstein distance between images is upper-bounded by L_1 distance in this flow-space. We can then apply existing randomized smoothing certificates for the L_1 metric. In MNIST and CIFAR-10 datasets, we find that our proposed defense is also practically effective, demonstrating significantly improved accuracy under Wasserstein adversarial attack compared to unprotected models.

1 Introduction

In recent years, adversarial attacks against machine learning systems, and defenses against these attacks, have been heavily studied (Szegedy et al., 2013; Madry et al., 2017; Carlini and Wagner, 2017). Although these attacks have been applied in a variety of domains, image classification tasks remain a major focus of research. In general, for a specified image classifier $\boldsymbol{f}$, the goal of an adversarial attack on an image $\boldsymbol{x}$ is to produce a perturbed image $\tilde{\boldsymbol{x}}$ that is imperceptibly 'close' to $\boldsymbol{x}$, such that $\boldsymbol{f}$ classifies $\tilde{\boldsymbol{x}}$ differently than $\boldsymbol{x}$. This 'closeness' notion can be measured in a variety of different ways under different threat models. Most existing attacks and defenses consider additive threat models where the L_p norm of $\tilde{\boldsymbol{x}} - \boldsymbol{x}$ is bounded.

Recently, non-additive threat models (Wong et al., 2019; Laidlaw and Feizi, 2019; Engstrom et al., 2019; Assion et al., 2019) have been introduced which aim to minimize the distance between $\boldsymbol{x}$ and $\tilde{\boldsymbol{x}}$ according to other metrics. Among these attacks is the attack introduced by Wong et al. (2019) which considers the Wasserstein distance between $\boldsymbol{x}$ and $\tilde{\boldsymbol{x}}$, normalized such that the pixel intensities of the image can be treated as probability distributions. Informally, the Wasserstein distance between probability distributions $\boldsymbol{x}$ and $\tilde{\boldsymbol{x}}$ measures the minimum cost to 'transport' probability mass in order to transform $\boldsymbol{x}$ into $\tilde{\boldsymbol{x}}$, where the cost scales with both the amount of mass transported and the distance over which it is transported with respect to some underlying metric. The intuition behind this threat model is that shifting pixel intensity a short distance across an image is less perceptible than moving the same amount of pixel intensity a larger distance (See Figure 1 for an example of a Wasserstein adversarial attack.)

A variety of practical approaches have been proposed to make classifiers robust against adversarial attack, including adversarial training (Madry et al., 2017), defensive distillation (Papernot et al., 2016), and obfuscated gradients (Papernot et al., 2017). However, as

Proceedings of the 23$^{\mathrm{rd}}$International Conference on Artificial Intelligence and Statistics (AISTATS) 2020, Palermo, Italy. PMLR: Volume 108. Copyright 2020 by the author(s).

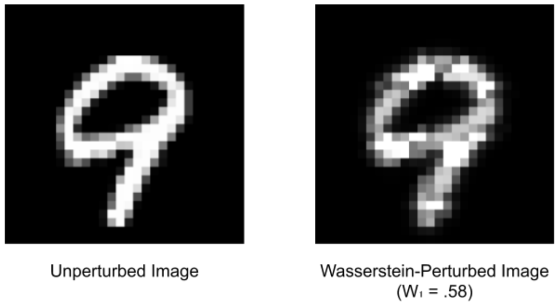


Figure 1: An illustration of Wasserstein adversarial attack (Wong et al., 2019).

new defenses are proposed, new attack methods are often developed which defeat them (Tramèr et al., 2017; Athalye et al., 2018; Carlini and Wagner, 2016). While updated defenses are often then proposed (Tramèr et al., 2017), in general, we cannot be confident that newer attacks will not in turn defeat these defenses.

To escape this cycle, approaches have been proposed to develop certifiably robust classifiers (Wong and Kolter, 2018; Gowal et al., 2018; Lecuyer et al., 2019; Li et al., 2018; Cohen et al., 2019; Salman et al., 2019): in these classifiers, for each image $\boldsymbol{x}$, one can calculate a radius ρ such that it is provably guaranteed that any other image $\tilde{\boldsymbol{x}}$ with distance less than ρ from $\boldsymbol{x}$ will be classified similarly to $\boldsymbol{x}$. This means that no adversarial attack can ever be developed which produces adversarial examples to the classifier within the certified radius.

One effective approach to develop certifiably robust classification is to use randomized smoothing with a probabilistic robustness certificate (Lecuyer et al., 2019; Li et al., 2018; Cohen et al., 2019; Salman et al., 2019). In this approach, one uses a smoothed classifier $\bar{\boldsymbol{f}}(\boldsymbol{x})$, which represents the expectation of $\boldsymbol{f}(\boldsymbol{x})$ over random perturbations of $\boldsymbol{x}$. Based on this smoothing, one can derive an upper bound on how steeply the scores assigned to each class by $\bar{\boldsymbol{f}}$ can change, which can then be used to derive a radius ρ in which the highest class score must remain highest[1].

In this work, we present the first certified defense against Wasserstein adversarial attacks using an adapted randomized smoothing approach, which we call *Wasserstein smoothing.* To develop the robustness certificate, we define a (non-unique) representation of the difference between two images, based on the flow of pixel intensity necessary to construct one image from another. In this representation, we show that the L_1 norm of the minimal flow between two images is equal to the Wasserstein distance between the images. This allows us to apply existing L_1 smoothing-based defenses, by adding noise in the space of these representations of flows. We show empirically that this gives improved robustness certificates, compared to using a weak upper bound on Wasserstein distance given by randomized smoothing in the feature space of images directly. We also show that our Wasserstein smoothing defense protects against Wasserstein adversarial attacks in practice, with significantly improved empirical robustness compared to baseline models. For small adversarial perturbations on the MNIST dataset, our method achieves higher accuracy under adversarial attack than all existing practical defenses for the Wasserstein threat model. In summary, we make the following contributions:

- We develop a novel certified defense for the Wasserstein adversarial attack threat model. This is the first certified defense, to our knowledge, that has been proposed for this threat model.
- We demonstrate that our certificate is nonvacuous, in that it can certify Wasserstein radii larger than those which can be certified by exploiting a trivial L_1 upper bound on Wasserstein distance.
- We demonstrate that our defense effectively protects against existing Wasserstein adversarial attacks, compared to an unprotected baseline.

2 Background

Let $\boldsymbol{x} \in [0,1]^{n\times m}$ denote a two dimensional image, of height n and width m. We will normalize the image such that $\sum_i \sum_j x_{i,j} = 1$, so that $\boldsymbol{x}$ can be interpreted as a probability distribution on the discrete support of pixel coordinates of the 2D image.[2] Also, let $[n]$ denote the set of integers 1 through n, and let $< A, B >$ denote the elementwise inner product between A and B. Following the notation of Wong et al. (2019), we define the p-Wasserstein distance between $\boldsymbol{x}$ and $\boldsymbol{x}'$ as:

Definition 2.1. *Given two distributions $\boldsymbol{x}, \boldsymbol{x}' \in [0,1]^{n\times m}$, and a distance metric $d \in ([n]\times[m])\times([n]\times[m]) \to \mathbb{R}$, the p-Wasserstein distance is:*

$$W_p(\boldsymbol{x}, \boldsymbol{x}') = \min_{\Pi \in \mathbb{R}_+^{(n\cdot m)\times(n\cdot m)}} < \Pi, C >, \tag{1}$$
$$\Pi \mathbf{1} = \boldsymbol{x},\ \Pi^T \mathbf{1} = \boldsymbol{x}',$$
$$C_{(i,j),(i',j')} := \left[d\left((i,j),(i',j')\right)\right]^p .$$

[1] In practice, samples are used to estimate the expectation $\bar{\boldsymbol{f}}(\boldsymbol{x})$, producing an empirical smoothed classifier $\tilde{\boldsymbol{f}}(\boldsymbol{x})$: the certification is therefore probabilistic, with a degree of certainty dependent on the number of samples.

[2] In the case of multi-channel color images, the attack proposed by Wong et al. (2019) does not transport pixel intensity between channels. This allows us to defend against these attacks using our 2D Wasserstein smoothing with little modification. See Section 6.3, and Corollary 2 in the appendix.

Note that $C_{(i,j),(i',j')}$ is the cost of transporting a mass unit from the position (i,j) to (i',j') in the image. For the purpose of matrix multiplication, we are treating $\boldsymbol{x}, \boldsymbol{x}'$ as vectors of length nm. Similarly, the transport plan matrix Π and the cost matrix C are in $\mathbb{R}^{nm \times nm}$.

Intuitively, $\Pi_{(i,j),(i',j')}$ represents the amount of probability mass to be transported from pixel (i,j) to (i',j'), while $C_{(i,j),(i',j')}$ represents the cost per unit probability mass to transport this probability. We can choose $d(.,.)$ to be any measure of distance between pixel positions in an image. For example, in order to represent the L_1 distance metric between pixel positions, we can choose:

$$d\left((i,j),(i',j')\right) = |i-i'| + |j-j'|. \tag{2}$$

Moreover, to represent the L_2 distance metric between pixel positions, we can choose:

$$d\left((i,j),(i',j')\right) = \sqrt{(i-i')^2 + (j-j')^2}. \tag{3}$$

Our defense directly applies to the 1-Wasserstein metric using the L_1 distance as the metric $d(.,.)$, while the attack developed by Wong et al. (2019) uses the L_2 distance. However, because images are two dimensional, these differ by at most a constant factor of $\sqrt{2}$, so we adapt our certificates to the setting of Wong et al. (2019) by simply scaling our certificates by $1/\sqrt{2}$. All experimental results will be presented with this scaling. We emphasize that this it *not* the distinction between 1-Wasserstein and 2-Wasserstein distances: this paper uses the 1-Wasserstein metric, to match the majority of the experimental results of Wong et al. (2019).

To develop our certificate, we rely an alternative linear program formulation for the 1-Wasserstein distance on a two-dimensional image with the L_1 distance metric, provided by Ling and Okada (2007):

$$W_1(\boldsymbol{x}, \boldsymbol{x}') = \min_{\mathbf{g}} \sum_{(i,j)} \sum_{(i',j') \in \mathcal{N}(i,j)} \mathbf{g}_{(i,j),(i',j')} \tag{4}$$

where $\mathbf{g} \geq 0$ and $\forall (i,j)$,

$$\sum_{(i',j') \in \mathcal{N}(i,j)} \mathbf{g}_{(i,j),(i',j')} - \mathbf{g}_{(i',j'),(i,j)} = \mathbf{x}'_{i,j} - \mathbf{x}_{i,j}$$

Here, $\mathcal{N}(i,j)$ denotes the (up to) four immediate (non-diagonal) neighbors of the position (i,j); in other words, $\mathcal{N}(i,j) = \{(i',j') \mid |i-i'| + |j-j'| = 1\}$. For the L_1 distance in two dimensions, Ling and Okada (2007) prove that this formulation is in fact equivalent to the linear program given in Equation 1. Note that only elements of $\mathbf{g}$ with $|i-i'| + |j-j'| = 1$ need to be defined: this means that the number of variables in the linear program is approximately $4nm$, compared to the n^2m^2 elements of Π in Equation 1. While this was originally used to make the linear program more tractable to be solved directly, we exploit the form of this linear program to devise a randomized smoothing scheme in the next section.

3 Robustness Certificate

In order to present our robustness certificate, we first introduce some notation. Let $\boldsymbol{\delta} = \{\boldsymbol{\delta}^{\text{vert.}} \in \mathbb{R}^{(n-1) \times m}, \boldsymbol{\delta}^{\text{horiz.}} \in \mathbb{R}^{n \times (m-1)}\}$ denote a *local flow plan.* It specifies a net flow between adjacent pixels in an image $\boldsymbol{x}$, which, when applied, transforms $\boldsymbol{x}$ to a new image $\boldsymbol{x}'$. See Figure 2 for an explanation of the indexing. For compactness, we write $\boldsymbol{\delta} \in \mathbb{R}^r$ where $r = (n-1)m + n(m-1) \approx 2nm$, and in general refer to the space of possible local flow plans as the *flow domain.* We define the function Δ, which applies a local flow to a distribution.

Definition 3.1. *The local flow plan application function $\Delta \in \mathbb{R}^{n \times m} \times \mathbb{R}^r \to \mathbb{R}^{n \times m}$ is defined as:*

$$\Delta(\boldsymbol{x}, \boldsymbol{\delta})_{i,j} = \boldsymbol{x}_{i,j} + \boldsymbol{\delta}^{vert.}_{i-1,j} - \boldsymbol{\delta}^{vert.}_{i,j} + \boldsymbol{\delta}^{horiz.}_{i,j-1} - \boldsymbol{\delta}^{horiz.}_{i,j} \tag{5}$$

where we let $\boldsymbol{\delta}^{vert.}_{0,j} = \boldsymbol{\delta}^{vert.}_{n,j} = \boldsymbol{\delta}^{horiz.}_{i,0} = \boldsymbol{\delta}^{horiz.}_{i,m} = 0$.[3]

Note that local flow plans are additive:

$$\Delta(\Delta(\boldsymbol{x}, \boldsymbol{\delta}), \boldsymbol{\delta}') = \Delta(\boldsymbol{x}, \boldsymbol{\delta} + \boldsymbol{\delta}') \tag{6}$$

Using this notation, we make a simple transformation of the linear program given in Equation 4, removing the positivity constraint from the variables and reducing the number of variables to $\sim 2nm$:

Lemma 1. *For any normalized probability distributions $\boldsymbol{x}, \boldsymbol{x}' \in [0,1]^{n \times m}$:*

$$W_1(\boldsymbol{x}, \boldsymbol{x}') = \min_{\boldsymbol{\delta}:\, \boldsymbol{x}' = \Delta(\boldsymbol{x}, \boldsymbol{\delta})} \|\boldsymbol{\delta}\|_1 \tag{7}$$

where W_1 denotes the 1-Wasserstein metric, using the L_1 distance as the underlying distance metric d.

Therefore, we can upper-bound the Wasserstein distance between two images using the L_1 norm of *any feasible* local flow plan between them. This enables us to extend existing results for L_1 smoothing-based certificates (Lecuyer et al., 2019) to the Wasserstein metric, by adding noise in the flow domain.

Definition 3.2. *We denote by $\mathcal{L}(\sigma) = \text{Laplace}(0, \sigma)^r$ as the Laplace noise with parameter σ in the flow domain of dimension r.*

[3] Note that the new image $\boldsymbol{x}' = \Delta(\boldsymbol{x}, \boldsymbol{\delta})$ is not necessarily a probability distribution because it may have negative components. However, note that normalization is preserved: $\sum_i \sum_j x'_{i,j} = 1$. This is because every component of $\boldsymbol{\delta}$ is added once and subtracted once to elements in $\boldsymbol{x}$.

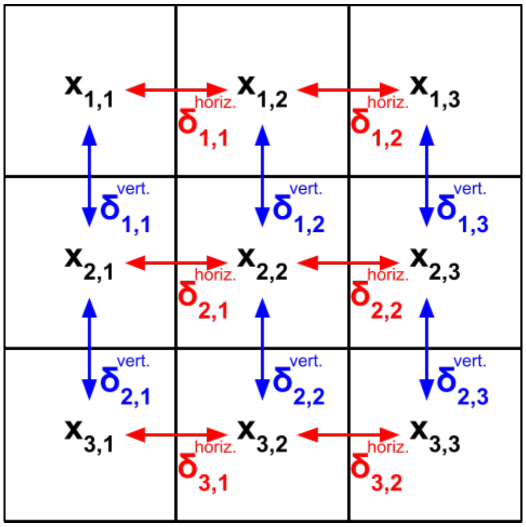


Figure 2: Indexing of the elements of the local flow map $\boldsymbol{\delta}$, in relation to the pixels of the image $\boldsymbol{x}$, with $n = m = 3$.

Given a classification score function $\boldsymbol{f} : \mathbb{R}^{n\times m} \to [0,1]^k$, we define $\bar{\boldsymbol{f}}$ as the *Wasserstein-smoothed* classification function as follows:

$$\bar{\boldsymbol{f}} = \mathop{\mathbb{E}}_{\boldsymbol{\delta}\sim\mathcal{L}(\sigma)} \left[\boldsymbol{f}(\Delta(\boldsymbol{x},\boldsymbol{\delta}))\right]. \tag{8}$$

Let i be the class assignment of $\boldsymbol{x}$ using the Wasserstein-smoothed classifier $\bar{\boldsymbol{f}}$ (in other words, $i = \arg\max_{i'} \bar{\boldsymbol{f}}_{i'}(\boldsymbol{x})$).

Theorem 1. *For any normalized probability distribution $\boldsymbol{x} \in [0,1]^{n\times m}$, if*

$$\bar{\boldsymbol{f}}_i(\boldsymbol{x}) \geq e^{2\sqrt{2}\rho/\sigma} \max_{i'\neq i} \bar{\boldsymbol{f}}_{i'}(\boldsymbol{x}) \tag{9}$$

then for any perturbed probability distribution $\tilde{\boldsymbol{x}}$ such that $W_1(\boldsymbol{x},\tilde{\boldsymbol{x}}) \leq \rho$, we have:

$$\bar{\boldsymbol{f}}_i(\tilde{\boldsymbol{x}}) \geq \max_{i'\neq i} \bar{\boldsymbol{f}}_{i'}(\tilde{\boldsymbol{x}}). \tag{10}$$

All proofs are presented in the appendix.

4 Intuition: One-Dimensional Case

To provide an intuition about the proposed Wasserstein smoothing certified robustness scheme, we consider a simplified model, in which the support of $\boldsymbol{x}$ is a one-dimensional array of length n, rather than a two-dimensional grid (i.e. $\boldsymbol{x} \in \mathbb{R}^n$). In this case, we can denote a local flow plan $\boldsymbol{\delta} \in \mathbb{R}^{n-1}$, so that for $\boldsymbol{x}' = \Delta(\boldsymbol{x},\boldsymbol{\delta})$:

$$\boldsymbol{x}'_i = \boldsymbol{x}_i + \boldsymbol{\delta}_{i-1} - \boldsymbol{\delta}_i \tag{11}$$

where $\boldsymbol{\delta}_0 = \boldsymbol{\delta}_n = 0$. In this one-dimensional case, for any fixed $\boldsymbol{x}, \boldsymbol{x}'$ (with the normalization constraint that $\sum_i x_i = \sum_i x'_i = 1$), there is a unique solution $\boldsymbol{\delta}$ to $\boldsymbol{x}' = \Delta(\boldsymbol{x},\boldsymbol{\delta})$:

$$\boldsymbol{\delta}_i = \sum_{j=1}^{i} x_j - \sum_{j=1}^{i} x'_j \tag{12}$$

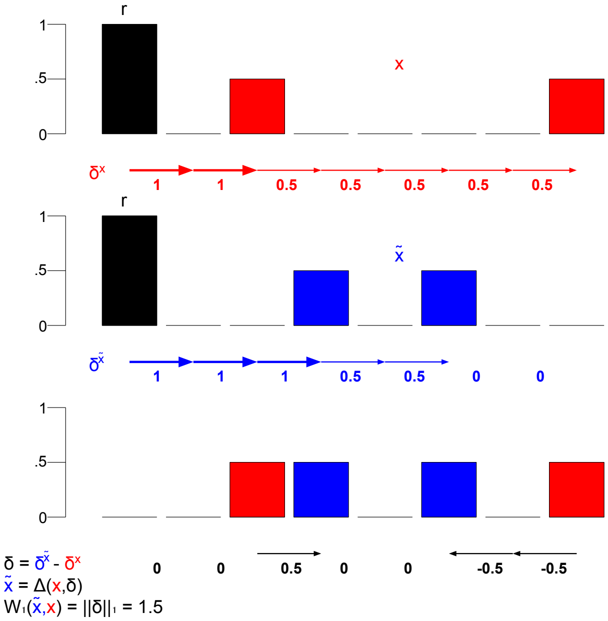


Figure 3: An illustrative example in one dimension. $\boldsymbol{r}$ (black) denotes a fixed reference distribution. With this starting distribution fixed, $\boldsymbol{x}$ (red) and $\tilde{\boldsymbol{x}}$ (blue) can both be uniquely represented in the flow domain as $\boldsymbol{\delta}^{\boldsymbol{x}}$ and $\boldsymbol{\delta}^{\tilde{\boldsymbol{x}}}$. Note that the Wasserstein distance between $\boldsymbol{x}$ and $\tilde{\boldsymbol{x}}$ is then equivalent to the L_1 distance between $\boldsymbol{\delta}^{\boldsymbol{x}}$ and $\boldsymbol{\delta}^{\tilde{\boldsymbol{x}}}$. In the one-dimensional case, this shows that we can transform the samples into a space where the Wasserstein threat model is equivalent to the L_1 metric. We can then use a pre-existing L_1 certified defense in the flow space to defend our classifier.

Note at this reminds us a well-known identity describing optimal transport between two distributions X, Y which share a continuous, one-dimensional support (see Section 2.6 of Peyré et al. (2019), for example):

$$W_1(X,Y) = \int_{-\infty}^{\infty} |F_X(z) - F_Y(z)| dz \tag{13}$$

where F_X, F_Y denote cumulative density functions. If we apply this result to our discretized case, with the index i taking the place of z, and apply the identity to $\boldsymbol{x}$ and $\boldsymbol{x}'$, this becomes:

$$W_1(\boldsymbol{x},\boldsymbol{x}') = \sum_{i=1}^{n} \left| \sum_{j=1}^{i} x_j - \sum_{j=1}^{i} x'_j \right| = \sum_{i=1}^{n} |\boldsymbol{\delta}_i| = \|\boldsymbol{\delta}\|_1 \tag{14}$$

By the uniqueness of the solution given in Equation 12, for any $\boldsymbol{x}$, we can define $\boldsymbol{\delta}^{\boldsymbol{x}}$ as the solution to

$\boldsymbol{x} = \Delta(\boldsymbol{r}, \boldsymbol{\delta})$, where $\boldsymbol{r}$ is an arbitrary fixed reference distribution (e.g. suppose $r_1 = 1, r_i = 0$ for $i \neq 1$). Therefore, instead of operating on the images $\boldsymbol{x}, \tilde{\boldsymbol{x}} \in \mathbb{R}^n$ directly, we can equivalently operate on $\boldsymbol{\delta}^{\boldsymbol{x}}$ and $\boldsymbol{\delta}^{\tilde{\boldsymbol{x}}}$ in the flow domain instead. We will therefore define a flow-domain version of our classifier $\boldsymbol{f}$:

$$\boldsymbol{f}^{\text{flow}}(\boldsymbol{\delta}) := \boldsymbol{f}(\Delta(\boldsymbol{r}, \boldsymbol{\delta})). \tag{15}$$

We will now perform classification entirely in the flow-domain, by first calculating $\boldsymbol{\delta}^{\boldsymbol{x}}$ and then using $\boldsymbol{f}^{\text{flow}}(\boldsymbol{\delta}^{\boldsymbol{x}})$ as our classifier. Now, consider $\boldsymbol{x}$ and an adversarial perturbation $\tilde{\boldsymbol{x}}$, and let $\boldsymbol{\delta}$ be the unique solution to $\tilde{\boldsymbol{x}} = \Delta(\boldsymbol{x}, \boldsymbol{\delta})$. By Equation 14, $\|\boldsymbol{\delta}\|_1 = W_1(\boldsymbol{x}, \tilde{\boldsymbol{x}})$. Then:

$$\tilde{\boldsymbol{x}} = \Delta(\boldsymbol{x}, \boldsymbol{\delta}) = \Delta(\Delta(\boldsymbol{r}, \boldsymbol{\delta}^{\boldsymbol{x}}), \boldsymbol{\delta}) = \Delta(\boldsymbol{r}, \boldsymbol{\delta}^{\boldsymbol{x}} + \boldsymbol{\delta}) \tag{16}$$

where the second equality is by Equation 6. Moreover, by the uniqueness of Equation 12, $\boldsymbol{\delta}^{\tilde{\boldsymbol{x}}} = \boldsymbol{\delta}^{\boldsymbol{x}} + \boldsymbol{\delta}$, or $\boldsymbol{\delta}^{\tilde{\boldsymbol{x}}} - \boldsymbol{\delta}^{\boldsymbol{x}} = \boldsymbol{\delta}$. Therefore

$$\|\boldsymbol{\delta}^{\tilde{\boldsymbol{x}}} - \boldsymbol{\delta}^{\boldsymbol{x}}\|_1 = W_1(\boldsymbol{x}, \tilde{\boldsymbol{x}}). \tag{17}$$

In other words, if we classify in the flow-domain, using $\boldsymbol{f}^{\text{flow}}$, the L_1 distance between point $\boldsymbol{\delta}^{\boldsymbol{x}}, \boldsymbol{\delta}^{\tilde{\boldsymbol{x}}}$ is the Wasserstein distance between the distributions $\boldsymbol{x}$ and $\tilde{\boldsymbol{x}}$. Then, we can perform smoothing in the flow-domain, and use the existing L_1 robustness certificate provided by Lecuyer et al. (2019), to certify robustness. Extending this argument to two-dimensional images adds some complication: images can no longer be represented uniquely in the flow domain, and the relationship between L_1 distance and the Wasserstein distance is now an upper bound. Nevertheless, the same conclusion still holds for 2D images as we state in Theorem 1. Proofs for the two-dimensional case are given in the appendix.

5 Practical Certification Scheme

To generate probabilistic robustness certificates from randomly sampled evaluations of the base classifier $\boldsymbol{f}$, we adapt the procedure outlined by Cohen et al. (2019) for L_2 certificates. We consider a *hard smoothed classifier* approach: we set $\boldsymbol{f}_j(\mathbf{x}) = 1$ if the base classifier selects class j at point $\mathbf{x}$, and $\boldsymbol{f}_j(\mathbf{x}) = 0$ otherwise. We also use a stricter form of the condition given as Equation 9:

$$\bar{\boldsymbol{f}}_i(\boldsymbol{x}) \geq e^{2\sqrt{2}\rho/\sigma}(1 - \bar{\boldsymbol{f}}_i(\boldsymbol{x})) \tag{18}$$

This means that we only need to provide a probabilistic lower bound of the expectation of the largest class score, rather than bounding every class score. This reduces the number of samples necessary to estimate a high-confidence lower bound on $\bar{\boldsymbol{f}}_i(\boldsymbol{x})$, and therefore to estimate the certificate with high confidence. Cohen et al. (2019) provides a statistically sound procedure for this, which we use: refer to that paper for details. Note that, when simply evaluating the classification given by $\bar{\boldsymbol{f}}(\mathbf{x})$, we will also need to approximate $\bar{\boldsymbol{f}}$ using random samples. Cohen et al. (2019) also provides a method to do this which yields the expected classification with high confidence, but may abstain from classifying. We will also use this method when evaluating accuracies.

Since the Wasserstein adversarial attack introduced by Wong et al. (2019) uses the L_2 distance metric, to have a fair performance evaluation against this attack, we are interested in certifying a radius in the 1-Wasserstein distance with underlying L_2 distance metric, rather than L_1. Let us denote this radius as ρ_2. In two-dimensional images, the elements of the cost matrix C in this metric may be smaller by up to a factor of $\sqrt{2}$, so we have:

$$\rho_2 \geq \frac{1}{\sqrt{2}}\rho \tag{19}$$

Therefore, by certifying to a radius of $\rho = \sqrt{2}\rho_2$, we can effectively certify against the L_2 metric 1-Wasserstein attacks of radius ρ_2. (We provide a more formal proof of this claim as Corollary 3 in the appendix.) Our condition then becomes:

$$\bar{\boldsymbol{f}}_i(\boldsymbol{x}) \geq e^{4\rho_2/\sigma}(1 - \bar{\boldsymbol{f}}_i(\boldsymbol{x})). \tag{20}$$

6 Experimental Results

In all experiments, we use 10,000 random noised samples to predict the smoothed classification of each image; to generate certificates, we first use 1000 samples to infer which class has highest smoothed score, and then 10,000 samples to lower-bound this score. All probabilistic certificates and classifications are reported to 95% confidence. The model architectures used for the base classifiers for each data set are the same as used in Wong et al. (2019). When reporting results, *median certified accuracy* refers to the maximum radius ρ_2 such that at least 50% of classifications for images in the data set are certified to be robust to at least this radius, and these certificates are for the correct ground truth class. If over 50% of images are not certified for the correct class, this statistic is reported as N/A.

6.1 Comparison to naive Laplace Smoothing

Note that one can derive a trivial but sometimes tight bound, that, under any L_p distance metric, if

Table 1 Certified Wasserstein Accuracy of Wasserstein and Laplace smoothing on MNIST

Noise standard deviation σ	**Wasserstein Smoothing** Classification accuracy (Percent abstained)	**Wasserstein Smoothing** Median certified robustness	**Wasserstein Smoothing** Base Classifier Accuracy
0.005	98.71(00.04)	0.0101	97.94
0.01	97.98(00.19)	0.0132	94.95
0.02	93.99(00.58)	0.0095	79.72
0.05	74.22(03.95)	0	43.67
0.1	49.41(01.29)	0	30.26
0.2	31.80(08.40)	N/A	25.13
0.5	22.58(00.84)	N/A	22.67
Noise standard deviation σ	**Laplace Smoothing** Classification accuracy (Percent abstained)	**Laplace Smoothing** Median certified robustness	**Laplace Smoothing** Base Classifier Accuracy
0.005	98.87(00.06)	0.0062	97.47
0.01	97.44(00.19)	0.0053	89.32
0.02	91.11(01.29)	0.0030	67.08
0.05	61.44(07.45)	0	33.80
0.1	34.92(09.36)	N/A	25.56
0.2	24.02(05.67)	N/A	22.85
0.5	22.57(01.05)	N/A	22.70

$W_1(\boldsymbol{x}, \tilde{\boldsymbol{x}}) \leq \rho/2$, then $\|\boldsymbol{x} - \tilde{\boldsymbol{x}}\|_1 \leq \rho$. (See Corollary 1 in the appendix.) This enables us to write a condition for ρ_2-radius Wasserstein certified robustness by applying Laplace smoothing directly, and simply converting the certificate. In our notation, this condition is:

$$\bar{\boldsymbol{f}}_i^{\text{Laplace}}(\boldsymbol{x}) \geq e^{4\sqrt{2}\rho_2/\sigma}(1 - \bar{\boldsymbol{f}}_i^{\text{Laplace}}(\boldsymbol{x})) \tag{21}$$

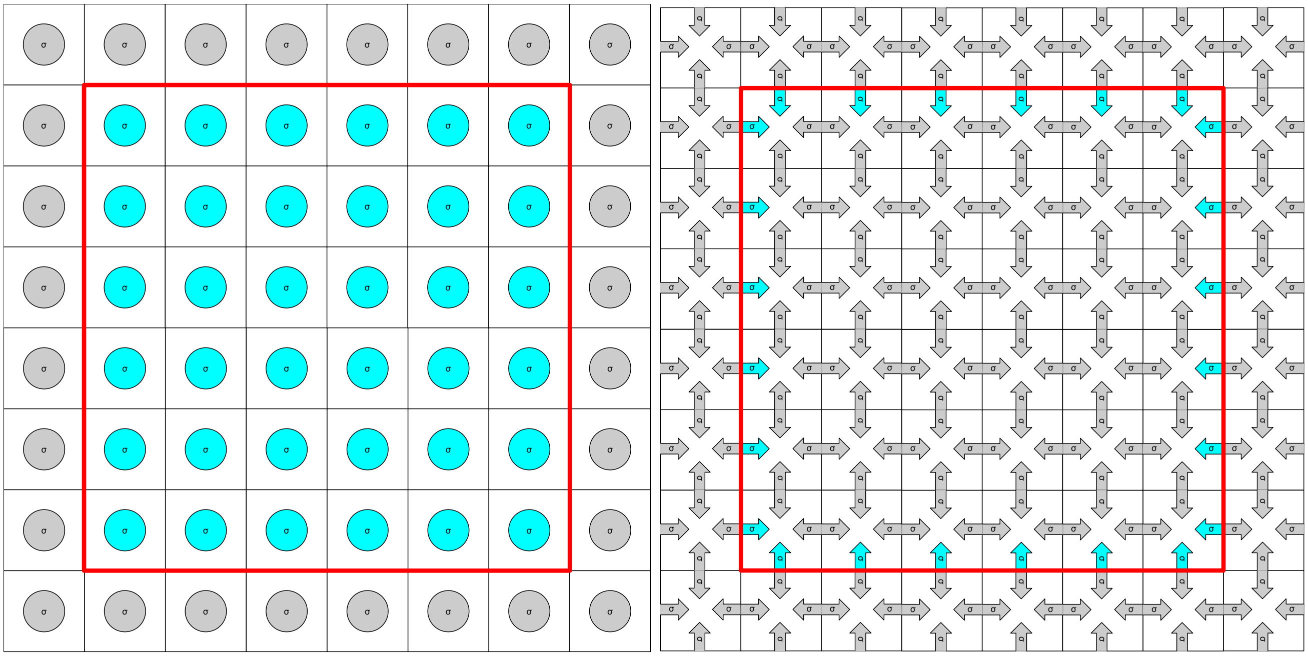
where $\bar{\boldsymbol{f}}^{\text{Laplace}}(\boldsymbol{x})$ is a smoothed classifier with Laplace noise added to every pixel independently. It may appear as if our Wasserstein-smoothed bound should only be an improvement over this bound by a factor of $\sqrt{2}$ in the certified radius ρ_2. However, as shown in Table 1, we in fact improve our certificates by a larger factor. This is because, for a fixed noise standard deviation, the base classifier is able to achieve a higher accuracy after adding noise in the flow-domain, compared to adding noise directly to the pixels. When adding noise in the flow-domain, we add and subtract noise in equal amounts between adjacent pixels, preserving more information for the base classifier.

To give a concrete example, consider some $k \times k$ square patch of an image. Suppose that the overall aggregate pixel intensity in this patch (i.e. the sum of the pixel values) is a salient feature for classification (This is a highly plausible situation: for example, in MNIST, this may indicate whether or not some region of an image is occupied by part of a digit.) Let us call this feature μ, and calculate the variance of μ in smoothing samples under Laplace and Wasserstein smoothing, both with variance σ^2. Under Laplace smoothing (Figure 4-a), k^2 independent instances of Laplace noise are added to μ, so the resulting variance will be $k^2\sigma^2$: this is proportional to the area of the region. In the case of Wasserstein smoothing, by contrast, probability mass exchanged between between pixels in the interior of the patch has no effect on the aggregate quantity μ. Instead, only noise on the perimeter will affect the total feature value μ: the variance is therefore $4k\sigma^2$ (Figure 4-b). Wasserstein smoothing then reduces the effective noise variance on the feature μ by a factor of $O(k)$.

6.2 Empirical adversarial accuracy

We measure the performance of our smoothed classifier against the Wasserstein-metric adversarial attack proposed in Wong et al. (2019), and compare to models tested in that work. Results are presented in Figure 5. For testing, we use the same attack parameters as in Wong et al. (2019): the "Standard" and "Adversarial Training" results are therefore replications of the experiments from that paper, using the publicly available code and pretrained models.

In order to attack our hard smoothed classifier, we adapt the method proposed by Salman et al. (2019): in particular, note that we cannot directly calculate the gradient of the classification loss with respect to the image for a *hard* smoothed classifier, because the derivatives of the logits of the base classifier are not propagated. Therefore, we must instead attack a *soft* smooth classifier: we take the expectation over samples of the *softmaxed* logits of the base classifier, instead of the final classification output. In each step of



(a) Laplace Smoothing

(b) Wasserstein Smoothing

Figure 4: Schematic diagram showing the difference between Laplace and Wasserstein smoothing on the variance of the aggregate pixel intensity in a square region, outlined in red. See the text of Section 6.1. In both figures, pixels are represented as square tiles. In (a), noise on individual pixels is represented with circles, which are gray if they do *not* contribute to the overall pixel intensity in the outlined region, but are cyan if they *do* contribute. We see that the noise is proportional (in variance) to the area of the region. In (b), under Wasserstein smoothing, noise is represented by arrows between pixels which exchange intensity. Again, these are gray if they do not contribute to the overall pixel intensity in the outlined region, and cyan if they do contribute. Note that arrows in the interior do not contribute to the aggregate intensity, because equal values are added and subtracted from adjacent pixels. The noise is proportional (in variance) to the perimeter of the region. This provides a plausible intuition as to why base classifiers, when given noisy images, classify with higher accuracy on Wasserstein smoothed images compared to Laplace smoothed images, as seen empirically in Table 1.

the attack, we use 128 noised samples to estimate this gradient, as used in Salman et al. (2019).

In the attack proposed by Wong et al. (2019), the images are attacked over 200 iterations of projected gradient descent, projected onto a Wasserstein ball, with the radius of the ball every 10 iterations. The attack succeeds, and the final radius is recorded, once the classifier misclassifies the image. In order to preserve as much of the structure (and code) of the attack as possible to provide a fair comparison, it is thus necessary for us to evaluate each image using our hard classifier, with the full 10,000 smoothing samples, at each iteration of the attack. We count the classifier abstaining as a misclassification for these experiments. However, note that this may somewhat underestimate the true robustness of our classifier: recall that our classifier is nondeterministic; therefore, because we are repeatedly evaluating the classifier and reporting a perturbed image as adversarial the first time it is missclassified, we may tend to over-count misclassifications. However, because we are using a large number of noise samples to generate our classifications, this is only likely to happen with examples which are close to being adversarial. Still, the presented data should be regarded as a lower bound on the true accuracy under attack of our Wasserstein smoothed classifier.

In Figure 5, we note two things: first, our Wasserstein smoothing technique appears to be an effective empirical defense against Wasserstein adversarial attacks, compared to an unprotected ('Standard') network. (It is also more robust than the binarized and L_∞-robust models tested by Wong et al. (2019): see appendix.) However, for large perturbations, our defense is less effective than the adversarial training defense proposed by Wong et al. (2019). This suggests a promising direction for future work: Salman et al. (2019) proposed an adversarial training method for smoothed classifiers, which could be applied in this case. Note however that both Wasserstein adversarial attacks and smoothed adversarial training are com-

Table 2 Certified Wasserstein Accuracy of Wasserstein smoothing on CIFAR10

Noise standard deviation σ	Classification accuracy (Percent abstained)	Median certified robustness	Base Classifier Accuracy
0.00005	87.01(00.24)	0.000101	86.02
0.0001	83.39(00.42)	0.000179	82.08
0.0002	77.57(00.66)	0.000223	75.46
0.0005	68.75(01.01)	0.000209	65.12
0.001	61.65(01.77)	0.000127	57.03

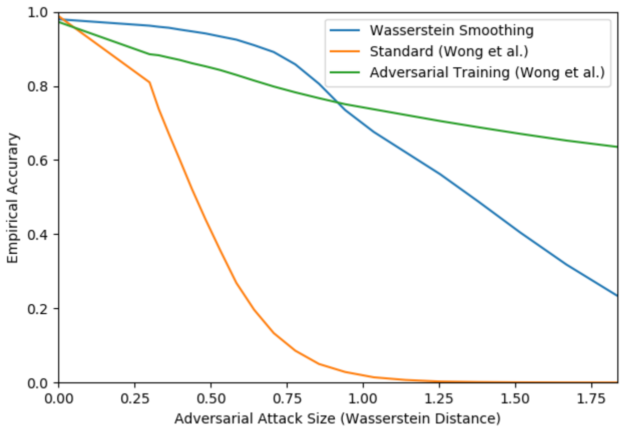


Figure 5: Comparison of empirical robustness on MNIST to models from (Wong et al., 2019). Wasserstein smoothing is with $\sigma = 0.01$. (This is the amount of noise which maximizes certified robustness, as seen in Table 1.)

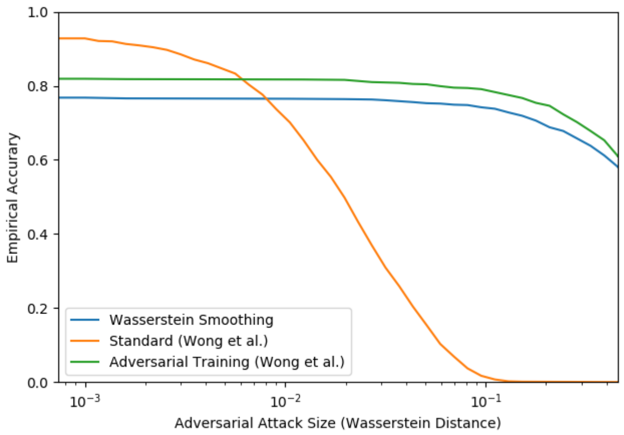


Figure 6: Comparison of empirical robustness on CIFAR-10 to models from (Wong et al., 2019). Wasserstein smoothing is with $\sigma = 0.0002$. (This is the amount of noise which maximizes certified robustness, as seen in Table 2.) Note that we test on a random sample of 1000 images from CIFAR-10, rather than the entire data set.

putationally expensive, so this may require significant computational resources.

Second, the median radius of attack to which our smoothed classifier is empirically robust is larger than the median certified robustness of our smoothed classifier by two orders of magnitude. This calls for future work both to develop improved robustness certificates as well as to develop more effective attacks in the Wasserstein metric.

6.3 Experiments on color images (CIFAR-10)

Wong et al. (2019) also apply their attack to color images in CIFAR-10. In this case, the attack does not transport probability mass between color channels: therefore, in our defense, it is sufficient to add noise in the flow domain to each channel independently to certify robustness (See Corollary 2 in the appendix for a proof of the validity of this method). Certificates are presented in Table 2, while empirical robustness is presented in Figure 6. Again, we compare directly to models from Wong et al. (2019). We note that again, empirically, our model significantly outperforms an unprotected model, but is not as robust as a model trained adversarially. We also note that the certified robustness is orders of magnitude smaller than computed for MNIST: however, the unprotected model is also significantly less robust empirically than the equivalent MNIST model.

7 Conclusion

In this paper, we developed a smoothing-based certifiably robust defense for Wasserstein-metric adversarial examples. To do this, we add noise in the space of possible flows of pixel intensity between images. To our knowledge, this is the first certified defense method specifically tailored to the Wasserstein threat model. Our method proves to be an effective practical defense against Wasserstein adversarial attacks, with significantly improved empirical adversarial robustness compared to a baseline model.

8 Acknowledgements

This project was supported in part by NSF CAREER AWARD 1942230, HR 00111990077, HR 001119S0026-GARD-FP-052, and Simons Fellowship on "Foundations of Deep Learning."

References

Assion, F., Schlicht, P., Greßner, F., Gunther, W., Huger, F., Schmidt, N., and Rasheed, U. (2019). The attack generator: A systematic approach towards constructing adversarial attacks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0.

Athalye, A., Carlini, N., and Wagner, D. (2018). Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In *International Conference on Machine Learning*, pages 274–283.

Carlini, N. and Wagner, D. (2016). Defensive distillation is not robust to adversarial examples. *arXiv preprint arXiv:1607.04311*.

Carlini, N. and Wagner, D. (2017). Towards evaluating the robustness of neural networks. In *2017 IEEE Symposium on Security and Privacy (SP)*, pages 39–57. IEEE.

Cohen, J. M., Rosenfeld, E., and Kolter, J. Z. (2019). Certified adversarial robustness via randomized smoothing. *arXiv preprint arXiv:1902.02918*.

Engstrom, L., Tran, B., Tsipras, D., Schmidt, L., and Madry, A. (2019). Exploring the landscape of spatial robustness. In Chaudhuri, K. and Salakhutdinov, R., editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 1802–1811, Long Beach, California, USA. PMLR.

Gowal, S., Dvijotham, K., Stanforth, R., Bunel, R., Qin, C., Uesato, J., Mann, T., and Kohli, P. (2018). On the effectiveness of interval bound propagation for training verifiably robust models. *arXiv preprint arXiv:1810.12715*.

Laidlaw, C. and Feizi, S. (2019). Functional adversarial attacks. *arXiv preprint arXiv:1906.00001*.

Lecuyer, M., Atlidakis, V., Geambasu, R., Hsu, D., and Jana, S. (2019). Certified robustness to adversarial examples with differential privacy. In *2019 2019 IEEE Symposium on Security and Privacy (SP)*, pages 726–742, Los Alamitos, CA, USA. IEEE Computer Society.

Li, B., Chen, C., Wang, W., and Carin, L. (2018). Second-order adversarial attack and certifiable robustness. *arXiv preprint arXiv:1809.03113*.

Ling, H. and Okada, K. (2007). An efficient earth mover's distance algorithm for robust histogram comparison. *IEEE transactions on pattern analysis and machine intelligence*, 29(5):840–853.

Madry, A., Makelov, A., Schmidt, L., Tsipras, D., and Vladu, A. (2017). Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*.

Papernot, N., McDaniel, P., Goodfellow, I., Jha, S., Celik, Z. B., and Swami, A. (2017). Practical black-box attacks against machine learning. In *Proceedings of the 2017 ACM on Asia conference on computer and communications security*, pages 506–519. ACM.

Papernot, N., McDaniel, P., Wu, X., Jha, S., and Swami, A. (2016). Distillation as a defense to adversarial perturbations against deep neural networks. In *2016 IEEE Symposium on Security and Privacy (SP)*, pages 582–597. IEEE.

Peyré, G., Cuturi, M., et al. (2019). Computational optimal transport. *Foundations and Trends® in Machine Learning*, 11(5-6):355–607.

Salman, H., Yang, G., Li, J., Zhang, P., Zhang, H., Razenshteyn, I., and Bubeck, S. (2019). Provably robust deep learning via adversarially trained smoothed classifiers. *arXiv preprint arXiv:1906.04584*.

Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., and Fergus, R. (2013). Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*.

Tramèr, F., Kurakin, A., Papernot, N., Goodfellow, I., Boneh, D., and McDaniel, P. (2017). Ensemble adversarial training: Attacks and defenses. *arXiv preprint arXiv:1705.07204*.

Wong, E. and Kolter, Z. (2018). Provable defenses against adversarial examples via the convex outer adversarial polytope. In *International Conference on Machine Learning*, pages 5283–5292.

Wong, E., Schmidt, F., and Kolter, Z. (2019). Wasserstein adversarial examples via projected Sinkhorn iterations. In Chaudhuri, K. and Salakhutdinov, R., editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 6808–6817, Long Beach, California, USA. PMLR.