TSS: Transformation-Specific Smoothing for Robustness Certification

Linyi Li*
University of Illinois
Urbana, Illinois, USA
linyi2@illinois.edu

Maurice Weber*
ETH Zürich
Zürich, Switzerland
maurice.weber@inf.ethz.ch

Xiaojun Xu
University of Illinois
Urbana, Illinois, USA
xiaojun3@illinois.edu

Luka Rimanic
ETH Zürich
Zürich, Switzerland
luka.rimanic@inf.ethz.ch

Bhavya Kailkhura
Lawrence Livermore National
Laboratory
Livermore, California, USA
kailkhura1@llnl.gov

Tao Xie
Key Laboratory of High Confidence
Software Technologies, MoE (Peking
University)
Beijing, China
taoxie@pku.edu.cn

Ce Zhang
ETH Zürich
Zürich, Switzerland
ce.zhang@inf.ethz.ch

Bo Li
University of Illinois
Urbana, Illinois, USA
lbo@illinois.edu

ABSTRACT

As machine learning (ML) systems become pervasive, safeguarding their security is critical. However, recently it has been demonstrated that motivated adversaries are able to mislead ML systems by perturbing test data using semantic transformations. While there exists a rich body of research providing provable robustness guarantees for ML models against ℓ_p norm bounded adversarial perturbations, guarantees against semantic perturbations remain largely underexplored. In this paper, we provide **TSS**—*a unified framework for certifying ML robustness against general adversarial semantic transformations*. First, depending on the properties of each transformation, we divide common transformations into two categories, namely *resolvable* (e.g., Gaussian blur) and *differentially resolvable* (e.g., rotation) transformations. For the former, we propose transformation-specific randomized smoothing strategies and obtain strong robustness certification. The latter category covers transformations that involve interpolation errors, and we propose a novel approach based on stratified sampling to certify the robustness. Our framework **TSS** leverages these certification strategies and combines with consistency-enhanced training to provide rigorous certification of robustness. We conduct extensive experiments on over ten types of challenging semantic transformations and show that **TSS** significantly outperforms the state of the art. Moreover, to the best of our knowledge, **TSS** is the first approach that achieves nontrivial certified robustness on the large-scale ImageNet dataset. For instance, our framework achieves 30.4% certified robust accuracy against rotation attack (within ±30°) on ImageNet. Moreover, to consider a broader range of transformations, we show **TSS** is also robust against adaptive attacks and unforeseen image corruptions such as CIFAR-10-C and ImageNet-C.

CCS CONCEPTS

• **General and reference** → **Verification**; • **Security and privacy** → **Logic and verification**; *Software security engineering*; • **Computing methodologies** → *Neural networks*.

KEYWORDS

Certified Robustness, Semantic Transformation Attacks

1 INTRODUCTION

Recent advances in machine learning (ML) have enabled a plethora of applications in tasks such as image recognition [20] and game playing [38, 48]. Despite all of these advances, ML systems are also found exceedingly vulnerable to adversarial attacks: image recognition systems can be adversarially misled [17, 50, 61], and malware detection systems can be evaded easily [54, 64].

The existing practice of security in ML has fallen into the cycle where new empirical defense techniques are proposed [33, 56], followed by new adaptive attacks breaking these defenses [2, 13, 17, 62]. In response, recent research has attempted to provide *provable robustness guarantees* for an ML model. Such certification usually follows the form that the ML model is provably robust against

---

* Equal contribution.

Linyi Li*, Maurice Weber*, Xiaojun Xu, Luka Rimanic, Bhavya Kailkhura, Tao Xie, Ce Zhang, and Bo Li
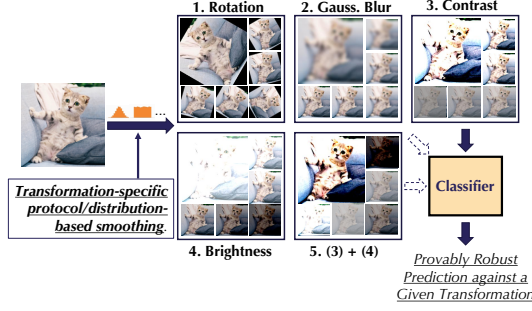

Figure 1: *Transformation-Specific Smoothing-based robustness certification*, a general robustness certification framework against various semantic transformations. We develop a range of different transformation-specific smoothing protocols and various techniques to provide substantially better certified robustness bounds than state-of-the-art approaches on large-scale datasets.

arbitrary adversarial attacks, as long as the perturbation magnitude is below a certain threshold. Different certifiable defenses and robustness verification approaches have provided nontrivial robustness guarantees against ℓ_p perturbations where the perturbation is bounded by small ℓ_p norm [8, 32, 53, 59, 63].

However, certifying robustness only against ℓ_p perturbations is not sufficient for attacks based on semantic transformation. For instance, image rotation, scaling, and other semantic transformations are able to mislead ML models effectively [12, 15, 16, 62]. These transformations are common and practical [5, 22, 40]. For example, it has been shown [24] that brightness/contrast attacks can achieve 91.6% attack success on CIFAR-10, and 71%-100% attack success rate on ImageNet [22]. In practice, brightness- and contrast-based attacks have been demonstrated to be successful in autonomous driving [40, 52]. These attacks incur large ℓ_p-norm differences and are thus beyond the reach of existing certifiable defenses [4, 19, 28, 45]. *Can we provide provable robustness guarantees against these semantic transformations?*

In this paper, we propose theoretical and empirical analyses to certify the ML robustness against a wide range of semantic transformations beyond ℓ_p bounded perturbations. The theoretical analysis is nontrivial given different properties of the transformations, and our empirical results set the new state-of-the-art robustness certification for a range of semantic transformations, exceeding existing work by a large margin. In particular, we propose **T**ransformation-**S**pecific **S**moothing-based robustness certification — a *general framework* based on function smoothing providing certified robustness for ML models against a range of adversarial transformations (Figure 1). To this end, we first categorize semantic transformations as either *resolvable* or *differentially resolvable*. We then provide certified robustness against *resolvable* transformations, which include brightness, contrast, translation, Gaussian blur, and their composition. Second, we develop novel certification techniques for *differentially resolvable* transformations (e.g., rotation and scaling), based on the *building block* that we have developed for resolvable transformations.

For resolvable transformations, we leverage the framework to *jointly* reason about (1) function smoothing under different smoothing distributions and (2) the properties inherent to each specific

transformation. To our best knowledge, this is the first time that the interplay between smoothing distribution and semantic transformation has been analyzed as existing work [8, 30, 65] that studies different smoothing distributions considers only ℓ_p perturbations. Based on this analysis, we find that against certain transformations such as Gaussian blur, exponential distribution is better than Gaussian smoothing, which is commonly used in the ℓ_p-case.

For differentially resolvable transformations, such as rotation, scaling, and their composition with other transformations, the common **challenge** is that they naturally induce *interpolation error*. Existing work [3, 14] can provide robustness guarantees but it cannot rigorously certify robustness for ImageNet-scale data. We develop a collection of novel techniques, including stratified sampling and Lipschitz bound computation to provide a tighter and sound upper bound for the interpolation error. We integrate these novel techniques into our **TSS** framework and further propose a progressive-sampling-based strategy to accelerate the robustness certification. We show that these techniques comprise a scalable and general framework for certifying robustness against differentially resolvable transformations.

We conduct extensive experiments to evaluate the proposed certification framework and show that our framework significantly outperforms the state-of-the-art on different datasets including the large-scale ImageNet against a series of practical semantic transformations. In summary, this paper makes the following <u>contributions</u>:

(1) We propose a *general* function smoothing framework, **TSS**, to certify ML robustness against general semantic transformations.

(2) We categorize common adversarial semantic transformations in the literature into *resolvable* and *differentially resolvable* transformations and show that our framework is general enough to certify both types of transformations.

(3) We theoretically explore different smoothing strategies by sampling from different distributions including non-isotropic Gaussian, uniform, and Laplace distributions. We show that for specific transformations, such as Gaussian blur, smoothing with exponential distribution is better.

(4) We propose a pipeline, **TSS-DR**, including a stratified sampling approach, an effective Lipschitz-based bounding technique, and a progressive sampling strategy to provide rigorous, tight, and scalable robustness certification against differentially resolvable transformations such as rotation and scaling.

(5) We conduct extensive experiments and show that our framework **TSS** can provide significantly higher certified robustness compared with the state-of-the-art approaches, against a range of semantic transformations and their composition on MNIST, CIFAR-10, and ImageNet.

(6) We show that **TSS** also provides much higher empirical robustness against adaptive attacks and unforeseen corruptions such as CIFAR-10-C and ImageNet-C.

The code implementation and all trained models are publicly available at https://github.com/AI-secure/semantic-randomized-smoothing.

2 BACKGROUND

We next provide an overview of different semantic transformations and explain the intuition behind the randomized smoothing [8] that has been proposed to certify the robustness against ℓ_p perturbations.

Semantic Transformation Based Attacks. Beyond adversarial ℓ_p perturbations, a realistic threat model is given by image transformations that preserve the underlying semantics. Examples for these types of transformations include changes to contrast or brightness levels, or rotation of the entire image. These attacks share three common characteristics: (1) The perturbation stemming from a successful semantic attack typically has higher ℓ_p norm compared to ℓ_p-bounded attacks. However, these attacks still preserve the underlying semantics (a car image rotated by $10°$ still contains a car). (2) These attacks are governed by a low-dimensional parameter space. For example, the rotation attack chooses a single-dimensional rotation angle. (3) Some of such adversarial transformations would lead to high interpolation error (e.g., rotation), which makes it challenging to certify. Nevertheless, these types of attacks can also cause significant damage [22, 24] and pose realistic threats for practical ML applications such as autonomous driving [40]. We remark that our proposed framework can be extended to certify robustness against other attacks sharing these characteristics even beyond the image domain, such as GAN-based attacks against ML based malware detection [25, 57], where a limited dimension of features of the malware can be manipulated in order to preserve the malicious functionalities and such perturbation usually incurs large ℓ_p differences for the generated instances.

Randomized Smoothing. On a high level, randomized smoothing [8, 29, 30] provides a way to certify robustness based on randomly perturbing inputs to an ML model. The intuition behind such randomized classifier is that noise smoothens the decision boundaries and suppresses regions with high curvature. Since adversarial examples aim to exploit precisely these high curvature regions, the vulnerability to this type of attack is reduced. Formally, a base classifier h is smoothed by adding noise ε to a test instance. The prediction of the smoothed classifier is then given by the most likely prediction under this smoothing distribution. Subsequently, a tight robustness guarantee can be obtained, based on the noise variance and the class probabilities of the smoothed classifier. It is guaranteed that, as long as the ℓ_2 norm of the perturbation is bounded by a certain amount, the prediction on an adversarial vs. benign input will stay the same. This technique provides a powerful framework to study the robustness of classification models against adversarial attacks for which the primary figure of merit is a low ℓ_p norm with a simultaneously high success rate of fooling the classifier [11, 65]. However, semantic transformations incur large ℓ_p perturbations, which renders classical randomized smoothing infeasible [4, 19, 28], making it of great importance to generalize randomized smoothing to this kind of threat model.

3 THREAT MODEL & TSS OVERVIEW

In this section, we first introduce the notations used throughout this paper. We then define our **threat model**, the **defense goal** and outline the **challenges** for certifying the robustness against semantic transformations. Finally, we will provide a brief **overview** of our **TSS** certification framework.

We denote the space of inputs as $\mathcal{X} \subseteq \mathbb{R}^d$ and the set of labels as $\mathcal{Y} = \{1, \ldots, C\}$ (where $C \geq 2$ is the number of classes). The set of transformation parameters is given by $\mathcal{Z} \subseteq \mathbb{R}^m$ (e.g., rotation angles). We use the notation $\mathbb{P}_X$ to denote the probability measure
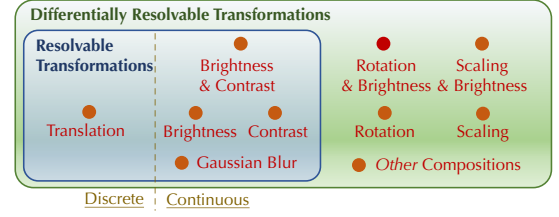

Figure 2: We provide strong robustness certification for both resolvable transformations and differentially revolvable transformations. These two categories cover common adversarial semantic transformations.

induced by the random variable X and write f_X for its probability density function. For a set S, we denote its probability by $\mathbb{P}_X(S)$. A classifier is defined to be a deterministic function h mapping inputs $x \in \mathcal{X}$ to classes $y \in \mathcal{Y}$. Formally, a classifier learns a conditional probability distribution $p(y \mid x)$ over labels and outputs the class that maximizes p, i.e., $h(x) = \arg\max_{y \in \mathcal{Y}} p(y \mid x)$.

3.1 Threat Model and Certification Goal

Semantic Transformations. We model semantic transformations as deterministic functions $\phi \colon \mathcal{X} \times \mathcal{Z} \to \mathcal{X}$, transforming an image $x \in \mathcal{X}$ with a $\mathcal{Z}$-valued parameter α. For example, we use $\phi_R(x, \alpha)$ to model a rotation of the image x by α degrees counterclockwise with bilinear interpolation. We further partition semantic transformations into two different categories, namely resolvable and differentially resolvable transformations. We will show that these two categories could cover commonly known semantic attacks. This categorization depends on whether or not it is possible to write the composition of the transformation ϕ with itself as applying the same transformation just once, but with a different parameter, i.e., whether for any $\alpha, \beta \in \mathcal{Z}$ there exists γ such that $\phi(\phi(x, \alpha), \beta) = \phi(x, \gamma)$. Precise definitions are given in Sections 5 and 6. Figure 2 presents an overview of the transformations considered in this work.

Threat model. We consider an adversary that launches a semantic attack, a type of data evasion attack, against a given classification model h by applying a semantic transformation ϕ with parameter α to an input image $x \to \phi(x, \alpha)$. We allow the attacker to choose an *arbitrary* parameter α within a predefined (attack) parameter space $\mathcal{S}$. For instance, a naïve adversary who randomly changes brightness from within ±40% is able to reduce the accuracy of a state-of-the-art ImageNet classifier from 74.4% to 21.8% (Table 2). While this attack is an example random adversarial attack, our threat model also covers other types of semantic attacks and we provide the first taxonomy for semantic attacks (i.e., resolvable and differentially resolvable) in detail in Sections 5 and 6.

Certification Goal. Since the only degree of freedom that a semantic adversary has is the parameter, **our goal** is to characterize a set of parameters for which the model under attack is guaranteed to be robust. Formally, we wish to find a set $\mathcal{S}_{\text{adv}} \subseteq \mathcal{Z}$ such that, for a classifier h and adversarial transformation ϕ, we have

$$h(x) = h(\phi(x, \alpha)) \quad \forall \alpha \in \mathcal{S}_{\text{adv}}. \tag{1}$$

Challenges for Certifying Semantic Transformations. Certifying ML robustness against semantic transformations is nontrivial and
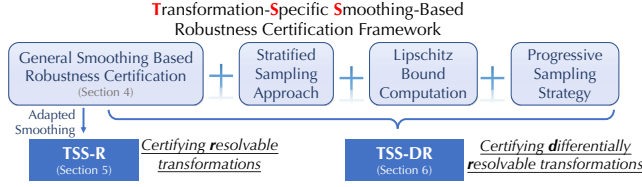
Figure 3: An overview of **TSS**.

requires careful analysis. We identify the following two main challenges that we aim to address in this paper:

(C1) The absolute difference between semantically transformed images in terms of ℓ_p-norms is typically high. This factor causes existing certifiable defenses against ℓ_p bounded perturbations to be inapplicable [4, 19, 28, 45].

(C2) Certain semantic transformations incur additional *interpolation errors*. To derive a robustness certificate, it is required to bound these errors, an endeavour that has been proven to be hard both analytically and computationally. This challenge applies to transformations that involve interpolation, such as rotation and scaling.

We remark that it is in general not feasible to use brute-force approaches such as grid search to enumerate all possible transformation parameters (e.g., rotation angles) since the parameter spaces are typically continuous. Given that different transformations have their own unique properties, it is crucial to provide a *unified* framework that takes into account transformation-specific properties in a general way.

To address these challenges, we generalize randomized smoothing via our proposed *function smoothing framework* to certify arbitrary input transformations via different smoothing distributions, paving the way to robustness certifications that go beyond ℓ_p perturbations. This result addresses challenge **(C1)** in a unified way. Based on this generalization and depending on specific transformation properties, we address challenge **(C2)** and propose a series of smoothing strategies and computing techniques that provide robustness certifications for a diverse range of transformations.

We next introduce our generalized function smoothing framework and show how it can be leveraged to certify semantic transformations. We then categorize transformations as either *resolvable* transformations (Section 5) such as Gaussian blur, or *differentially resolvable* transformations (Section 6) such as rotations.

3.2 Framework Overview

An overview of our proposed framework **TSS** is given in Figure 3. We propose the function smoothing framework, a generalization of randomized smoothing, to provide robustness certifications under general smoothing distributions (Section 4). This generalization enables us to smooth the model on specific transformation dimensions. We then consider two different types of transformation attacks. For *resolvable* transformations, using function smoothing framework, we adapt different smoothing strategies for specific transformations and propose **TSS-R** (Section 5). We show that some smoothing distributions are more suitable for certain transformations. For *differentially resolvable* transformations, to address the interpolation error, we combine function smoothing with the proposed stratified

sampling approach and a novel technique for Lipschitz bound computation to compute a rigorous upper bound of the error. We then develop a progressive sampling strategy to accelerate the certification. This pipeline is termed **TSS-DR**, and we provide details and the theoretical groundwork in Section 6.

4 TSS: TRANSFORMATION SPECIFIC SMOOTHING BASED CERTIFICATION

In this section, we extend randomized smoothing and propose a function smoothing framework **TSS** (**T**ransformation-**S**pecific **S**moothing-based robustness certification) for certifying robustness against semantic transformations. This framework constitutes the main building block for **TSS-R** and **TSS-DR** against specific types of adversarial transformations.

Given an arbitrary base classifier h, we construct a smoothed classifier g by randomly transforming inputs with parameters sampled from a smoothing distribution. Specifically, given an input x, the smoothed classifier g predicts the class that h is most likely to return when the input is perturbed by some random transformation. We formalize this intuition in the following definition.

DEFINITION 1 (ε-SMOOTHED CLASSIFIER). *Let $\phi\colon \mathcal{X} \times \mathcal{Z} \to \mathcal{X}$ be a transformation, $\varepsilon \sim \mathbb{P}_\varepsilon$ a random variable taking values in $\mathcal{Z}$ and let $h\colon \mathcal{X} \to \mathcal{Y}$ be a base classifier. We define the ε-smoothed classifier $g\colon \mathcal{X} \to \mathcal{Y}$ as $g(x; \varepsilon) = \arg\max_{y \in \mathcal{Y}} q(y | x; \varepsilon)$ where q is given by the expectation with respect to the smoothing distribution ε, i.e.,*

$$q(y | x; \varepsilon) := \mathbb{E}(p(y | \phi(x, \varepsilon))). \tag{2}$$

A key to certifying robustness against a specific transformation is the choice of transformation ϕ in the definition of the smoothed classifier (2). For example, if the goal is to certify the Gaussian blur transformation, a reasonable choice is to use the same transformation in the smoothed classifier. However, for other types of transformations this choice does not lead to the desired robustness certificate, and a different approach is required. In Sections 5 and 6, we derive approaches to overcome this challenge and certify robustness against a broader family of semantic transformations.

General Robustness Certification. Given an input $x \in \mathcal{X}$ and a random variable ε taking values in $\mathcal{Z}$, suppose that the base classifier h predicts $\phi(x, \varepsilon)$ to be of class y_A with probability at least p_A and the second most likely class with probability at most p_B (i.e., (4)). Our goal is to derive a robustness certificate for the ε-smoothed classifier g, i.e., we aim to find a set of perturbation parameters $\mathcal{S}_{\text{adv}}$ depending on p_A, p_B, and smoothing parameter ε such that, for all possible perturbation $\alpha \in \mathcal{S}_{\text{adv}}$, it is guaranteed that

$$g(\phi(x, \alpha); \varepsilon) = g(x; \varepsilon) \tag{3}$$

In other words, the prediction of the smoothed classifier can never be changed by applying the transformation ϕ with parameters α that are in the robust set $\mathcal{S}_{\text{adv}}$. The following theorem provides a generic robustness condition that we will subsequently leverage to obtain conditions on transformation parameters. In particular, this result addresses the first challenge **(C1)** for certifying semantic transformations since this result allows to certify robustness beyond additive perturbations.

THEOREM 1. *Let $\varepsilon_0 \sim \mathbb{P}_0$ and $\varepsilon_1 \sim \mathbb{P}_1$ be $\mathcal{Z}$-valued random variables with probability density functions f_0 and f_1 with respect to a*

measure μ on $\mathcal{Z}$ and let $\phi \colon \mathcal{X} \times \mathcal{Z} \to \mathcal{X}$ be a semantic transformation. Suppose that $y_A = g(x; \varepsilon_0)$ and let $p_A, p_B \in [0, 1]$ be bounds to the class probabilities, i.e.,

$$q(y_A | x, \varepsilon_0) \geq p_A > p_B \geq \max_{y \neq y_A} q(y | x, \varepsilon_0). \tag{4}$$

For $t \geq 0$, let $\underline{S}_t, \overline{S}_t \subseteq \mathcal{Z}$ be the sets defined as $\underline{S}_t := \{f_1/f_0 < t\}$ and $\overline{S}_t := \{f_1/f_0 \leq t\}$ and define the function $\xi \colon [0, 1] \to [0, 1]$ by

$$\xi(p) := \sup\{\mathbb{P}_1(S) \colon \underline{S}_{\tau_p} \subseteq S \subseteq \overline{S}_{\tau_p}\}$$
$$\text{where} \quad \tau_p := \inf\{t \geq 0 \colon \mathbb{P}_0(\overline{S}_t) \geq p\}. \tag{5}$$

Then, if the condition

$$\xi(p_A) + \xi(1 - p_B) > 1 \tag{6}$$

is satisfied, then it is guaranteed that $g(x; \varepsilon_1) = g(x; \varepsilon_0)$.

A detailed proof for this statement is provided in Appendix C. At a high level, the condition (4) defines a family of classifiers based on class probabilities obtained from smoothing the input x with the distribution ε_0. Based on the Neyman Pearson Lemma from statistical hypothesis testing, shifting $\varepsilon_0 \to \varepsilon_1$ results in bounds to the class probabilities associated with smoothing x with ε_1. For class y_A, the lower bound is given by $\xi(p_A)$, while for any other class the upper bound is given by $1 - \xi(1 - p_B)$, leading to the the robustness condition $\xi(p_A) > 1 - \xi(1 - p_B)$. It is a more general version of what is proved by Cohen et al. [8], and its generality allows us to analyze a larger family of threat models. Notice that it is not immediately clear how one can obtain the robustness guarantee (3) and deriving such a guarantee from Theorem 1 is nontrivial. We will therefore explain in detail how this result can be instantiated to certify semantic transformations in Sections 5 and 6.

5 TSS-R: Resolveable Transformations

In this section, we define resolvable transformations and then show how Theorem 1 is used to certify this class of semantic transformations. We then proceed to providing a robustness verification strategy for each specific transformation. In addition, we show how the generality of our framework allows us to reason about the best smoothing strategy for a given transformation, which is beyond the reach of related randomized smoothing based approaches [14, 65].

Intuitively, we call a semantic transformation resolvable if we can separate transformation parameters from inputs with a function that acts on parameters and satisfies certain regularity conditions.

Definition 2 (Resolvable transform). A transformation $\phi \colon \mathcal{X} \times \mathcal{Z} \to \mathcal{X}$ is called resolvable if for any $\alpha \in \mathcal{Z}$ there exists a resolving function $\gamma_\alpha \colon \mathcal{Z} \to \mathcal{Z}$ that is injective, continuously differentiable, has non-vanishing Jacobian and for which

$$\phi(\phi(x, \alpha), \beta) = \phi(x, \gamma_\alpha(\beta)) \quad x \in \mathcal{X}, \beta \in \mathcal{Z}. \tag{7}$$

Furthermore, we say that ϕ is additive, if $\gamma_\alpha(\beta) = \alpha + \beta$.

The following result provides a more intuitive view on Theorem 1, expressing the condition on probability distributions as a condition on the transformation parameters.

Corollary 1. Suppose that the transformation ϕ in Theorem 1 is resolvable with resolving function γ_α. Let $\alpha \in \mathcal{Z}$ and set $\varepsilon_1 := \gamma_\alpha(\varepsilon_0)$ in the definition of the function ξ. Then, if α satisfies condition (6), it is guaranteed that $g(\phi(x, \alpha); \varepsilon_0) = g(x; \varepsilon_0)$.

This corollary implies that for resolvable transformations, after we choose the smoothing distribution for the random variable ε_0, we can infer the distribution of $\varepsilon_1 = \gamma_\alpha(\varepsilon_0)$. Then, by plugging in ε_0 and ε_1 into Theorem 1, we can derive an explicit robustness condition from (6) such that for any α satisfying this condition, we can certify the robustness. In particular, for additive transformations we have $\varepsilon_1 = \gamma_\alpha(\varepsilon_0) = \alpha + \varepsilon_0$. For common smoothing distributions ε_0 along with additive transformation, we derive robustness conditions in Appendix D.

In the next subsection, we focus on specific resolvable transformations. For certain transformations, this result can be applied directly. However, for some transformations, e.g., the composition of brightness and contrast, more careful analysis is required. We remark that this corollary also serves a stepping stone to certifying more complex transformations that are in general not resolvable, such as rotations as we will present in Section 6.

5.1 Certifying Specific Transformations

Here we build on our theoretical results from the previous section and provide approaches to certifying a range of different semantic transformations that are resolvable. We state all results here and provide proofs in appendices.

5.1.1 Gaussian Blur. This transformation is widely used in image processing to reduce noise and image detail. Mathematically, applying Gaussian blur amounts to convolving an image with a Gaussian function

$$G_\alpha(k) = \frac{1}{\sqrt{2\pi\alpha}} \exp\left(-k^2/(2\alpha)\right) \tag{8}$$

where $\alpha > 0$ is the *squared* kernel radius. For $x \in \mathcal{X}$, we define Gaussian blur as the transformation $\phi_B \colon \mathcal{X} \times \mathbb{R}_{\geq 0} \to \mathcal{X}$ where

$$\phi_B(x, \alpha) = x * G_\alpha \tag{9}$$

and $*$ denotes the convolution operator. The following lemma shows that Gaussian blur is an *additive transform*. Thus, existing robustness conditions for additive transformations shown in Appendix D are directly applicable.

Lemma 1. *The Gaussian blur transformation is additive, i.e., for any $\alpha, \beta \geq 0$, we have $\phi_B(\phi_B(x, \alpha), \beta) = \phi_B(x, \alpha + \beta)$.*

We notice that the Gaussian blur transformation uses only positive parameters. We therefore consider uniform noise on $[0, a]$ for $a > 0$, folded Gaussians and exponential distribution for smoothing.

5.1.2 Brightness and Contrast. This transformation first changes the brightness of an image by adding a constant value $b \in \mathbb{R}$ to every pixel, and then alters the contrast by multiplying each pixel with a positive factor e^k, for some $k \in \mathbb{R}$. We define the brightness and contrast transformation $\phi_{BC} \colon \mathcal{X} \times \mathbb{R}^2 \to \mathcal{X}$ as

$$(x, \alpha) \mapsto \phi_{BC}(x, \alpha) := e^k(x + b), \quad \alpha = (k, b)^T \tag{10}$$

where $k, b \in \mathbb{R}$ are contrast and brightness parameters, respectively. We remark that ϕ_{BC} is resolvable; however, it is not additive and applying Corollary 1 directly using the resolving function γ_α leads to analytically intractable expressions. On the other hand, if the parameters k and b follow independent Gaussian distributions, we can circumvent this difficulty as follows. Given $\varepsilon_0 \sim \mathcal{N}(0, \text{diag}(\sigma^2, \tau^2))$,

we compute the bounds p_A and p_B to the class probabilities associated with the classifier $g(x; \varepsilon_0)$, i.e., smoothed with ε_0. In the next step, we identify a distribution ε_1 with the property that we can map any lower bound p of $q(y \mid x; \varepsilon_0)$ to a lower bound on $q(y \mid x; \varepsilon_1)$. Using ε_1 as a bridge, we then derive a robustness condition, which is based on Theorem 1, and obtain the guarantee that $g(\phi_{BC}(x, \alpha); \varepsilon_0) = g(x; \varepsilon_0)$ whenever the transformation parameters satisfy this condition. The next lemma shows that the distribution ε_1 with the desired property (lower bound to the classifier smoothed with ε_1) is given by a Gaussian with transformed covariance matrix.

LEMMA 2. *Let $x \in \mathcal{X}$, $k \in \mathbb{R}$, $\varepsilon_0 \sim \mathcal{N}(0, \operatorname{diag}(\sigma^2, \tau^2))$ and $\varepsilon_1 \sim \mathcal{N}(0, \operatorname{diag}(\sigma^2, e^{-2k}\tau^2))$. Suppose that $q(y \mid x; \varepsilon_0) \geq p$ for some $p \in [0, 1]$ and $y \in \mathcal{Y}$. Let Φ be the cumulative density function of the standard Gaussian. Then*

$$q(y \mid x; \varepsilon_1) \geq \begin{cases} 2\Phi\left(e^k \Phi^{-1}\left(\frac{1+p}{2}\right)\right) - 1 & k \leq 0 \\ 2\left(1 - \Phi\left(e^k \Phi^{-1}\left(1 - \frac{p}{2}\right)\right)\right) & k > 0. \end{cases} \quad (11)$$

Now suppose that $g(\cdot; \varepsilon_0)$ makes the prediction y_A at x with probability at least p_A. Then, the preceding lemma tells us that the prediction confidence of $g(\cdot; \varepsilon_1)$ satisfies the lower bound (11) for the same class. Based on these confidence levels, we instantiate Theorem 1 with the random variables ε_1 and $\alpha + \varepsilon_1$ to get a robustness condition.

LEMMA 3. *Let ε_0 and ε_1 be as in Lemma 2 and suppose that*

$$q(y_A \mid x; \varepsilon_1) \geq \tilde{p}_A > \tilde{p}_B \geq \max_{y \neq y_A} q(y \mid x; \varepsilon_1). \quad (12)$$

Then it is guaranteed that $y_A = g(\phi_{BC}(x, \alpha); \varepsilon_0)$ as long as $\alpha = (k, b)^T$ satisfies

$$\sqrt{(k/\sigma)^2 + \left(b/(e^{-k}\tau)\right)^2} < \frac{1}{2}\left(\Phi^{-1}(\tilde{p}_A) - \Phi^{-1}(\tilde{p}_B)\right). \quad (13)$$

In practice, we apply this lemma by replacing $\tilde{p}_A$ and $\tilde{p}_B$ in (13) with the bound computed from (11) based on the class probability bounds p_A and p_B associated with the classifier $g(x; \varepsilon_0)$.

5.1.3   Translation. Let $\bar{\phi}_T : \mathcal{X} \times \mathbb{Z}^2 \to \mathcal{X}$ be the transformation moving an image k_1 pixels to the right and k_2 pixels to the bottom with reflection padding. In order to handle continuous noise distributions, we define the translation transformation $\phi_T : \mathcal{X} \times \mathbb{R}^2 \to \mathcal{X}$ as $\phi_T(x, \alpha) = \bar{\phi}_T(x, [\alpha])$ where $[\cdot]$ denotes rounding to the nearest integer, applied element-wise. We note that ϕ_T is an *additive transform*, allowing us to directly apply Corollary 1 and derive robustness conditions. We note that if we use black padding instead of reflection padding, the transformation is not additive. However, since the number of possible translations is finite, another possibility is to use a simple brute-force approach that can handle black padding, which has already been studied extensively [37, 41].

5.1.4   Composition of Gaussian Blur, Brightness, Contrast, and Translation. Interestingly, the composition of all these four transformations is still resolvable. Thus, we are able to derive the explicit robustness condition for this composition based on Corollary 1, as shown in details in Appendix B. Based on this robustness condition, we compute practically meaningful robustness certificates as we will present in experiments in Section 7.

Table 1: Summary of the Robustness Certification Strategies for Resolvable Transformations.

Transformation	Step 1	Step 2
Gaussian Blur	Compute p_A and p_B with Monte-Carlo Sampling	Check via Corollary 8 (in Appendix D)
Brightness		Check via Corollary 7 (in Appendix D)
Translation		Check via Corollary 7 (in Appendix D)
Brightness and Contrast		Compute $\tilde{p}_A$ via Lemma 2, then check via Lemma 3 (detail in Appendix B.1)
Gaussian Blur, Brightness, Contrast and Translation		Compute $\tilde{p}_A$ via Corollary 3, then check via Lemma B.1 (detail in Appendix B.2)
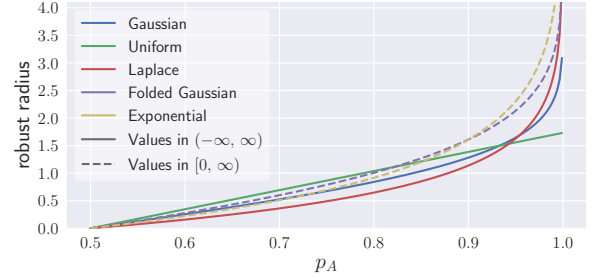
Figure 4: Robust radius comparison for different noise distributions, each with unit variance and dimension.

5.1.5   Robustness Certification Strategies. With these robustness conditions, for a given clean input x, a transformation ϕ, and a set of parameters $\mathcal{S}_{\text{adv}}$, we certify the robustness of the smoothed classifier g with two steps: 1) estimate p_A and p_B (Equation (4)) with Monte-Carlo sampling and high-confidence bound following Cohen et al. [8]; and 2) leverage the robustness conditions to obtain the certificate. A summary for each transformation including the used robustness conditions are shown in Table 1.

5.2   Properties of Smoothing Distributions

The robustness condition in Theorem 1 is generic and leaves a degree of freedom with regards to which smoothing distribution should be used. Previous work mainly provides results for cases in which this distribution is Gaussian [8, 66], while it is nontrivial to extend it to other distributions. Here, we aim to answer this question and provide results for a range of distributions, and discuss their differences. As we will see, *for different scenarios, different distributions behave differently and can certify different radii.* We instantiate Theorem 1 with an arbitrary transformation ϕ and with $\varepsilon_1 := \alpha + \varepsilon_0$ where ε_0 is the smoothing distribution and α is the transformation parameter. The robust radius is then derived by solving condition (6) for α.

Figure 4 illustrates robustness radii associated with different smoothing distributions, each scaled to have unit variance. The bounds are derived in Appendix D and summarized in Table 5. We emphasize that the contribution of this work is not merely these results on different smoothing distributions but, more importantly, the *joint study between different smoothing mechanisms and different semantic transformations.* To compare the different radii for a fixed base classifier, we assume that *the smoothed classifier $g(\cdot; \varepsilon)$ always has the same confidence p_A for noise distributions with equal variance.* Finally, we provide the following conclusions and we will verify them empirically in Section 7.3.1.

(1) *Exponential noise can provide larger robust radius.* We notice that smoothing with exponential noise generally allows for larger adversarial perturbations than other distributions. We also observe that, while all distributions behave similarly for low confidence levels, it is only non-uniform noise distributions that converge toward $+\infty$ when $p_A \to 1$ and exponential noise converges quickest.

(2) *Additional knowledge can lead to larger robust radius.* When we have additional information on the transformation, e.g., all perturbations in Gaussian blur are positive, we can take advantage of this additional information and certify larger radii. For example, under this assumption, we can use folded Gaussian noise for smoothing instead of a standard Gaussian, resulting in a larger radius.

6 TSS-DR: DIFFERENTIALLY RESOLVABLE TRANSFORMATIONS

As we have seen, our proposed function smoothing framework can directly deal with resolvable transformations. However, due to their use of interpolation, some important transformations do not fall into this category, including rotation, scaling, and their composition with resolvable transformations. In this section, we show that they belong to the more general class termed *differentially resolvable transformations* and to address challenge **(C2)**, we propose a novel pipeline **TSS-DR** to provide rigorous robustness certification using our function smoothing framework as a central building block.

Common semantic transformations such as rotations and scaling do not fall into the category of resolvable transformations due to their use of interpolation. To see this issue, consider for example the rotation transformation denoted by ϕ_R. As shown in Figure 5b, despite very similar, the image rotated by $30°$ is different from the image rotated separately by $15°$ and then again by $15°$. The reason is the bilinear interpolation occurring during the rotation. Therefore, if the attacker inputs $\phi_R(x, 15)$, the smoothed classifier in Section 5 outputs

$$g(\phi_R(x, 15); \varepsilon) = \arg\max_{y \in \mathcal{Y}} \mathbb{E}\left(p(y \mid \phi_R(\phi_R(x, 15), \varepsilon))\right), \quad (14)$$

which is a weighted average over the predictions of the base classifier on the randomly perturbed set $\{\phi_R(\phi_R(x, 15), \alpha): \alpha \in \mathcal{Z}\}$. However, in order to use Corollary 1 and to reason about whether this prediction agrees with the prediction on the clean input (i.e., the average prediction on $\{\phi_R(x, \alpha): \alpha \in \mathcal{Z}\}$), we need ϕ_R to be resolvable. As it turns out, this is not the case for transformations that involve interpolation such as rotation and scaling.

To address these issues, we define a transformation ϕ to be *differentially resolvable*, if it can be written in terms of a resolvable transformation ψ and a parameter mapping δ.

DEFINITION 3 (DIFFERENTIALLY RESOLVABLE TRANSFORM). *Let $\phi: X \times \mathcal{Z}_\phi \to X$ be a transformation with noise space $\mathcal{Z}_\phi$ and let $\psi: X \times \mathcal{Z}_\psi \to X$ be a resolvable transformation with noise space $\mathcal{Z}_\psi$. We say that ϕ can be resolved by ψ if for any $x \in X$ there exists function $\delta_x: \mathcal{Z}_\phi \times \mathcal{Z}_\phi \to \mathcal{Z}_\psi$ such that for any $\alpha \in \mathcal{Z}_\phi$ and any $\beta \in \mathcal{Z}_\phi$,*

$$\phi(x, \alpha) = \psi(\phi(x, \beta), \delta_x(\alpha, \beta)). \quad (15)$$
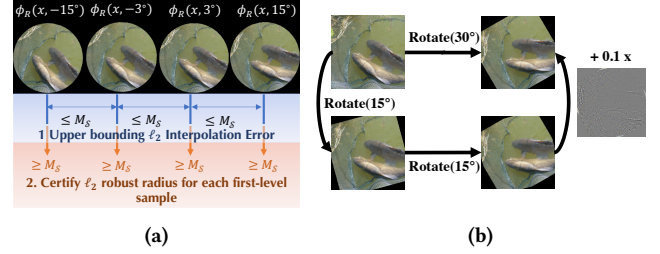

Figure 5: (a) High-level illustration of our robustness certification pipeline **TSS-DR** for differentially resolvable transformations; (b) interpolation error.

This definition leaves open a certain degree of freedom with regard to the choice of resolvable transformation ψ. For example, we can choose the resolvable transformation corresponding to additive noise $(x, \delta) \mapsto \psi(x, \delta) := x + \delta$, which lets us write any transformation ϕ as $\phi(x, \alpha) = \phi(x, \beta) + (\phi(x, \alpha) - \phi(x, \beta)) = \psi(\phi(x, \beta), \delta)$ with $\delta = (\phi(x, \alpha) - \phi(x, \beta))$. In other words, $\phi(x, \alpha)$ can be viewed as first being transformed to $\phi(x, \beta)$ and then to $\phi(x, \beta) + \delta$.

6.1 Overview of TSS-DR

Here, we derive a general robustness certification strategy for differentially resolvable transformations. Suppose that our goal is to certify the robustness against a transformation ϕ that can be resolved by ψ and for transformation parameters from the set $\mathcal{S} \subseteq \mathcal{Z}_\phi$. To that end, we first sample a set of parameters $\{\alpha_i\}_{i=1}^N \subseteq \mathcal{S}$, and transform the input (with those sampled parameters) that yields $\{\phi(x, \alpha_i)\}_{i=1}^N$. In the second step, we compute the class probabilities for each transformed input $\phi(x, \alpha_i)$ with the classifier smoothed with the resolvable transformation ψ. Finally, the intuition is that, if every $\alpha \in \mathcal{S}$ is close enough to one of the sampled parameters, then the classifier is guaranteed to be robust against parameters from the set $\mathcal{S}$. In the next theorem, we show the existence of such a "proximity set" for general δ_x.

THEOREM 2. *Let $\phi: X \times \mathcal{Z}_\phi \to X$ be a transformation that is resolved by $\psi: X \times \mathcal{Z}_\psi \to X$. Let $\varepsilon \sim \mathbb{P}_\varepsilon$ be a $\mathcal{Z}_\psi$-valued random variable and suppose that the smoothed classifier $g: X \to \mathcal{Y}$ given by $q(y \mid x; \varepsilon) = \mathbb{E}(p(y \mid \psi(x, \varepsilon)))$ predicts $g(x; \varepsilon) = y_A = \arg\max_y q(y \mid x; \varepsilon)$. Let $\mathcal{S} \subseteq \mathcal{Z}_\phi$ and $\{\alpha_i\}_{i=1}^N \subseteq \mathcal{S}$ be a set of transformation parameters such that for any i, the class probabilities satisfy*

$$q(y_A \mid \phi(x, \alpha_i); \varepsilon) \ge p_A^{(i)} \ge p_B^{(i)} \ge \max_{y \ne y_A} q(y \mid \phi(x, \alpha_i); \varepsilon). \quad (16)$$

Then there exists a set $\Delta^* \subseteq \mathcal{Z}_\psi$ with the property that, if for any $\alpha \in \mathcal{S}$, $\exists \alpha_i$ with $\delta_x(\alpha, \alpha_i) \in \Delta^*$, then it is guaranteed that

$$q(y_A \mid \phi(x, \alpha); \varepsilon) > \max_{y \ne y_A} q(y \mid \phi(x, \alpha); \varepsilon). \quad (17)$$

In the theorem, the smoothed classifier $g(\cdot; \varepsilon)$ is based on the resolvable transformation ψ that serves as a starting point to certify the target transformation ϕ. To certify ϕ over its parameter space $\mathcal{S}$, we input N transformed samples $\phi(x, \alpha_i)$ to the smoothed classifier $g(\cdot; \varepsilon)$. Then, we get Δ^*, the certified robust parameter set for resolvable transformation ψ. This Δ^* means that for any $\phi(x, \alpha_i)$, if we apply the transformation ψ with any parameter $\delta \in \Delta^*$, the resulting instance $\psi(\phi(x, \alpha_i), \delta)$ is robust for $g(\cdot; \varepsilon)$. Since ϕ is resolvable by ψ, i.e., for any $\alpha \in \mathcal{S}$, there exists an α_i and $\delta \in \Delta^*$ such

that $\phi(x, \alpha) = \psi(\phi(x, \alpha_i), \delta)$, we can assert that for any $\alpha \in \mathcal{S}$, the output of $g(\cdot; \varepsilon)$ on $\phi(x, \alpha)$ is robust.

The key of using this theorem for a specific transformation is to choose the resolvable transformation ψ that can enable a tight calculation of Δ^* under a specific way of sampling $\{\alpha_i\}_{i=1}^N$. First, we observe that a large family of transformations including rotation and scaling can be resolved by the additive transformation $\psi: \mathcal{X} \times \mathcal{X} \to \mathcal{X}$ defined by $(x, \delta) \mapsto x + \delta$. Indeed, any transformation whose pixel value changes are continuous (or with finite discontinuities) with respect to the parameter changes are differentially resolvable—they all can be resolved by the additive transformation. Choosing isotropic Gaussian noise $\varepsilon \sim \mathcal{N}(0, \sigma^2 \mathbb{1}_d)$ as smoothing noise then leads to the condition that the maximum ℓ_2-interpolation error between the interval $\mathcal{S} = [a, b]$ (which is to be certified) and the sampled parameters α_i must be bounded by a radius R. This result is shown in the next corollary, which is derived from Theorem 2.

COROLLARY 2. *Let* $\psi(x, \delta) = x + \delta$ *and let* $\varepsilon \sim \mathcal{N}(0, \sigma^2 \mathbb{1}_d)$. *Furthermore, let* ϕ *be a transformation with parameters in* $\mathcal{Z}_\phi \subseteq \mathbb{R}^m$ *and let* $\mathcal{S} \subseteq \mathcal{Z}_\phi$ *and* $\{\alpha_i\}_{i=1}^N \subseteq \mathcal{S}$. *Let* $y_A \in \mathcal{Y}$ *and suppose that for any* i, *the* ε-*smoothed classifier defined by* $q(y|x;\varepsilon) := \mathbb{E}(p(y|x+\varepsilon))$ *has class probabilities that satisfy*

$$q(y_A| \phi(x, \alpha_i); \varepsilon) \geq p_A^{(i)} \geq p_B^{(i)} \geq \max_{y \neq y_A} q(y| \phi(x, \alpha_i); \varepsilon). \quad (18)$$

Then it is guaranteed that $\forall \alpha \in \mathcal{S}: \ y_A = \arg\max_y q(y| \phi(x, \alpha); \varepsilon)$ *if the maximum interpolation error*

$$M_\mathcal{S} := \max_{\alpha \in \mathcal{S}} \min_{1 \leq i \leq N} \|\phi(x, \alpha) - \phi(x, \alpha_i)\|_2 \quad (19)$$

$$\text{satisfies} \quad M_\mathcal{S} < R := \frac{\sigma}{2} \min_{1 \leq i \leq N} \left(\Phi^{-1}\left(p_A^{(i)}\right) - \Phi^{-1}\left(p_B^{(i)}\right) \right). \quad (20)$$

In a nutshell, this corollary shows that if the smoothed classifier classifies all samples of transformed inputs $\{\phi(x, \alpha_i)\}_{i=1}^N$ consistent with the original input and the smallest gap between confidence levels $p_A^{(i)}$ and $p_B^{(i)}$ is large enough, then it is guaranteed to make the same prediction on transformed inputs $\phi(x, \alpha)$ for *any* $\alpha \in \mathcal{S}$.

The main challenge now lies in computing a tight and scalable upper bound $M \geq M_\mathcal{S}$. Given this bound, a set of transformation parameters $\mathcal{S}$ can then be certified by computing R in (20) and checking that $R > M_\mathcal{S}$. With this methodology, we address challenge **(C2)** and provide means to certify transformations that incur interpolation errors. Figure 5a illustrates this methodology on a high level for the rotation transformation as an example. In the following, we present the general methodology that provides an upper bound of the interpolation error $M_\mathcal{S}$ and provide closed form expressions for rotation and scaling. In Appendix B, we further extend this methodology to certify transformation compositions such as rotation + brightness change + ℓ_2 perturbations.

We remark that dealing with the interpolation error has already been tried before [3, 14]. However, these approaches either leverage explicit linear or interval bound propagation – techniques that are either not scalable or not tight enough. Therefore, on large datasets such as ImageNet, they can provide only limited certification (e.g., against certain random attack instead of any attack).
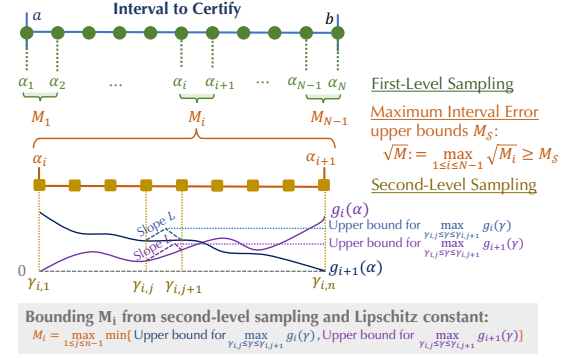
Figure 6: An overview of our interpolation error bounding technique based on stratified sampling and Lipschitz computation.

6.2 Upper Bounding the Interpolation Error

Here, we present the general methodology to compute a rigorous upper bound of the interpolation error introduced in Corollary 2. The methodology presented here is based on stratified sampling and is of a general nature; an explicit computation is shown for the case of rotation and scaling toward the end of this subsection.

Let $\mathcal{S} = [a, b]$ be an interval of transformation parameters that we wish to certify and let $\{\alpha_i\}_{i=1}^N$ be parameters dividing $\mathcal{S}$ uniformly, i.e.,

$$\alpha_i = a + (b - a) \cdot \frac{i - 1}{N - 1}, \quad i = 1, \dots, N. \quad (21)$$

The set of these parameters corresponds to the first-level samples in stratified sampling. With respect to these first-level samples, we define the functions $g_i: [a, b] \to \mathbb{R}_{\geq 0}$ as

$$\alpha \mapsto g_i(\alpha) := \|\phi(x, \alpha) - \phi(x, \alpha_i)\|_2^2 \quad (22)$$

corresponding to the squared ℓ_2 interpolation error between the image x transformed with α and α_i, respectively. For each first-level interval $[\alpha_i, \alpha_{i+1}]$ we look for an upper bound M_i such that

$$M_i \geq \max_{\alpha_i \leq \alpha \leq \alpha_{i+1}} \min\{g_i(\alpha), g_{i+1}(\alpha)\}. \quad (23)$$

It is easy to see that $\max_{1 \leq i \leq N-1} M_i \geq M_\mathcal{S}^2$ and hence setting

$$\sqrt{M} := \max_{1 \leq i \leq N-1} \sqrt{M_i} \quad (24)$$

is a valid upper bound to $M_\mathcal{S}$. The problem has thus reduced to computing the upper bounds M_i associated with each first-level interval $[\alpha_i, \alpha_{i+1}]$. To that end, we now continue with a second-level sampling within the interval $[\alpha_i, \alpha_{i+1}]$ for each i. Namely, let $\{\gamma_{i,j}\}_{j=1}^n$ be parameters dividing $[\alpha_i, \alpha_{i+1}]$ uniformly, i.e.,

$$\gamma_{i,j} = \alpha_i + (\alpha_{i+1} - \alpha_i) \cdot \frac{j - 1}{n - 1}, \quad j = 1, \dots, n. \quad (25)$$

Now, suppose that L is a global Lipschitz constant for all functions $\{g_i\}_{i=1}^N$. By definition, for any $1 \leq i \leq N - 1$, L satisfies

$$L \geq \max \left\{ \max_{c,d \in [\alpha_i, \alpha_{i+1}]} \left| \frac{g_i(c) - g_i(d)}{c - d} \right|, \max_{c,d \in [\alpha_i, \alpha_{i+1}]} \left| \frac{g_{i+1}(c) - g_{i+1}(d)}{c - d} \right| \right\}. \quad (26)$$

Figure 7: An illustration of Grid Pixel Generator $G_{i,j}$, Color Extractors $\bar{m}$ and m_Δ (take blue channel as example), and the set $\mathcal{P}_{r,s}$.

In the following, we will derive explicit expressions for L for rotation and scaling. Given the Lipschitz constant L, one can show the following closed-form expression for M_i:

$$M_i = \frac{1}{2} \max_{1 \le j \le n-1} \left(\min\left\{g_i(\gamma_{i,j}) + g_i(\gamma_{i,j+1}), \right. \right.$$
$$\left. \left. g_{i+1}(\gamma_{i,j}) + g_{i+1}(\gamma_{i,j+1}) \right\} \right) + L \cdot \frac{b-a}{(N-1)(n-1)}. \quad (27)$$
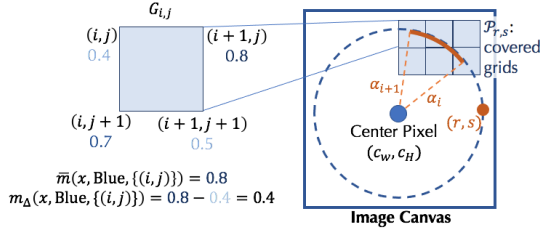
An illustration of this bounding technique using stratified sampling is shown in Figure 6. We notice that, as the number N of first-level samples is increased, the interpolation error M_i becomes smaller by shrinking the sampling interval $[\alpha_i, \alpha_{i+1}]$; similarly, increasing the number of second-level samples n makes the upper bound of the interpolation error M_i tighter since the term $L(b-a)/((N-1)(n-1))$ decreases. Furthermore, it is easy to see that as $N \to \infty$ or $n \to \infty$ we have $M \to M_S^2$, i.e., our interpolation error estimation is *asymptotically tight*. Finally, this tendency also highlights an important advantage of our two-level sampling approach: without stratified sampling, it is required to sample $N \times n$ α_i's in order to achieve the same level of approximation accuracy. As a consequence, these $N \times n$ α_i's in turn require to evaluate the smoothed classifier in Corollary 2 $N \times n$ times, compared to just N times in our case.

It thus remains to find a way to efficiently compute the Lipschitz constant L for different transformations. In the following, we derive closed form expressions for rotation and scaling transformations.

6.3 Computing the Lipschitz Constant

Here, we derive a global Lipschitz constant L for the functions $\{g_i\}_{i=1}^N$ defined in (22), for rotation and scaling transformations. In the following, we define K-channel images of width W and height H to be tensors $x \in \mathbb{R}^{K \times W \times H}$ and define the region of valid pixel indices as $\Omega := [0, W-1] \times [0, H-1] \cap \mathbb{N}^2$. Furthermore, for $(r, s) \in \Omega$, we define $d_{r,s}$ to be the ℓ_2-distance to the center of an image, i.e.,

$$d_{r,s} = \sqrt{(r - (W-1)/2)^2 + (s - (H-1)/2)^2}. \quad (28)$$

For ease of notation we make the following definitions that are illustrated in Figure 7.

DEFINITION 4 (GRID PIXEL GENERATOR). *For pixels $(i, j) \in \Omega$, we define the grid pixel generator G_{ij} as*

$$G_{ij} := \{(i, j), (i+1, j), (i, j+1), (i+1, j+1)\}. \quad (29)$$

DEFINITION 5 (MAX-COLOR EXTRACTOR). *We define the operator that extracts the channel-wise maximum pixel wise on a grid $S \subseteq \Omega$*

as the map $\bar{m} : \mathbb{R}^{K \times W \times H} \times \{0, \dots, K-1\} \times 2^\Omega \to \mathbb{R}$ with

$$\bar{m}(x, k, S) := \max_{(i,j) \in S} \left(\max_{(r,s) \in G_{ij}} x_{k,r,s} \right). \quad (30)$$

DEFINITION 6 (MAX-COLOR DIFFERENCE EXTRACTOR). *We define the operator that extracts the channel-wise maximum change in color on a grid $S \subseteq \Omega$ as the map $m_\Delta : \mathbb{R}^{K \times W \times H} \times \{0, \dots, K-1\} \times 2^\Omega \to \mathbb{R}$ with*

$$m_\Delta(x, k, S) := \max_{(i,j) \in S} \left(\max_{(r,s) \in G_{ij}} x_{k,r,s} - \min_{(r,s) \in G_{ij}} x_{k,r,s} \right). \quad (31)$$

Rotation. The rotation transformation is defined as rotating an image by an angle α counter-clock wise, followed by bilinear interpolation I. Clearly, when rotating an image, some pixels may be padded that results in a sudden change of pixel colors. To mitigate this issue, we apply black padding to all pixels that are outside the largest centered circle in a given image (see Figure 5a for an illustration). We define the rotation transformation ϕ_R as the (raw) rotation $\tilde{\phi}_R$ followed by interpolation and the aforementioned preprocessing step P so that $\phi_R = P \circ I \circ \tilde{\phi}_R$ and refer the reader to Appendix H for details. We remark that our certification is independent of different rotation padding mechanisms, since these padded pixels are all refilled by black padding during preprocessing. The following lemma provides a closed form expression for L in (27) for rotation. A detailed proof is given in Appendix I.

LEMMA 4. *Let $x \in \mathbb{R}^{K \times W \times H}$ be a K-channel image and let $\phi_R = P \circ I \circ \tilde{\phi}_R$ be the rotation transformation. Then, a global Lipschitz constant L for the functions $\{g_i\}_{i=1}^N$ is given by*

$$L_r = \max_{1 \le i \le N-1} \sum_{k=0}^{K-1} \sum_{r,s \in V} 2d_{r,s} \cdot m_\Delta(x, k, \mathcal{P}_{r,s}^{(i)}) \cdot \bar{m}(x, k, \mathcal{P}_{r,s}^{(i)}) \quad (32)$$

where $V = \left\{(r, s) \in \mathbb{N}^2 \mid d_{r,s} < \frac{1}{2}(\min\{W, H\} - 1)\right\}$. The set $\mathcal{P}_{r,s}^{(i)}$ is given by all integer grid pixels that are covered by the trajectory of source pixels of (r, s) when rotating from angle α_i to α_{i+1}.

Scaling. In Appendix A we introduce how to compute the Lipschitz bound for the scaling transformation and provide the certification. The process is similar to that for rotation.

Computational complexity. We provide pseudo-code for computing bound M in Appendix J. The algorithm is composed of two main parts, namely the computation of the Lipschitz constant L, and the computation of the interpolation error bound M based on L. The former is of computational complexity $O(N \cdot KWH)$, and the latter is of $O(NR \cdot KWH)$, for both scaling and rotation. We note that $\mathcal{P}_{r,s}$ contains only a constant number of pixels since each interval $[\alpha_i, \alpha_{i+1}]$ is small. Thus, the bulk of costs come from the transformation operation. We improve the speed by implementing a fast and fully-parallelized C kernel for rotation and scaling of images. As a result, on CIFAR-10, the algorithm takes less than 2 s on average with 10 processes for rotation with $N = 556$ and $n = 200$ and the time for scaling is faster. We refer readers to Section 7 for detailed experimental evaluation. Also, we remark that the algorithm is model-independent. Thus, we can precompute M for test set and reuse for any models that need a certification.

6.4 Discussion

Here, we briefly summarize the computation procedure of robustness certification, introduce an acceleration strategy—progressive sampling—and discuss the extensions beyond rotation and scaling.

6.4.1 Computation of Robustness Certification. With the methodology mentioned above, for differentially resolvable transformations such as rotation and scaling, computing robustness certification follows two steps: (1) computing the interpolation error bound M; (2) generate transformed samples $\{\phi(x, \alpha_i)\}_{i=1}^N$, compute $p_A^{(i)}$ and $p_B^{(i)}$ for each sample, and check whether $M_S < R$ holds for each sample according to Corollary 2.

6.4.2 Acceleration: Progressive Sampling. In step (2) above, we need to estimate $p_A^{(i)}$ and $p_B^{(i)}$ for each sample $\phi(x, \alpha_i)$ to check whether $M_S < R$. In the brute-force approach, to obtain a high-confidence bound on $p_A^{(i)}$ and $p_B^{(i)}$, we typically sample $n_s = 10,000$ or more [8] then apply the binomial statistical test. In total, we thus need to sample the classifier's prediction $N \times n_s$ times, which is costly.

To accelerate the computation, we design a *progressive sampling strategy* from the following two insights: (1) we only need to check whether $R > M_S$, but are not required to compute R precisely; (2) for any sample $\phi(x, \alpha_i)$ if the check fails, the model is not certifiably robust and there is no need to proceed. Based on (1), for the current $\phi(x, \alpha_i)$, we sample n_s samples in batches and maintain high-confidence lower bound of R based on existing estimation. Once the lower bound exceeds M_S we proceed to the next $\phi(x, \alpha_{i+1})$. Based on (2), we terminate early if the check $R > M_S$ for the current $\phi(x, \alpha_i)$ fails. More details are provided in Appendix J.

6.4.3 Extension to More Transformations. For other transformations that involve interpolation, we can similarly compute the interpolation error bound using intermediate results in our above lemmas. For transformation compositions, we extend our certification pipeline for the composition of (1) rotation/scaling with brightness, and (2) rotation/scaling with brightness and ℓ_p-bounded additive perturbations. These compositions simulate an attacker who does not precisely perform the specified transformation. We present these extensions in Appendix B.3 and Appendix B.4 in detail, and in Appendix B.5 we discuss how to analyze possible new transformations and then extend **TSS** to provide certification.

7 EXPERIMENTS

We validate our framework **TSS** by certifying robustness over semantic transformations experimentally. We compare with state of the art for each transformation, highlight our main results, and present some interesting findings and ablation studies.

7.1 Experimental Setup

7.1.1 Dataset. Our experiments are conducted on three classical image classification datasets: MNIST, CIFAR-10, and ImageNet. For all images, the pixel color is normalized to $[0, 1]$. We follow common practice to resize and center cropping the ImageNet images to 224×224 size [8, 27, 42, 65]. To our best knowledge, we are the *first* to provide rigorous certifiable robustness against semantic transformations on the large-scale *standard* ImageNet dataset.

7.1.2 Model. The undefended model is very vulnerable even under simple random semantic attacks. Therefore, we apply existing data

augmentation training [8] combined with consistency regularization [27] to train the base classifiers. We then use the introduced smoothing strategies, to obtain the models for robustness certification. On MNIST and CIFAR-10, the models are trained from scratch while on ImageNet, we either finetune undefended models in `torchvision` library or finetune from state-of-the-art certifiably robust models against ℓ_2 perturbations [44]. Details are available in Appendix K.1. We remark that our framework focuses on robustness certification and did not fully explore the training methods for improving the certified robustness or tune the hyperparameters.

7.1.3 Implementation and Hardware. We implement our framework **TSS** based on `PyTorch`. We improve the running efficiency by tensor parallelism and embedding `C` modules. Details are available in Appendix K.2. All experiments were run on 24-core Intel Xeon Platinum 8259CL CPU and one Tesla T4 GPU with 15 GB RAM.

7.1.4 Evaluation Metric. On each dataset, we uniformly pick 500 samples from the test set and evaluate all results on this *test subset* following Cohen et al [8]. In line with related work [8, 27, 44, 65], we report the **certified robust accuracy** that is defined as the fraction of samples (within the test subset) that are both *certified robust* and *classified correctly*, and set the certification confidence level to $p = 0.1\%$. We use $n_s = 10^5$ samples to obtain a confidence lower bound $\underline{p_A}$ for resolvable transformations, and $n_s = 10^4$ samples to obtain each $\underline{p_A}^{(i)}$ for differentially resolvable transformations. Due to Progressive Sampling (Algorithm 2), the actual samples used for differentially resolvable transformations are usually far fewer than n_s. In addition, we report the **benign accuracy** in Appendix K.5 defined as the fraction of *correctly classified* samples when no attack is present, and the **empirical robust accuracy**, defined as the fraction of samples in the test subset that are classified correctly under either a simple random attack (following [3, 14]) or two adaptive attacks (namely Random+ Attack and PGD Attack). We introduce all these attacks in Appendix K.3 and provide a detailed comparison in Appendix K.7. Note that the empirical robust accuracy under any attacks is lower bounded by the certified accuracy.

7.1.5 Notations for Robust Radii. In the tables, we use these notations: α for squared kernel radius for Gaussian blur; $\sqrt{\Delta x^2 + \Delta y^2}$ for translation distance; b and c for brightness shift and contrast change respectively as in $x \mapsto (1+c)x + b$; r for rotation angle; s for size scaling ratio; and $\|\delta\|_2$ for ℓ_2 norm of additional perturbations.

7.1.6 Vanilla Models and Baselines. We compare with vanilla (undefended) models and baselines from related work. The vanilla models are trained to achieve high accuracy only on clean data. For fairness, on all datasets we use the same model architectures as in our approach. On the test subset, the *benign accuracy* of vanilla models is 98.6%/88.6%/74.4% on MNIST/CIFAR-10/ImageNet. We also report their empirical robust accuracy under attacks in Table 2. Since vanilla models are not smoothed, we cannot have certified robust accuracy for them. In terms of baselines, we consider the approaches that provide certification against semantic transformations: DeepG [3], Interval [49], VeriVis [41], Semanify-NN [37], and DistSPT [14]. In Appendix K.4, we provide more detailed discussion and comparison with these baseline approaches, and list how we run these approaches for fair comparison.

Table 2: Comparison of certified robust accuracy achieved by our framework **TSS** and other known baselines and empirical robust accuracy achieved by **TSS** and vanilla models under random and adaptive attacks. "-" denotes the settings where the baselines cannot support. The parentheses show the weaker baseline settings. For certified robust accuracy, the existing state of the art is **bolded**. For empirical robust accuracy, the higher accuracy under each setting are **bolded**.

| Transformation | Type | Dataset | Attack Radius | Certified Robust Accuracy | | | | | | Empirical Robust Accuracy | | | |
| | | | | | | | | | | Random Attack | | Adaptive Attacks | |
				TSS	DeepG [3]	Interval [49]	VeriVis [41]	Semanify-NN [37]	DistSPT [14]	TSS	Vanilla	TSS	Vanilla
Gaussian Blur	Resolvable	MNIST	Squared Radius $\alpha \leq 36$	**90.6%**	-	-	-	-	-	**91.4%**	12.2%	**91.2%**	12.2%
		CIFAR-10	Squared Radius $\alpha \leq 16$	**63.6%**	-	-	-	-	-	**65.8%**	3.4%	**65.8%**	3.4%
		ImageNet	Squared Radius $\alpha \leq 36$	**51.6%**	-	-	-	-	-	**52.8%**	8.4%	**52.6%**	8.2%
Translation (Reflection Pad.)	Resolvable, Discrete	MNIST	$\sqrt{\Delta x^2 + \Delta y^2} \leq 8$	**99.6%**	-	-	98.8%	98.8%	-	**99.6%**	0.0%	**99.6%**	0.0%
		CIFAR-10	$\sqrt{\Delta x^2 + \Delta y^2} \leq 20$	**80.8%**	-	-	65.0%	65.0%	-	**86.2%**	4.4%	**86.0%**	4.2%
		ImageNet	$\sqrt{\Delta x^2 + \Delta y^2} \leq 100$	**50.0%**	-	-	43.2%	43.2%	-	**69.2%**	46.6%	**69.2%**	46.2%
Brightness	Resolvable	MNIST	$b \pm 50\%$	**98.2%**	-	-	-	-	-	**98.2%**	96.6%	**98.2%**	96.6%
		CIFAR-10	$b \pm 40\%$	**87.0%**	-	-	-	-	-	87.2%	44.4%	**87.4%**	42.6%
		ImageNet	$b \pm 40\%$	**70.0%**	-	-	-	-	-	**70.4%**	19.6%	**70.4%**	18.4%
Contrast and Brightness	Resolvable, Composition	MNIST	$c \pm 50\%, b \pm 50\%$	**97.6%**	$\leq 0.4\%$ $(c, b \pm 30\%)$	0.0% $(c, b \pm 30\%)$	-	$\leq 74\%$ $(c \pm 5\%, b \pm 50\%)$	-	**98.0%**	94.6%	**98.0%**	93.2%
		CIFAR-10	$c \pm 40\%, b \pm 40\%$	**82.4%**	0.0% $(c, b \pm 30\%)$	0.0% $(c, b \pm 30\%)$	-	-	-	**86.0%**	21.0%	**85.8%**	9.6%
		ImageNet	$c \pm 40\%, b \pm 40\%$	**61.4%**	-	-	-	-	-	**68.4%**	1.2%	**68.4%**	0.0%
Gaussian Blur, Translation, Brightness, and Contrast	Resolvable, Composition	MNIST	$\alpha \leq 1, \sqrt{\Delta x^2 + \Delta y^2} \leq 5, c, b \pm 10\%$	**90.2%**	-	-	-	-	-	**97.2%**	0.4%	**97.0%**	0.4%
		CIFAR-10	$\alpha \leq 1, \sqrt{\Delta x^2 + \Delta y^2} \leq 5, c, b \pm 10\%$	**58.2%**	-	-	-	-	-	67.6%	9.6%	**67.8%**	5.6%
		ImageNet	$\alpha \leq 10, \sqrt{\Delta x^2 + \Delta y^2} \leq 10, c, b \pm 20\%$	**32.8%**	-	-	-	-	-	**48.8%**	9.4%	**47.4%**	4.0%
Rotation	Differentially Resolvable	MNIST	$r \pm 50°$	**97.4%**	$\leq 85.8\%$ $(r \pm 30°)$	$\leq 6.0\%$ $(r \pm 30°)$	-	$\leq 92.48\%$	82%	**98.4%**	12.2%	**98.2%**	11.0%
		CIFAR-10	$r \pm 10°$	**70.6%**	62.5%	20.2%	-	-	37%	**76.6%**	65.6%	**76.4%**	65.4%
			$r \pm 30°$	**63.6%**	10.6%	0.0%	-	$\leq 49.37\%$	22%	69.2%	21.6%	**69.4%**	21.4%
		ImageNet	$r \pm 30°$	**30.4%**	-	-	-	-	16% (rand. attack)	37.8%	**40.0%**	37.8%	37.0%
Scaling	Differentially Resolvable	MNIST	$s \pm 30\%$	**97.2%**	85.0%	16.4%	-	-	-	**99.2%**	90.2%	**99.2%**	89.2%
		CIFAR-10	$s \pm 30\%$	**58.8%**	0.0%	0.0%	-	-	-	**67.2%**	51.6%	**67.0%**	51.2%
		ImageNet	$s \pm 30\%$	**26.4%**	-	-	-	-	-	37.4%	**50.0%**	36.4%	**49.8%**
Rotation and Brightness	Differentially Resolvable, Composition	MNIST	$r \pm 50°, b \pm 20\%$	**97.0%**	-	-	-	-	-	**98.2%**	11.0%	**98.0%**	10.4%
		CIFAR-10	$r \pm 10°, b \pm 10\%$	**70.2%**	-	-	-	-	-	**76.6%**	59.4%	**76.0%**	56.8%
			$r \pm 30°, b \pm 20\%$	**61.4%**	-	-	-	-	-	**68.4%**	13.0%	**68.2%**	9.0%
		ImageNet	$r \pm 30°, b \pm 20\%$	**26.8%**	-	-	-	-	-	**37.4%**	22.4%	**36.8%**	21.2%
Scaling and Brightness	Differentially Resolvable, Composition	MNIST	$s \pm 50\%, b \pm 50\%$	**96.6%**	-	-	-	-	-	**97.8%**	24.8%	**97.8%**	15.6%
		CIFAR-10	$s \pm 30\%, b \pm 30\%$	**54.2%**	-	-	-	-	-	**67.2%**	17.4%	**66.8%**	11.6%
		ImageNet	$s \pm 30\%, b \pm 30\%$	**23.4%**	-	-	-	-	-	**36.4%**	16.0%	**36.0%**	8.8%
Rotation, Brightness, and ℓ_2	Differentially Resolvable, Composition	MNIST	$r \pm 50°, b \pm 20\%, \|\delta\|_2 \leq .05$	**96.6%**	-	-	-	-	-	**97.6%**	10.8%	**97.4%**	9.0%
		CIFAR-10	$r \pm 10°, b \pm 10\%, \|\delta\|_2 \leq .05$	**64.2%**	-	-	-	-	-	**71.6%**	31.8%	**71.2%**	29.6%
			$r \pm 30°, b \pm 20\%, \|\delta\|_2 \leq .05$	**55.2%**	-	-	-	-	-	**65.2%**	0.8%	**64.0%**	0.4%
		ImageNet	$r \pm 30°, b \pm 20\%, \|\delta\|_2 \leq .05$	**26.6%**	-	-	-	-	-	**37.0%**	17.6%	**36.4%**	14.0%
Scaling, Brightness, and ℓ_2	Differentially Resolvable, Composition	MNIST	$s \pm 50\%, b \pm 50\%, \|\delta\|_2 \leq .05$	**96.4%**	-	-	-	-	-	**97.6%**	22.2%	**97.6%**	12.2%
		CIFAR-10	$s \pm 30\%, b \pm 30\%, \|\delta\|_2 \leq .05$	**51.2%**	-	-	-	-	-	**65.0%**	4.4%	**61.8%**	2.6%
		ImageNet	$s \pm 30\%, b \pm 30\%, \|\delta\|_2 \leq .05$	**22.6%**	-	-	-	-	-	**36.0%**	7.4%	**35.6%**	4.8%

7.2 Main Results

Here, we present our main results from five aspects: (1) certified robustness compared to baselines; (2) empirical robustness comparison; (3) certification time statistics; (4) empirical robustness under unforeseen physical attacks; (5) certified robustness under attacks exceeding the certified radii.

7.2.1 Certified Robustness Compared to Baselines.

Our results are summarized in Table 2. For each transformation, we ensure that our setting is either the same as or strictly stronger than all other baselines. When our setting is strictly stronger, the baseline setting is shown in corresponding parentheses, and our certified robust accuracy implies a higher or equal certified robust accuracy in the corresponding baseline setting. To our best knowledge, we are the first to provide certified robustness for Gaussian blur, brightness, composition of rotation and brightness, etc. Moreover, on the large-scale standard ImageNet dataset, we are the first to provide nontrivial certified robustness against certain semantic attacks. Note that DistSPT [14] is theoretically feasible to provide robustness certification for the ImageNet dataset. However, its certification is not tight enough to handle ImageNet and it provides robustness certification for only a certain random attack instead of arbitrarily worst-case attacks [14, Section 7.4]. We observe that, across transformations,

our framework *significantly* outperforms the state of the art, if present, in terms of robust accuracy. For example, on the composition of contrast and brightness, we improve the certified robust accuracy from 74% to 97.6% on MNIST, from 0.0% (failing to certify) to 82.4% on CIFAR-10, and from 0% (absence of baseline) to 61.4% on ImageNet. On the rotation transformation, we improve the certified robust accuracy from 92.48% to 97.4% on MNIST, from 49.37% to 63.6% on CIFAR-10 (rotation angle within 30°), and from 16% against a certain random attack to 30.4% against arbitrary attacks on ImageNet. Some baselines are able to provide certification under other certification goals and the readers can refer to Appendix K.4 for a detailed discussion.

7.2.2 Comparison of Empirical Robust Accuracy.

In Table 2, we report the empirical robust accuracy for both (undefended) vanilla models and trained **TSS** models. The empirical robust accuracy is either evaluated under random attack or two adaptive attacks– Random+ and PGD attack. When it is under adaptive attacks, we report the lower accuracy to evaluate against stronger attackers.

(1) For almost all settings, **TSS** models have significantly higher *empirical robust accuracy*, which means that **TSS** models are also practical in terms of defending against existing attacks. The only exception is rotation and scaling on ImageNet. The

reason is that a single rotation/scaling transformation is too weak to attack even an undefended model. At the same time, our robustness certification comes at the cost of benign accuracy, which also affects the empirical robust accuracy. This exception is eliminated when rotation and scaling are composed with other transformations.

(2) Similar observations arise when comparing the *empirical robust accuracy of the vanilla model with the certified robust accuracy of ours.* Hence, even compared to *empirical* metrics, our *certified* robust accuracy is nontrivial and guarantees high accuracy.

(3) Our *certified* robust accuracy is always lower or equal compared to the *empirical* one, verifying the validity of our robustness certification. The gaps range from ∼ 2% on MNIST to ∼ 10% - 20% on ImageNet. Since empirical robust accuracy is an upper bound of the certified accuracy, this implies that our certified bounds are usually tight, particularly on small datasets.

(4) The adaptive attack decreases the empirical accuracy of **TSS** models *slightly*, while it decreases that of vanilla models significantly. Taking contrast and brightness on CIFAR-10 as example, **TSS** accuracy decreases from 86% to 85.8% while the vanilla model accuracy decreases from 21.0% to 9.6%. Thus, **TSS** is still robust against adaptive attacks. Indeed, **TSS** has robustness guarantee against any attack within the certified radius.

7.2.3 Certification Time Statistics. Our robustness certification time is usually less than 100 s on MNIST and 200 s on CIFAR-10; on ImageNet it is around 200 s - 2000 s. Compared to other baselines, ours is slightly faster and achieves much higher certified robustness. For fairness, we give 1000 s time limit per instance when running baselines on MNIST and CIFAR-10. Note that other baselines cannot scale up to ImageNet. Our approach is scalable due to the black-box nature of smoothing-based certification, the tight interpolation error upper bound, and the efficient progressive sampling strategy. Details on hyperparameters including smoothing variance and average certification time are given in Appendix K.6.

7.2.4 Generalization to Unforeseen Common Corruptions. Are **TSS** models still more robust when it comes to potential unforeseen physical attacks? To answer this question, we evaluate the robustness of **TSS** models on the realistic CIFAR-10-C and ImageNet-C datasets [22]. These two datasets are comprised of corrupted images from CIFAR-10 and ImageNet. They apply around 20 types of common corruptions to model *physical attacks*, such as fog, snow, and frost. We evaluate the *empirical robust accuracy* against the highest corruption level (level 5) to model the strongest physical attacker. We apply **TSS** models trained against a transformation composition attack, Gaussian blur + brightness + contrast + translation, to defend against these corruptions. We select two baselines: vanilla models and AugMix [23]. AugMix is the state of the art model on CIFAR-10-C and ImageNet-C [9].

The results are shown in Table 3. The answer is *yes*—**TSS** models are more robust than undefended vanilla models. It even exceeds the state of the art, AugMix, on CIFAR-10-C. On ImageNet-C, **TSS** model's empirical accuracy is between vanilla and AugMix. We emphasize that in contrast to **TSS**, both vanilla and AugMix fail to provide robustness certification. Details on evaluation protocols and additional findings are in Appendix K.8.

Table 3: Comparison of **empirical accuracy** of different models under physical corruptions (CIFAR-10-C and ImageNet-C) and **certified accuracy** against composition of transformations. **TSS** achieves higher or comparable empirical accuracy against unforeseen corruptions and significantly higher certified accuracy (under attack radii in Table 2).

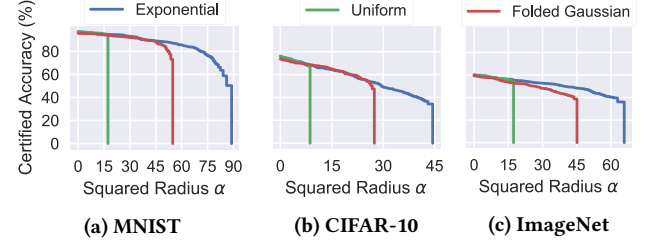|	CIFAR-10			ImageNet		
	Vanilla	AugMix [23]	**TSS**	Vanilla	AugMix [23]	**TSS**
Empirical Accuracy on CIFAR-10-C and ImageNet-C	53.9%	65.6%	**67.4%**	18.3%	**25.7%**	21.9%
Certified Accuracy against Composition of Gaussian Blur, Translation, Brightness, and Contrast	0.0%	0.4%	**58.2%**	0.0%	0.0%	**32.8%**

Figure 8: Certified accuracy for different smoothing distributions for Gaussian blur. On MNIST/CIFAR-10/ImageNet the noise std. is 10/5/10.

Table 4: Study of the impact of different smoothing variance levels on certified robust accuracy and benign accuracy on **ImageNet** for **TSS**. The attack radii are consistent with Table 2. "Dist." refers to both training and smoothing distribution. The variance used in Table 2 is labeled in gray.

Transformation	Attack Radii	Certified Accuracy and Benign Accuracy under Different Variance Levels			
Gaussian Blur	$\alpha \leq 36$	Dist. of α	Exp(1/5)	Exp(1/10)	Exp(1/20)
		Cert. Rob. Acc.	0.0%	51.6%	48.4%
		Benign Acc.	63.4%	59.2%	53.2%
Translation (Reflection Pad.)	$\sqrt{\Delta x^2 + \Delta y^2} \leq 100$	Dist. of $(\Delta x, \Delta y)$	$\mathcal{N}(0, 20^2 I)$	$\mathcal{N}(0, 30^2 I)$	$\mathcal{N}(0, 40^2 I)$
		Cert. Rob. Acc.	0.0%	50.0%	55.4%
		Benign Acc.	70.0%	72.6%	70.0%
Brightness	$b \pm 40\%$	Dist. of (c, b)	$\mathcal{N}(0, 0.3^2 I)$	$\mathcal{N}(0, 0.4^2 I)$	$\mathcal{N}(0, 0.5^2 I)$
		Cert. Rob. Acc.	**70.2%**	70.0%	67.6%
		Benign Acc.	73.2%	72.2%	69.4%
Contrast	$c \pm 40\%$	Dist. of (c, b)	$\mathcal{N}(0, 0.3^2 I)$	$\mathcal{N}(0, 0.4^2 I)$	$\mathcal{N}(0, 0.5^2 I)$
		Cert. Rob. Acc.	58.4%	63.6%	**65.0%**
		Benign Acc.	72.8%	71.4%	68.6%
Rotation	$r \pm 30°$	Dist. of ε	$\mathcal{N}(0, 0.25^2 I)$	$\mathcal{N}(0, 0.50^2 I)$	$\mathcal{N}(0, 1.00^2 I)$
		Cert. Rob. Acc.	9.8%	**30.4%**	20.0%
		Benign Acc.	**55.6%**	46.2%	32.2%
Scaling	$s \pm 30\%$	Dist. of ε	$\mathcal{N}(0, 0.25^2 I)$	$\mathcal{N}(0, 0.50^2 I)$	$\mathcal{N}(0, 1.00^2 I)$
		Cert. Rob. Acc.	7.2%	**26.4%**	17.4%
		Benign Acc.	**58.8%**	50.8%	33.8%

7.2.5 Evaluation on Attacks Beyond Certified Radii. The semantic attacker in the physical world may not constrain itself to be within the specified attack radii. In Appendix K.9 we present a thorough evaluation of **TSS**'s robustness when the attack radii go beyond the certified ones. We show, for example, for **TSS** model defending against ±40% brightness change on ImageNet, when the radius increases to 50%, the certified accuracy only slightly drops from 70.4% to 70.0%. In a nutshell, there is no significant or immediate degradation on both certified robust accuracy and empirical robust accuracy when the attack radii go beyond the certified ones.

7.3 Ablation Studies

Here, we provide two ablation studies: (1) Comparison of different smoothing distributions; (2) Comparison of different smoothing variances. In Appendix K.11, we present another ablation study on

different numbers of samples for differentially resolvable transformations, which reveals a tightness-efficiency trade-off.

7.3.1 Comparison of Smoothing Distributions.
To study the effects of different smoothing distributions, we compare the certified robust accuracy for Gaussian blur when the model is smoothed by different smoothing distributions. We consider three smoothing distributions, namely exponential (blue line), uniform (green line), and folded Gaussian (red line). On each dataset, we adjust the distribution parameters such that each distribution has the same variance. All other hyperparameters are kept the same throughout training and certification. As shown Figure 8, we notice that on all three datasets, the exponential distribution has the highest average certified radius. This observation is in line with our theoretical reasoning in Section 5.2.

7.3.2 Comparison of Different Smoothing Variances.
The variance of the smoothing distribution is a hyperparameter that controls the accuracy-robustness trade-off. In Table 4, we evaluate different smoothing variances for several transformations on ImageNet and report both the certified accuracy and benign accuracy. The results on MNIST and CIFAR-10 and more discussions are in Appendix K.10. From these results, we observe that usually, when the smoothing variance increases, the benign accuracy drops and the certified robust accuracy first rises and then drops. This tendency is also observed in classical randomized smoothing [8, 65]. However, the range of acceptable variance is usually wide. Thus, without careful tuning the smoothing variances, we are able to achieve high certified and benign accuracy as reported in Table 2 and Table 6.

8 RELATED WORK

Certified Robustness against ℓ_p perturbations. Since the studies of adversarial vulnerability of neural networks [17, 50], there has emerged a rich body of research on evasion attacks (e.g., [2, 6, 55, 61]) and empirical defenses (e.g., [34, 46, 47]). To provide robustness certification, different robustness training and verification approaches have been proposed. In particular, interval bound propagation [18, 68], linear relaxations [35, 58–60, 63], and semidefinite programming [10, 43] have been applied to certify NN robustness. Recently, robustness certification based on randomized smoothing has shown to be scalable and with tight guarantees [8, 29, 30]. With improvements on optimizing the smoothing distribution [11, 51, 65] and better training mechanisms [7, 27, 44, 66], the verified robustness of randomized smoothing is further improved. A recent survey summarizes certified robustness approaches [31].

Semantic Attacks for Neural Networks. Recent work has shown that semantic transformations are able to mislead ML models [15, 24, 62]. For instance, image rotations and translations can attack ML models with 40% - 99% degradation on MNIST, CIFAR-10, and ImageNet on both vanilla models and models that are robust against ℓ_p-bounded perturbations [12]. Brightness/contrast attacks can achieve 91.6% attack success on CIFAR-10, and 71%-100% attack success rate on ImageNet [22]. Our evaluation on empirical robust accuracy (Table 2) for vanilla models also confirms these observations. Moreover, brightness attacks have been shown to be of practical concern in autonomous driving [40]. Empirical defenses against semantic transformations have been investigated [12, 22].

Certified Robustness against Semantic Transformations. While heuristic defenses against semantic attacks have been proposed, *provable* robustness requires further investigation. Existing certified robustness against transformations is based on heuristic enumeration, interval bound propagation, linear relaxation, or smoothing. Efficient enumeration in VeriVis [41] can handle only discrete transformations. Interval bound propagation has been used to certify common semantic transformations [3, 14, 49]. To tighten the interval bounds, linear relaxations are introduced. DeepG [3] optimizes linear relaxations for given semantic transformations, and Semantify-NN [37] encodes semantic transformations by neural networks and applies linear relaxations for NNs [58, 68]. However, linear relaxations are loose and computationally intensive compared to our **TSS**. Recently, Fischer et al [14] have applied a smoothing scheme to provide provable robustness against transformations but on the large ImageNet dataset, it can provide certification only against random attacks that draw transformation parameters from a pre-determined distribution. More details are available in Appendix K.4.

9 CONCLUSION

In this paper, we have presented a unified framework, **TSS**, for certifying ML robustness against general semantic adversarial transformations. Extensive experiments have shown that **TSS** significantly outperforms the state of the art or, if no previous work exists, set new baselines. In future work, we plan to further improve the efficiency and tightness of our robustness certification and explore more transformation-specific smoothing strategies.

REFERENCES

[1] M. Abramowitz and I. A. (Eds.) Stegun. 1972. *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables, 9th printing.* New York: Dover, New York.

[2] Anish Athalye, Nicholas Carlini, and David Wagner. 2018. Obfuscated Gradients Give a False Sense of Security: Circumventing Defenses to Adversarial Examples. In *2018 International Conference on Machine Learning (ICML)*. PMLR, 274–283.

[3] Mislav Balunovic, Maximilian Baader, Gagandeep Singh, Timon Gehr, and Martin Vechev. 2019. Certifying Geometric Robustness of Neural Networks. In *2019 Advances in Neural Information Processing Systems (NeurIPS)*. Curran Associates, Inc., 15287–15297.

[4] Avrim Blum, Travis Dick, Naren Manoj, and Hongyang Zhang. 2020. Random Smoothing Might be Unable to Certify ℓ_∞ Robustness for High-Dimensional Images. *Journal of Machine Learning Research (JMLR)* 21, 211 (2020), 1–21.

[5] Saikiran Bulusu, Bhavya Kailkhura, Bo Li, Pramod K Varshney, and Dawn Song. 2020. Anomalous Example Detection in Deep Learning: A Survey. *IEEE Access* 8 (2020), 132330–132347.

[6] Nicholas Carlini and David Wagner. 2017. Towards Evaluating the Robustness of Neural Networks. In *2017 IEEE Symposium on Security and Privacy (SP)*. IEEE Computer Society, 39–57.

[7] Yair Carmon, Aditi Raghunathan, Ludwig Schmidt, John C Duchi, and Percy S Liang. 2019. Unlabeled Data Improves Adversarial Robustness. In *2019 Advances in Neural Information Processing Systems (NeurIPS)*. Curran Associates, Inc., 11192–11203.

[8] Jeremy Cohen, Elan Rosenfeld, and Zico Kolter. 2019. Certified Adversarial Robustness via Randomized Smoothing. In *2019 International Conference on Machine Learning (ICML)*. PMLR, 1310–1320.

[9] Francesco Croce, Maksym Andriushchenko, Vikash Sehwag, Nicolas Flammarion, Mung Chiang, Prateek Mittal, and Matthias Hein. 2020. RobustBench: A Standardized Adversarial Robustness Benchmark. *arXiv preprint arXiv:2010.09670* 10 (2020).

[10] Sumanth Dathathri, Krishnamurthy Dvijotham, Alexey Kurakin, Aditi Raghunathan, Jonathan Uesato, Rudy Bunel, Shreya Shankar, Jacob Steinhardt, Ian Goodfellow, Percy Liang, et al. 2020. Enabling Certification of Verification-Agnostic Networks via Memory-Efficient Semidefinite Programming. In *2020 Advances in Neural Information Processing Systems (NeurIPS)*. Curran Associates, Inc., 5318–5331.

[11] Krishnamurthy Dvijotham, Jamie Hayes, Borja Balle, Zico Kolter, Chongli Qin, Andras Gyorgy, Kai Xiao, Sven Gowal, and Pushmeet Kohli. 2020. A Framework for Robustness Certification of Smoothed Classifiers using f-Divergences. In *2020 International Conference on Learning Representations (ICLR)*. OpenReview.

[12] Logan Engstrom, Brandon Tran, Dimitris Tsipras, Ludwig Schmidt, and Aleksander Madry. 2019. Exploring the Landscape of Spatial Robustness. In *2019 International Conference on Machine Learning (ICML)*. PMLR, 1802–1811.

[13] Kevin Eykholt, Ivan Evtimov, Earlence Fernandes, Bo Li, Amir Rahmati, Chaowei Xiao, Atul Prakash, Tadayoshi Kohno, and Dawn Song. 2018. Robust Physical-World Attacks on Deep Learning Visual Classification. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 1625–1634.

[14] Marc Fischer, Maximilian Baader, and Martin Vechev. 2020. Certified Defense to Image Transformations via Randomized Smoothing. In *2020 Advances in Neural Information Processing Systems (NeurIPS)*. Curran Associates, Inc., 8404–8417.

[15] Amin Ghiasi, Ali Shafahi, and Tom Goldstein. 2020. Breaking Certified Defenses: Semantic Adversarial Examples with Spoofed Robustness Certificates. In *2020 International Conference on Learning Representations (ICLR)*. OpenReview.

[16] Tejas Gokhale, Rushil Anirudh, Bhavya Kailkhura, Jayaraman J Thiagarajan, Chitta Baral, and Yezhou Yang. 2021. Attribute-Guided Adversarial Training for Robustness to Natural Perturbations. In *2021 AAAI Conference on Artificial Intelligence (AAAI)*, Vol. 35. Association for the Advancement of Artificial Intelligence Press, 7574–7582.

[17] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. 2015. Explaining and Harnessing Adversarial Examples. In *2015 International Conference on Learning Representations (ICLR)*. OpenReview.

[18] Sven Gowal, Krishnamurthy Dvijotham, Robert Stanforth, Rudy Bunel, Chongli Qin, Jonathan Uesato, Relja Arandjelovic, Timothy Mann, and Pushmeet Kohli. 2019. Scalable Verified Training for Provably Robust Image Classification. In *2019 IEEE International Conference on Computer Vision (ICCV)*. IEEE, 4842–4851.

[19] Jamie Hayes. 2020. Extensions and Limitations of Randomized Smoothing for Robustness Guarantees. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. IEEE, 3413–3421.

[20] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2015. Delving deep into rectifiers: Surpassing Human-Level Performance on ImageNet Classification. In *2015 IEEE International Conference on Computer Vision (ICCV)*. IEEE, 1026–1034.

[21] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 770–778.

[22] Dan Hendrycks and Thomas Dietterich. 2018. Benchmarking Neural Network Robustness to Common Corruptions and Perturbations. In *2018 International*

[23] Dan Hendrycks, Norman Mu, Ekin Dogus Cubuk, Barret Zoph, Justin Gilmer, and Balaji Lakshminarayanan. 2020. AugMix: A Simple Data Processing Method to Improve Robustness and Uncertainty. In *2020 International Conference on Learning Representations (ICLR)*. OpenReview.

[24] Hossein Hosseini and Radha Poovendran. 2018. Semantic Adversarial Examples. In *2018 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. IEEE, 1614–1619.

[25] Weiwei Hu and Ying Tan. 2017. Generating Adversarial Malware Examples for Black-Box Attacks Based on GAN. *arXiv preprint arXiv:1702.05983* 02 (2017).

[26] Jean Jacod and Philip Protter. 2000. *Probability essentials.* Springer Science & Business Media, Berlin.

[27] Jongheon Jeong and Jinwoo Shin. 2020. Consistency Regularization for Certified Robustness of Smoothed Classifiers. In *2020 Advances in Neural Information Processing Systems (NeurIPS)*. Curran Associates, Inc., 10558–10570.

[28] Aounon Kumar, Alexander Levine, Tom Goldstein, and Soheil Feizi. 2020. Curse of Dimensionality on Randomized Smoothing for Certifiable Robustness. In *2020 International Conference on Machine Learning (ICML)*. PMLR, 5458–5467.

[29] Mathias Lecuyer, Vaggelis Atlidakis, Roxana Geambasu, Daniel Hsu, and Suman Jana. 2019. Certified Robustness to Adversarial Examples with Differential Privacy. In *2019 IEEE Symposium on Security and Privacy (SP)*. IEEE, 656–672.

[30] Bai Li, Changyou Chen, Wenlin Wang, and Lawrence Carin. 2019. Certified Adversarial Robustness with Additive Noise. In *2019 Advances in Neural Information Processing Systems (NeurIPS)*. Curran Associates, Inc., 9459–9469.

[31] Linyi Li, Xiangyu Qi, Tao Xie, and Bo Li. 2020. SoK: Certified Robustness for Deep Neural Networks. *arXiv preprint arXiv:2009.04131* 09 (2020).

[32] Linyi Li, Zexuan Zhong, Bo Li, and Tao Xie. 2019. Robustra: Training Provable Robust Neural Networks over Reference Adversarial Space. In *2019 International Joint Conference on Artificial Intelligence (IJCAI)*. International Joint Conferences on Artificial Intelligence Organization, 4711–4717.

[33] Xingjun Ma, Bo Li, Yisen Wang, Sarah M Erfani, Sudanthi Wijewickrema, Grant Schoenebeck, Dawn Song, Michael E Houle, and James Bailey. 2018. Characterizing Adversarial Subspaces Using Local Intrinsic Dimensionality. In *2018 International Conference on Learning Representations (ICLR)*. OpenReview.

[34] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. 2018. Towards Deep Learning Models Resistant to Adversarial Attacks. In *2018 International Conference on Learning Representations (ICLR)*. OpenReview.

[35] Matthew Mirman, Timon Gehr, and Martin Vechev. 2018. Differentiable Abstract Interpretation for Provably Robust Neural Networks. In *2018 International Conference on Machine Learning (ICML)*. PMLR, 3575–3583.

[36] Jeet Mohapatra, Ching-Yun Ko, Lily Weng, Pin-Yu Chen, Sijia Liu, and Luca Daniel. 2021. Hidden Cost of Randomized Smoothing. In *2021 International Conference on Artificial Intelligence and Statistics (AISTATS)*. PMLR, 4033–4041.

[37] Jeet Mohapatra, Tsui-Wei Weng, Pin-Yu Chen, Sijia Liu, and Luca Daniel. 2020. Towards Verifying Robustness of Neural Networks Against A Family of Semantic Perturbations. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 244–252.

[38] Matej Moravčík, Martin Schmid, Neil Burch, Viliam Lisỳ, Dustin Morrill, Nolan Bard, Trevor Davis, Kevin Waugh, Michael Johanson, and Michael Bowling. 2017. DeepStack: Expert-Level Artificial Intelligence in Heads-Up No-Limit Poker. *Science* 356, 6337 (2017), 508–513.

[39] OpenCV. 2020. OpenCV: Transformations of Images. https://docs.opencv.org/master/dd/d52/tutorial_js_geometric_transformations.html.

[40] Kexin Pei, Yinzhi Cao, Junfeng Yang, and Suman Jana. 2017. DeepXplore: Automated Whitebox Testing of Deep Learning Systems. In *2017 Symposium on Operating Systems Principles (SOSP)*. Association for Computing Machinery, 1–18.

[41] Kexin Pei, Yinzhi Cao, Junfeng Yang, and Suman Jana. 2017. Towards Practical Verification of Machine Learning: The Case of Computer Vision Systems. *arXiv preprint: arXiv:1712.01785* 12 (2017).

[42] PyTorch. 2021. torchvision.models — Torchvision 0.10.0 documentation. https://pytorch.org/vision/stable/models.html.

[43] Aditi Raghunathan, Jacob Steinhardt, and Percy S Liang. 2018. Semidefinite Relaxations for Certifying Robustness to Adversarial Examples. In *2018 Advances in Neural Information Processing Systems (NeurIPS)*. Curran Associates, Inc., 10877–10887.

[44] Hadi Salman, Jerry Li, Ilya Razenshteyn, Pengchuan Zhang, Huan Zhang, Sebastian Bubeck, and Greg Yang. 2019. Provably Robust Deep Learning via Adversarially Trained Smoothed Classifiers. In *2019 Advances in Neural Information Processing Systems (NeurIPS)*. Curran Associates, Inc., 11289–11300.

[45] Hadi Salman, Greg Yang, Huan Zhang, Cho-Jui Hsieh, and Pengchuan Zhang. 2019. A Convex Relaxation Barrier to Tight Robustness Verification of Neural Networks. In *2019 Advances in Neural Information Processing Systems (NeurIPS)*. Curran Associates, Inc., 9835–9846.

[46] Pouya Samangouei, Maya Kabkab, and Rama Chellappa. 2018. Defense-GAN: Protecting Classifiers Against Adversarial Attacks Using Generative Models. In *2018 International Conference on Learning Representations (ICLR)*. OpenReview.

[47] Ali Shafahi, Mahyar Najibi, Mohammad Amin Ghiasi, Zheng Xu, John Dickerson, Christoph Studer, Larry S Davis, Gavin Taylor, and Tom Goldstein. 2019. Adversarial Training for Free!. In *2019 Advances in Neural Information Processing Systems (NeurIPS)*. Curran Associates, Inc., 3358–3369.

[48] David Silver, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang, Arthur Guez, Thomas Hubert, Lucas Baker, Matthew Lai, Adrian Bolton, et al. 2017. Mastering the Game of Go without Human Knowledge. *Nature* 550, 7676 (2017), 354–359.

[49] Gagandeep Singh, Timon Gehr, Markus Püschel, and Martin Vechev. 2019. An Abstract Domain for Certifying Neural Networks. *Proc. ACM Program. Lang.* 3, POPL, Article 41 (Jan. 2019), 30 pages.

[50] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. 2014. Intriguing Properties of Neural Networks. In *2014 International Conference on Learning Representations (ICLR)*. OpenReview.

[51] Jiaye Teng, Guang-He Lee, and Yang Yuan. 2020. ℓ_1 Adversarial Robustness Certificates: A Randomized Smoothing Approach. https://openreview.net/forum?id=H1lQIgrFDS

[52] Yuchi Tian, Kexin Pei, Suman Jana, and Baishakhi Ray. 2018. DeepTest: Automated Testing of Deep-Neural-Network-Driven Autonomous Cars. In *2018 International Conference on Software Engineering (ICSE)*. Association for Computing Machinery, 303–314.

[53] Vincent Tjeng, Kai Y Xiao, and Russ Tedrake. 2018. Evaluating Robustness of Neural Networks with Mixed Integer Programming. In *2018 International Conference on Learning Representations (ICLR)*. OpenReview.

[54] Liang Tong, Bo Li, Chen Hajaj, Chaowei Xiao, Ning Zhang, and Yevgeniy Vorobeychik. 2019. Improving Robustness of ML Classifiers against Realizable Evasion Attacks Using Conserved Features. In *2019 USENIX Conference on Security Symposium (USENIX Security)*. USENIX Association, 285–302.

[55] Florian Tramèr, Nicholas Carlini, Wieland Brendel, and Aleksander Madry. 2020. On Adaptive Attacks to Adversarial Example Defenses. In *2020 Advances in Neural Information Processing Systems (NeurIPS)*. Curran Associates, Inc., 1633–1645.

[56] Florian Tramèr, Alexey Kurakin, Nicolas Papernot, Ian Goodfellow, Dan Boneh, and Patrick McDaniel. 2018. Ensemble Adversarial Training: Attacks and Defenses. In *2018 International Conference on Learning Representations (ICLR)*. OpenReview.

[57] Yaxiao Wang, Yuanzhang Li, Quanxin Zhang, Jingjing Hu, and Xiaohui Kuang. 2019. Evading PDF Malware Classifiers with Generative Adversarial Network. In *2019 International Symposium on Cyberspace Safety and Security (CSS)*. Springer International Publishing, 374–387.

[58] Lily Weng, Huan Zhang, Hongge Chen, Zhao Song, Cho-Jui Hsieh, Luca Daniel, Duane Boning, and Inderjit Dhillon. 2018. Towards Fast Computation of Certified Robustness for ReLU Networks. In *2018 International Conference on Machine Learning (ICML)*. PMLR, 5276–5285.

[59] Eric Wong and Zico Kolter. 2018. Provable Defenses against Adversarial Examples via the Convex Outer Adversarial Polytope. In *2018 International Conference on Machine Learning (ICML)*. PMLR, 5286–5295.

[60] Eric Wong, Frank Schmidt, Jan Hendrik Metzen, and J Zico Kolter. 2018. Scaling Provable Adversarial Defenses. In *2018 Advances in Neural Information Processing Systems (NeurIPS)*. Curran Associates, Inc., 8400–8409.

[61] Chaowei Xiao, Bo Li, Jun Yan Zhu, Warren He, Mingyan Liu, and Dawn Song. 2018. Generating Adversarial Examples with Adversarial Networks. In *2018 International Joint Conference on Artificial Intelligence (IJCAI)*. International Joint Conferences on Artificial Intelligence Organization, 3905–3911.

[62] Chaowei Xiao, Jun-Yan Zhu, Bo Li, Warren He, Mingyan Liu, and Dawn Song. 2018. Spatially Transformed Adversarial Examples. In *2018 International Conference on Learning Representations (ICLR)*. OpenReview.

[63] Kaidi Xu, Zhouxing Shi, Huan Zhang, Yihan Wang, Kai-Wei Chang, Minlie Huang, Bhavya Kailkhura, Xue Lin, and Cho-Jui Hsieh. 2020. Automatic Perturbation Analysis for Scalable Certified Robustness and Beyond. In *2020 Advances in Neural Information Processing Systems (NeurIPS)*. Curran Associates, Inc., 1129–1141.

[64] Weilin Xu, Yanjun Qi, and David Evans. 2016. Automatically Evading Classifiers: A Case Study on PDF Malware Classifiers. In *2016 Network and Distributed Systems Symposium (NDSS)*, Vol. 10. Internet Society.

[65] Greg Yang, Tony Duan, J. Edward Hu, Hadi Salman, Ilya Razenshteyn, and Jerry Li. 2020. Randomized Smoothing of All Shapes and Sizes. In *2020 International Conference on Machine Learning (ICML)*. PMLR, 10693–10705.

[66] Runtian Zhai, Chen Dan, Di He, Huan Zhang, Boqing Gong, Pradeep Ravikumar, Cho-Jui Hsieh, and Liwei Wang. 2020. MACER: Attack-Free and Scalable Robust Training via Maximizing Certified Radius. In *2020 International Conference on Learning Representations (ICLR)*. OpenReview.

[67] Dinghuai Zhang, Mao Ye, Chengyue Gong, Zhanxing Zhu, and Qiang Liu. 2020. Black-Box Certification with Randomized Smoothing: A Functional Optimization Based Framework. In *2020 Advances in Neural Information Processing Systems (NeurIPS)*. Curran Associates, Inc., 2316–2326.

[68] Huan Zhang, Hongge Chen, Chaowei Xiao, Sven Gowal, Robert Stanforth, Bo Li, Duane Boning, and Cho-Jui Hsieh. 2020. Towards Stable and Efficient Training of Verifiably Robust Neural Networks. In *2020 International Conference on Learning Representations (ICLR)*. OpenReview.

Appendix A introduces our method for Lipschitz bound computation for scaling transformation. **Appendix B** introduces the certification procedure of **TSS** for common transformation compositions and discusses the extension to more compositions. **Appendix C** contains the proofs for **TSS** general framework, which is introduced in Section 4. **Appendix D** contains the theorem statements and proofs for the robustness conditions derived for common smoothing distributions. These statements are instantiations of Theorem 1, and serve for both certifying resolvable transformations and differentially resolvable transformations. **Appendix E** compares the closed-form expressions of the robustness radii derived for different smoothing distributions, which corresponds to Figure 4. **Appendix F** contains the proofs of robustness conditions for various resolvable transformations. **Appendix G** contains the proofs of general theorems for certifying differentially resolvable transformations. **Appendix H** formally defines the rotation and scaling transformations, two typical differentially resolvable transformations. **Appendix I** proves our supporting theorems for computing the interpolation bound, which is used when certifying differentially resolvable transformations. **Appendix J** contains concrete algorithm descriptions for certifying differentially resolvable transformations. Finally, **Appendix K** presents the omitted details in experiments, including experiment settings, detailed discussion of baseline approaches, implementation details, and additional results.

A COMPUTING THE LIPSCHITZ BOUND FOR SCALING TRANSFORMATION

The scaling transformation ϕ_S first stretches height and width of the input image by a factor $\alpha \in \mathbb{R}_+$ where values $\alpha < 1 (> 1)$ correspond to shrinking (enlarging) an image. Then, bilinear interpolation is applied, followed by black padding to determine pixel values. We refer the reader to Appendix H for a formal definition. Due to black padding, the functions g_i may contain discontinuities. To circumvent this issue, we enumerate all these discontinuities as $\mathcal{D}$. It can be shown that $\mathcal{D}$ contains at most $H + W$ elements. Hence, for large enough N, the interval $[\alpha_i, \alpha_{i+1}]$ contains at most one discontinuity. We thus modify the upper bounds M_i in (27) as

$$M_i := \begin{cases} \max\limits_{\alpha_i \leq \alpha \leq \alpha_{i+1}} \min\{g_i(\alpha), g_{i+1}(\alpha)\} & [\alpha_i, \alpha_{i+1}] \cap \mathcal{D} = \varnothing \\ \max\left\{\max\limits_{\alpha_i \leq \alpha \leq t_i} g_{i+1}(\alpha), \max\limits_{t_i \leq \alpha \leq \alpha_{i+1}} g_i(\alpha)\right\} & [\alpha_i, \alpha_{i+1}] \cap \mathcal{D} = \{t_i\} \end{cases}$$

(33)

In either case, the quantity M_i can again be bounded by a Lipschitz constant. With this definition, the following lemma provides a closed form expression for the Lipschitz constant L in (27) for scaling. A detailed proof is given in Appendix I.

LEMMA 5. *Let $x \in \mathbb{R}^{K \times W \times H}$ be a K-channel image and let ϕ_S be the scaling transformation. Then, a global Lipschitz constant L for the functions $\{g_i\}_{i=1}^N$ is given by*

$$L_s = \max_{1 \leq i \leq N-1} \sum_{k=0}^{K-1} \sum_{r,s \in \Omega \cap \mathbb{N}^2} \frac{\sqrt{2}d_{r,s}}{a^2} \cdot m_\Delta(x, k, \mathcal{P}_{r,s}^{(i)}) \cdot \overline{m}(x, k, \mathcal{P}_{r,s}^{(i)}) \quad (34)$$

where $\Omega = [0, W - 1] \times [0, H - 1]$ and a is the lower boundary value in $\mathcal{S} = [a, b]$. The set $\mathcal{P}_{r,s}^{(i)}$ is given by all integer grid pixels that are covered by the trajectory of source pixels of (r, s) when scaling with factors from α_{i+1} to α_i.

B CERTIFICATION OF TRANSFORMATION COMPOSITIONS

Here we state how **TSS** certifies typical transformation compositions in detail and discuss how **TSS** can be directly extended for providing robustness certificates of other transformations or their compositions.

B.1 Brightness and Contrast

As noted in Section 5.1.2, we certify the composition of brightness and contrast based on Lemma 2 and Lemma 3. To this end, we first obtain p_A, a lower bound of $q(y_A \mid x, \varepsilon_0)$ by Monte-Carlo sampling, where $\varepsilon_0 \sim \mathcal{N}(0, \text{diag}(\sigma^2, \tau^2))$ is the smoothing distribution. For the given $k, b \in \mathbb{R}$ that we would like to certify $g(\phi_{BC}(x, (k, b)^T); \varepsilon_0) = y_A$, we then trigger Lemma 2 to get $\tilde{p}_A$, a lower bound of $q(y_A \mid x, \varepsilon_1)$, and set $\tilde{p}_B = 1 - \tilde{p}_A$. Finally, we use the explicit condition in Lemma 3 to obtain the certification.

In the actual computation, instead of certifying a single pair (k, b), we usually certify the robustness against a set of transformation parameters

$$\mathcal{S}_{\text{adv}} = \{(k, b) \mid k \in [-k_0, k_0], b \in [-b_0, b_0]\}, \quad (35)$$

which stands for any contrast change within e^{k_0} and brightness change within b_0. It is infeasible to check every $(k, b) \in \mathcal{S}_{\text{adv}}$. To mitigate this, we relax the robustness condition in Lemma 3 from

$$\sqrt{(k/\sigma)^2 + (b/(e^{-k}\tau))^2} < \frac{1}{2}\left(\Phi^{-1}(\tilde{p}_A) - \Phi^{-1}(\tilde{p}_B)\right) \quad (36)$$

to

$$\sqrt{(k/\sigma)^2 + (b/(\min\{e^{-k}, 1\}\tau))^2} < \frac{1}{2}\left(\Phi^{-1}(\tilde{p}_A) - \Phi^{-1}(\tilde{p}_B)\right). \quad (37)$$

Thus, we only need to check the condition (37) for (k_0, b_0) and $(-k_0, b_0)$ to certify the robustness for any (k, b) in (35). This is because the LHS of (37) is monotonically increasing w.r.t. $|k|$ and $|b|$, and the RHS of (37) is equal to $\Phi^{-1}(\tilde{p}_A)$ that is monotonically decreasing w.r.t. $|k|$. Throughout the experiments, we use this strategy for certification of brightness and contrast.

B.2 Gaussian Blur, Brightness, Contrast, and Translation

The certification generally follows the same procedure as in certifying brightness and contrast. In the following, we first provide a formal definition of this transformation composition. Specifically, the transformation ϕ_{BTBC} is defined as:

$$\phi_{BTBC}(x, \alpha) := \phi_B(\phi_T(\phi_{BC}(x, \alpha_k, \alpha_b), \alpha_{Tx}, \alpha_{Ty}), \alpha_B), \quad (38)$$

where ϕ_B, ϕ_T and ϕ_{BC} are Gaussian blur, translation, and brightness and contrast transformations respectively as defined before; $\alpha := (\alpha_k, \alpha_b, \alpha_{Tx}, \alpha_{Ty}, \alpha_B)^T \in \mathbb{R}^4 \times \mathbb{R}_{\geq 0}$ is the transformation parameter.

Our certification relies on the following corollary (extended from Lemma 2) and lemma, which are proved in Appendix F.

COROLLARY 3. *Let $x \in \mathcal{X}$, $k \in \mathbb{R}$ and let $\varepsilon_0 := (\varepsilon_0^a, \varepsilon_0^b)^T$ be a random variable defined as*

$$\varepsilon_0^a \sim \mathcal{N}(0, \text{diag}(\sigma_k^2, \sigma_b^2, \sigma_T^2, \sigma_T^2)) \text{ and } \varepsilon_0^b \sim \text{Exp}(\lambda_B). \quad (39)$$

Similarly, let $\varepsilon_1 := (\varepsilon_1^a, \varepsilon_1^b)$ be a random variable with

$$\varepsilon_1^a \sim \mathcal{N}(0, \mathrm{diag}(\sigma_k^2, \, e^{-2k}\sigma_b^2, \, \sigma_T^2, \, \sigma_T^2)) \text{ and } \varepsilon_1^b \sim \mathrm{Exp}(\lambda_B). \quad (40)$$

For either random variable (denoted as ε), recall that $q(y|x; \varepsilon) := \mathbb{E}(p(y|\phi_{BTBC}(x, \varepsilon)))$. Suppose that $q(y|x; \varepsilon_0) \geq p$ for some $p \in [0, 1]$ and $y \in \mathcal{Y}$. Then $q(y|x; \varepsilon_1)$ satisfies Eq. (11).

LEMMA B.1. *Let ε_0 and ε_1 be as in Corollary 3 and suppose that*

$$q(y_A|x; \varepsilon_1) \geq \tilde{p}_A > \tilde{p}_B \geq \max_{y \neq y_A} q(y|x; \varepsilon_1). \quad (41)$$

Then it is guaranteed that $y_A = g(\phi_{BTBC}(x, \alpha); \varepsilon_0)$ as long as $p'_A > p'_B$, where

$$p'_A = \begin{cases} 0, & \text{if } \tilde{p}_A \leq 1 - \exp(-\lambda_B \alpha_B), \\[4pt] \Phi\left(\Phi^{-1}\left(1 - (1 - \tilde{p}_A)\exp(\lambda_B \alpha_B)\right) \right. \\[4pt] \left. - \sqrt{\alpha_k^2/\sigma_k^2 + \alpha_b^2/(e^{-2\alpha_k}\sigma_b^2) + (\alpha_{Tx}^2 + \alpha_{Ty}^2)/\sigma_T^2}\right) & \text{otherwise} \end{cases} \quad (42)$$

and

$$p'_B = \begin{cases} 1, & \text{if } \tilde{p}_B \geq \exp(-\lambda_B \alpha_B), \\[4pt] 1 - \Phi\left(\Phi^{-1}\left(1 - \tilde{p}_B \exp(\lambda_B \alpha_B)\right) \right. \\[4pt] \left. - \sqrt{\alpha_k^2/\sigma_k^2 + \alpha_b^2/(e^{-2\alpha_k}\sigma_b^2) + (\alpha_{Tx}^2 + \alpha_{Ty}^2)/\sigma_T^2}\right). & \text{otherwise} \end{cases} \quad (43)$$

The ε_0 specified by (39) is the smoothing distribution. Similar as in brightness and contrast certification, we first obtain p_A, a lower bound of $q(y_A | x, \varepsilon_0)$ by Monte-Carlo sampling. For a given transformation parameter $\alpha := (\alpha_k, \alpha_b, \alpha_{Tx}, \alpha_{Ty}, \alpha_B)^T$, we then trigger Corollary 3 to get $\tilde{p}_A$, a lower bound of $q(y_A | x, \varepsilon_1)$ and set $\tilde{p}_B = 1 - \tilde{p}_A$. Finally, we use the explicit condition in Lemma B.1 to obtain the certification. Indeed, with $\tilde{p}_B = 1 - \tilde{p}_A$, Lemma B.1 can be simplified to the following corollary.

COROLLARY 4. *Let ε_0 and ε_1 be as in Corollary 3 and suppose that*

$$q(y_A|x; \varepsilon_1) \geq \tilde{p}_A. \quad (44)$$

Then it is guaranteed that $y_A = g(\phi_{BTBC}(x, \alpha); \varepsilon_0)$ as long as

$$\tilde{p}_A > 1 - \exp(-\lambda_B \alpha_B)\left(1 - \Phi\left(\sqrt{\frac{\alpha_k^2}{\sigma_k^2} + \frac{\alpha_b^2}{e^{-2\alpha_k}\sigma_b^2} + \frac{\alpha_{Tx}^2 + \alpha_{Ty}^2}{\sigma_T^2}}\right)\right). \quad (45)$$

To certify against a set of transformation parameters

$$\mathcal{S}_{\mathrm{adv}} = \{(\alpha_k, \alpha_b, \alpha_{Tx}, \alpha_{Ty}, \alpha_B)^T \, | \\ \alpha_k \in [-k_0, k_0], \alpha_b \in [-b_0, b_0], \quad (46) \\ \|(\alpha_{Tx}, \alpha_{Ty})\|_2 \leq T, \alpha_B \leq B_0\},$$

we relax the robust condition in (45) to

$$\tilde{p}_A > 1 - \exp(-\lambda_B \alpha_B)\left(1 - \Phi\left(\sqrt{\frac{\alpha_k^2}{\sigma_k^2} + \frac{\alpha_b^2}{\min\{e^{-2\alpha_k}, 1\}\sigma_b^2} + \frac{\alpha_{Tx}^2 + \alpha_{Ty}^2}{\sigma_T^2}}\right)\right). \quad (47)$$

The LHS of Equation (47) is monotonically decreasing w.r.t. $|\alpha_k|$ and the RHS is monotonically increasing w.r.t. $|\alpha_k|$, $|\alpha_b|$, $\|(\alpha_{Tx}, \alpha_{Ty}\|_2$, and $|\alpha_B|$, and the RHS is symmetric w.r.t. α_b and $\|(\alpha_{Tx}, \alpha_{Ty})\|_2$. As a result, we only need to check the condition for $(-k_0, b_0, T, 0, B_0)$ and $(k_0, b_0, T, 0, B_0)$ to certify the entire set $\mathcal{S}_{\mathrm{adv}}$. Throughout the experiments, we use this strategy for certification.

B.3 Scaling/Rotation and Brightness

To certify the composition of scaling and brightness or rotation and brightness, we follow the same methodology as certifying scaling or rotation alone and reuse the computed interpolation error $M_{\mathcal{S}}$. We only make the following two changes: (1) alter the smoothing distribution from additive Gaussian noise $\psi(x, \delta) = x + \delta$ where $\delta \sim \mathcal{N}(0, \sigma^2 \mathbb{1}_d)$ to additive Gaussian noise and Gaussian brightness change $\psi(x, \delta, \delta_b) = x + \delta + b \cdot \mathbb{1}_d$ where $\delta \sim \mathcal{N}(0, \sigma^2 \mathbb{1}_d), b \sim \mathcal{N}(0, \sigma_b^2)$; (2) change the robustness condition from $R > M_{\mathcal{S}}$ in Corollary 2 to $R > \sqrt{M_{\mathcal{S}}^2 + (\sigma^2/\sigma_b^2)b_0^2}$. We formalize this robustness condition in the following corollary, and the proof is entailed in Appendix G.

COROLLARY 5. *Let $\psi_B(x, \delta, b) = x + \delta + b \cdot \mathbb{1}_d$ and let $\varepsilon \sim \mathcal{N}(0, \sigma^2 \mathbb{1}_d), \varepsilon_b \sim \mathcal{N}(0, \sigma_b^2)$. Furthermore, let ϕ be a transformation with parameters in $\mathcal{Z}_\phi \subseteq \mathbb{R}^m$ and let $\mathcal{S} \subseteq \mathcal{Z}_\phi$ and $\{\alpha_i\}_{i=1}^N \subseteq \mathcal{S}$. Let $y_A \in \mathcal{Y}$ and suppose that for any i, the $(\varepsilon, \varepsilon_b)$-smoothed classifier $q(y | x; \varepsilon, \varepsilon_b) := \mathbb{E}(p(y| \psi_B(x, \varepsilon, \varepsilon_b))$ satisfies*

$$q(y_A|x; \varepsilon, \varepsilon_b) \geq p_A^{(i)} > p_B^{(i)} \geq \max_{y \neq y_A} q(y|x; \varepsilon, \varepsilon_b). \quad (48)$$

for each i. Let

$$R := \frac{\sigma}{2} \min_{1 \leq i \leq N} \left(\Phi^{-1}\left(p_A^{(i)}\right) - \Phi^{-1}\left(p_B^{(i)}\right)\right) \quad (49)$$

Then, $\forall \alpha \in \mathcal{S}$ and $\forall b \in [-b_0, b_0]$ it is guaranteed that $y_A = \arg\max_y q(y| \phi(x, \alpha) + b \cdot \mathbb{1}_d; \varepsilon, \varepsilon_b)$ as long as

$$R > \sqrt{M_{\mathcal{S}}^2 + \frac{\sigma^2}{\sigma_b^2}b_0^2}, \quad (50)$$

where $M_{\mathcal{S}}$ is defined as in Corollary 2.

B.4 Scaling/Rotation, Brightness, and ℓ_2 Perturbations

We use the same smoothing distribution as above, and the following corollary directly allows us to certify the robustness against the composition of scaling/rotation, brightness, and an additional ℓ_2-bounded perturbations—we only need to change the robustness condition from $R > \sqrt{M_{\mathcal{S}}^2 + (\sigma^2/\sigma_b^2)b_0^2}$ to $R > \sqrt{(M_{\mathcal{S}} + r)^2 + (\sigma^2/\sigma_b^2)b_0^2}$. The proof is given in Appendix G.

COROLLARY 6. *Under the same setting as in Corollary 5, for $\forall \alpha \in \mathcal{S}$, $\forall b \in [-b_0, b_0]$ and $\forall \delta \in \mathbb{R}^d$ such that $\|\delta\|_2 \leq r$, it is guaranteed that $y_A = \arg\max_k q(y| \phi(x, \alpha) + b \cdot \mathbb{1}_d + \delta; \varepsilon, \varepsilon_d)$ as long as*

$$R > \sqrt{(M_{\mathcal{S}} + r)^2 + \frac{\sigma^2}{\sigma_b^2}b_0^2}, \quad (51)$$

where $M_{\mathcal{S}}$ is defined as in Corollary 2.

B.5 Discussion on More Transformations and Compositions

Our **TSS** is not limited to specific transformations. Here we briefly discuss how to extend **TSS** for any new transformations or new compositions.

For a new transformation, we first identify the parameter space $\mathcal{Z}$, where the identified parameter should completely and deterministically decide the output after transformation for any given input. Then, we use Definition 2 to check whether the transformation is resolvable. If so, we can write down the function γ_α. Next, we choose a smoothing distribution, i.e., the distribution of the random variable ε_0, and identify the distribution of $\varepsilon_1 = \gamma_\alpha(\varepsilon_0)$. Finally, we use Theorem 1 to derive the robustness certificates and follow the two-step template (Section 5.1.5) to compute the robustness certificate.

If the transformation is not resolvable, we identify a dimension in $\mathcal{Z}$ for which the transformation is *resolvable*. For example, the composition of rotation and brightness has a rotation and a brightness axis, where the brightness axis is itself resolvable. As a result, we can write the parameter space as Cartesian product of non-resolvable subspace and resolvable subspace: $\mathcal{Z} := \mathcal{Z}_{\text{no-resolve}} \times \mathcal{Z}_{\text{resolve}}$. We perform smoothing on the resolvable subspace and sample enough points in the non-resolvable subspace. Next, we bound the interpolation error between sampled points and arbitrary points in the non-resolvable subspace, using either ℓ_p difference as we did for rotation and scaling or other regimes. Specifically, our Lemma 9 shown in Appendix I is a useful tool to bound ℓ_p difference caused by interpolation error. Finally, we instantiate Theorem 2 to compute the robustness certificate.

Theoretically, we can certify against the composition of all the discussed transformations: Gaussian blur, brightness, contrast, translation, rotation, and scaling. However, as justified in [23, Figure 3], the composition of more than two transformations leads to unrealistic images that are even hard to distinguish by humans. Moreover, if the composition contains too many transformations, the parameter space would be no longer low dimensional. Therefore, there would be much more axes that are differentially resolvable (instead of resolvable). As a consequence, much more samples are required to obtain a small bound on interpolation error (which is necessary for a nontrivial robustness certification). Therefore, we mainly evaluate on single transformation or composition of two transformations to simulate a practical attack.

C PROOFS FOR THE GENERAL CERTIFICATION FRAMEWORK

Here we provide the proof for Theorem 1. For that purpose, recall the following definition from the main part of this paper:

Definition 1 (restated). *Let $\phi: \mathcal{X} \times \mathcal{Z} \to \mathcal{X}$ be a transformation, $\varepsilon \sim \mathbb{P}_\varepsilon$ a random variable taking values in $\mathcal{Z}$ and let $h: \mathcal{X} \to \mathcal{Y}$ be a base classifier. We define the ε-smoothed classifier $g: \mathcal{X} \to \mathcal{Y}$ as $g(x; \varepsilon) = \arg\max_{y \in \mathcal{Y}} q(y \mid x; \varepsilon)$ where q is given by the expectation with respect to the smoothing distribution ε, i.e.,*

$$q(y \mid x; \varepsilon) := \mathbb{E}(p(y \mid \phi(x, \varepsilon))). \tag{52}$$

Here, we additionally define the notion of level sets separately. These sets originate from statistical hypothesis testing correspond to rejection regions of likelihood ratio tests.

Definition 7 (Lower level sets). *Let $\varepsilon_0 \sim \mathbb{P}_0$, $\varepsilon_1 \sim \mathbb{P}_1$ be $\mathcal{Z}$-valued random variables with probability density functions f_0 and f_1 with respect to a measure μ. For $t \geq 0$ we define lower and strict lower level sets as*

$$\underline{S_t} := \{z \in \mathcal{Z}: \Lambda(z) < t\}, \quad \overline{S_t} := \{z \in \mathcal{Z}: \Lambda(z) \leq t\},$$

$$\text{where} \quad \Lambda(z) := \frac{f_1(z)}{f_0(z)}. \tag{53}$$

We also make the following definition in order to reduce clutter and simplify the notation. This definition will be used throughout the proofs presented here.

Definition 8 ((p_A, p_B)-Confident Classifier). *Let $x \in \mathcal{X}$, $y_A \in \mathcal{Y}$ and $p_A, p_B \in [0, 1]$ with $p_A > p_B$. We say that the ε-smoothed classifier q is (p_A, p_B)-confident at x if*

$$q(y_A \mid x; \varepsilon) \geq p_A \geq p_B \geq \max_{y \neq y_A} q(y \mid x; \varepsilon). \tag{54}$$

C.1 Auxiliary Lemmas

Lemma 6. *Let ε_0 and ε_1 be random variables taking values in $\mathcal{Z}$ and with probability density functions f_0 and f_1 with respect to a measure μ. Denote by Λ the likelihood ratio $\Lambda(z) = f_1(z)/f_0(z)$. For $p \in [0, 1]$ let $\tau_p := \inf\{t \geq 0: \mathbb{P}_0(\overline{S_t}) \geq p\}$. Then, it holds that*

$$\mathbb{P}_0\left(\underline{S_{\tau_p}}\right) \leq p \leq \mathbb{P}_0\left(\overline{S_{\tau_p}}\right). \tag{55}$$

Proof. We first show the RHS of inequality (55). This follows directly from the definition of τ_p if we show that the function $t \mapsto \mathbb{P}_0\left(\overline{S_t}\right)$ is right-continuous. For that purpose, let $t \geq 0$ and let $\{t_n\}_n$ be a sequence in $\mathbb{R}_{\geq 0}$ such that $t_n \downarrow t$. Define the sets $A_n := \{z: \Lambda(z) \leq t_n\}$ and note that $A_{n+1} \subseteq A_n$. Clearly, if $z \in \overline{S_t}$, then $\forall n: \Lambda(z) \leq t \leq t_n$, thus $z \in \cap_n A_n$ and hence $\overline{S_t} \subseteq \cap_n A_n$. If on the other hand $z \in \cap_n A_n$, then $\forall n: \Lambda(z) \leq t_n \to t$ as $n \to \infty$ and thus $z \in \overline{S_t}$, yielding $\overline{S_t} = \cap_n A_n$. Hence for any $t \geq 0$ we have that

$$\lim_{n \to \infty} \mathbb{P}_0(A_n) = \mathbb{P}_0\left(\bigcap_n A_n\right) = \mathbb{P}_0\left(\underline{S_t}\right). \tag{56}$$

Thus, the function $t \mapsto \mathbb{P}_0\left(\overline{S_t}\right)$ is right continuous and in particular it follows that $\mathbb{P}_0\left(\overline{S_{\tau_p}}\right) \geq p$. We now show the LHS of inequality (55). Consider the sets $B_n := \{z: \Lambda(z) < \tau_p - 1/n\}$ and note that $B_n \subseteq B_{n+1}$. Clearly, if $z \in \cup_n B_n$, then $\exists n$ such that $\Lambda(z) < \tau_p - 1/n < \tau_p$ and thus $z \in \underline{S_{\tau_p}}$. If on the other hand $z \in \underline{S_{\tau_p}}$, then we can choose n large enough such that $\Lambda(z) < \tau_p - 1/n$ and thus $z \in \cup_n B_n$ yielding $\underline{S_{\tau_p}} = \cup_n B_n$. Furthermore, by the definition of τ_p and since for any $n \in \mathbb{N}$ we have that $\mathbb{P}_0(B_n) = \mathbb{P}_0\left(\underline{S}(\tau_p - 1/n)\right) < p$ it follows that

$$\mathbb{P}_0\left(\underline{S_{\tau_p}}\right) = \mathbb{P}_0\left(\bigcup_n B_n\right) = \lim_{n \to \infty} \mathbb{P}_0(B_n) \leq p \tag{57}$$

concluding the proof. $\qquad\square$

Lemma 7. *Let ε_0 and ε_1 be random variables taking values in $\mathcal{Z}$ and with probability density functions f_0 and f_1 with respect to a measure μ. Let $h: \mathcal{Z} \to [0, 1]$ be a deterministic function. Then, for any $t \geq 0$ the following implications hold:*

(i) For any $S \subseteq \mathcal{Z}$ with $\underline{S_t} \subseteq S \subseteq \overline{S_t}$ the following implication holds:

$$\mathbb{E}[h(\varepsilon_0)] \geq \mathbb{P}_0(S) \Rightarrow \mathbb{E}[h(\varepsilon_1)] \geq \mathbb{P}_1(S). \tag{58}$$

(i) For any $S \subseteq \mathcal{Z}$ with $\overline{S_t}^c \subseteq S \subseteq \underline{S_t}^c$ the following implication holds:

$$\mathbb{E}[h(\varepsilon_0)] \leq \mathbb{P}_0(S) \Rightarrow \mathbb{E}[h(\varepsilon_1)] \leq \mathbb{P}_1(S). \tag{59}$$

PROOF. We first prove (i). For that purpose, consider

$$\mathbb{E}[f(\varepsilon_1)] - \mathbb{P}_1(S) = \int hf_1 \, d\mu - \int_S f_1 \, d\mu \tag{60}$$

$$= \int_{S^c} hf_1 \, d\mu - \left(\int_S (1-h)f_1 \, d\mu \right) \tag{61}$$

$$= \int_{S^c} h\Lambda f_0 \, d\mu - \left(\int_S (1-h)\Lambda f_0 \, d\mu \right) \tag{62}$$

$$\geq t \cdot \int_{S^c} hf_0 \, d\mu - t \cdot \left(\int_S (1-h)f_0 \, d\mu \right) \tag{63}$$

$$= t \cdot \left(\int hf_0 \, d\mu - \int_S f_0 \, d\mu \right) \tag{64}$$

$$= t \cdot (\mathbb{E}[f(\varepsilon_0)] - \mathbb{P}_0(S)) \geq 0. \tag{65}$$

The inequality in (63) follows from the fact that whenever $z \in S^c$, then $f_1(z) \geq t \cdot f_0(z)$ and if $z \in S$, then $f_1(z) \leq t \cdot f_0(z)$ since S is a lower level set. Finally, the inequality in (65) follows from the assumption. The proof of (ii) is analogous and omitted here. □

C.2 Proof of Theorem 1

Theorem 1 (restated). *Let $\varepsilon_0 \sim \mathbb{P}_0$ and $\varepsilon_1 \sim \mathbb{P}_1$ be $\mathcal{Z}$-valued random variables with probability density functions f_0 and f_1 with respect to a measure μ on $\mathcal{Z}$ and let $\phi \colon X \times \mathcal{Z} \to X$ be a semantic transformation. Suppose that $y_A = g(x; \varepsilon_0)$ and let $p_A, p_B \in [0, 1]$ be bounds to the class probabilities, i.e.,*

$$q(y_A | x, \varepsilon_0) \geq p_A > p_B \geq \max_{y \neq y_A} q(y | x, \varepsilon_0). \tag{66}$$

For $t \geq 0$, let $\underline{S_t}, \overline{S_t} \subseteq \mathcal{Z}$ be the sets defined as $\underline{S_t} := \{f_1/f_0 < t\}$ and $\overline{S_t} := \{f_1/f_0 \leq t\}$ and define the function $\xi \colon [0, 1] \to [0, 1]$ by

$$\xi(p) := \sup\{\mathbb{P}_1(S) : \underline{S}_{\tau_p} \subseteq S \subseteq \overline{S}_{\tau_p}\} \tag{67}$$

$$\text{where} \quad \tau_p := \inf\{t \geq 0 : \mathbb{P}_0(\overline{S_t}) \geq p\}.$$

If the condition

$$\xi(p_A) + \xi(1 - p_B) > 1 \tag{68}$$

is satisfied, then it is guaranteed that $g(x; \varepsilon_1) = g(x; \varepsilon_0)$.

PROOF. For ease of notation, let ζ be the function defined by

$$t \mapsto \zeta(t) := \mathbb{P}_0(\overline{S_t}) \tag{69}$$

and notice that $\tau_p = \zeta^{-1}(p)$ where ζ^{-1} denotes the generalized inverse of ζ. Furthermore, let $\tau_A := \tau_{p_A}$, $\tau_B := \tau_{1-p_B}$, $\underline{S_A} := \underline{S}(\tau_A)$, $\underline{S_B} := \underline{S}(\tau_B)$, $\overline{S_A} := \overline{S}(\tau_A)$ and $\overline{S_B} := \overline{S}(\tau_B)$. We first show that $q(y_A | x, \varepsilon_1)$ is lower bounded by $\xi(\tau_A)$. For that purpose, note that by Lemma 6 we have that $\zeta(\tau_A) = \mathbb{P}_0(\overline{S_A}) \geq p_A \geq \mathbb{P}_0(\underline{S_A})$. Thus, the collection of sets

$$\mathcal{S}_A := \{S \subseteq \mathcal{Z} : \underline{S_A} \subseteq S \subseteq \overline{S_A}, \mathbb{P}_0(S) \leq p_A\} \tag{70}$$

is not empty. Pick some $A \in \mathcal{S}_A$ arbitrary and note that, since by assumption $g(\cdot; \varepsilon_0)$ is (p_A, p_B)-confident at x it holds that

$$\mathbb{E}(p(y_A | \phi(x, \varepsilon_0))) = q(y_A | x; \varepsilon_0) \geq p_A \geq \mathbb{P}_0(A). \tag{71}$$

Since $\underline{S_A} \subseteq A \subseteq \overline{S_A}$ we can apply part (i) of Lemma 7 and obtain the lower bound

$$q(y_A | x, \varepsilon_1) = \mathbb{E}(p(y_A | \phi(x, \varepsilon_1))) \geq \mathbb{P}_1(A). \tag{72}$$

Since $A \in \mathcal{S}_A$ was arbitrary, we take the sup over all $A \in \mathcal{S}_A$ and obtain

$$q(y_A | x; \varepsilon_1) \geq \sup_{A \in \mathcal{S}_A} \mathbb{P}_1(A) = \xi(p_A) \tag{73}$$

We now show that for any $y \neq y_A$ the prediction $q(y | x; \varepsilon_1)$ is upper bounded by $1 - \xi(1 - p_B)$. For that purpose, note that by Lemma 6 we have that $\zeta(\tau_B) = \mathbb{P}_0(\overline{S_A}) \geq 1 - p_B \geq \mathbb{P}_0(\underline{S_B})$. Thus, the collection of sets

$$\mathcal{S}_B := \{S \subseteq \mathcal{Z} : \underline{S_B} \subseteq S \subseteq \overline{S_B}, \mathbb{P}_0(S) \leq 1 - p_B\} \tag{74}$$

is not empty. Pick some $B \in \mathcal{S}_A$ arbitrary and note that, since by assumption $g(\cdot; \varepsilon_0)$ is (p_A, p_B)-confident at x it holds that

$$\mathbb{E}(p(y | \phi(x, \varepsilon_0))) = q(y | x; \varepsilon_0) \leq p_B$$
$$= 1 - (1 - p_B) \leq 1 - \mathbb{P}_0(B). \tag{75}$$

Since $\underline{S_B}^c \subseteq B^c \subseteq \overline{S_B}^c$ we can apply part (ii) of Lemma 7 and obtain the upper bound

$$q(y | x; \varepsilon_1) = \mathbb{E}(p(y | \phi(x, \varepsilon_1))) \leq 1 - \mathbb{P}_1(B). \tag{76}$$

Since $B \in \mathcal{S}_B$ was arbitrary, we take the inf over all $B \in \mathcal{S}_B$ and obtain

$$q(y | x; \varepsilon_1) \leq \inf_{B \in \mathcal{S}_B} (1 - \mathbb{P}_1(B)) = 1 - \xi(1 - p_B). \tag{77}$$

Combining together (77) and (73), we find that, whenever

$$\xi(p_A) + \xi(1 - p_B) > 1 \tag{78}$$

it is guaranteed that

$$q(y_A | x; \varepsilon_1) > \max_{y \neq y_A} q(y | x; \varepsilon_1) \tag{79}$$

what concludes the proof. □

D PROOFS FOR CERTIFICATION WITH DIFFERENT SMOOTHING DISTRIBUTIONS

Here, we instantiate Theorem 1 with different smoothing distributions and solve the robustness condition (6) for the case where the distribution of ε_1 results from shifting the distribution of ε_0, i.e., $\varepsilon_1 = \alpha + \varepsilon_0$. For ease of notation, let $\zeta \colon \mathbb{R}_{\geq 0} \to [0, 1]$ be the function defined by

$$t \mapsto \zeta(t) := \mathbb{P}_0\left(\overline{S_t}\right) \tag{80}$$

where $\mathbb{P}_0$ is the distribution of ε_0 and $\overline{S_t}$ is a lower level set; recall that the definitions of lower level sets is

$$\underline{S_t} := \{z \in \mathcal{Z} : \Lambda(z) < t\}, \quad \overline{S_t} := \{z \in \mathcal{Z} : \Lambda(z) \leq t\},$$

$$\text{where} \quad \Lambda(z) := \frac{f_1(z)}{f_0(z)}. \tag{81}$$

Note that the generalized inverse of ζ corresponds to τ_p, i.e.,

$$\zeta^{-1}(p) = \inf\{t \geq 0 | \zeta(t) \geq p\} = \tau_p \tag{82}$$

and the function ξ is correspondingly given by

$$\xi(p) = \sup\{\mathbb{P}_1(S) | \underline{S}(\zeta^{-1}(p)) \subseteq S \subseteq \overline{S}(\zeta^{-1}(p))\} \tag{83}$$

D.1 Gaussian Smoothing

COROLLARY 7. *Suppose* $\mathcal{Z} = \mathbb{R}^m$, $\Sigma := \mathrm{diag}(\sigma_1^2, \ldots, \sigma_m^2)$, $\varepsilon_0 \sim \mathcal{N}(0, \Sigma)$ *and* $\varepsilon_1 := \alpha + \varepsilon_0$ *for some* $\alpha \in \mathbb{R}^m$. *Suppose that the* ε_0*-smoothed classifier* g *is* (p_A, p_B)*-confident at* $x \in \mathcal{X}$ *for some* $y_A \in \mathcal{Y}$. *Then, it holds that* $q(y_A | x; \varepsilon_1) > \max_{y \neq y_A} q(y | x; \varepsilon_1)$ *if* α *satisfies*

$$\sqrt{\sum_{i=1}^m \left(\frac{\alpha_i}{\sigma_i}\right)^2} < \frac{1}{2}\left(\Phi^{-1}(p_A) - \Phi^{-1}(p_B)\right). \tag{84}$$

PROOF. By Theorem 1 we know that if ε_1 satisfies

$$\xi(p_A) + \xi(1 - p_B) > 1, \tag{85}$$

then it is guaranteed that

$$q(y_A | x; \varepsilon_1) > \max_{y \neq y_A} q(y | x; \varepsilon_1). \tag{86}$$

The proof is thus complete if we show that (85) reduces to (84). For that purpose, denote by f_0 and f_1 density functions of ε_0 and ε_1, respectively. Let $A := \Sigma^{-1}$ and note that the bilinear form $(z_1, z_2) \mapsto z_1^T A z_2 =: \langle z_1, z_2 \rangle_A$ defines an inner product on $\mathbb{R}^m$. Let $z \in \mathbb{R}^m$ and consider

$$\Lambda(z) = \frac{f_1(z)}{f_0(z)} = \frac{\exp\left(-\frac{1}{2}\langle z - \alpha, z - \alpha \rangle_A\right)}{\exp\left(-\frac{1}{2}\langle z, z \rangle_A\right)} \tag{87}$$

$$= \exp\left(\langle z, \alpha \rangle_A - \frac{1}{2}\langle \alpha, \alpha \rangle_A\right) \tag{88}$$

and thus

$$\Lambda(z) \leq t \iff \langle z, \alpha \rangle_A \leq \log(t) + \frac{1}{2}\langle \alpha, \alpha \rangle. \tag{89}$$

Let $Z \sim \mathcal{N}(0, 1)$ and notice that

$$\frac{\langle \varepsilon_0, \alpha \rangle_A}{\sqrt{\langle \alpha, \alpha \rangle_A}} \overset{d}{=} Z \overset{d}{=} \frac{\langle \varepsilon_1, \alpha \rangle_A - \langle \alpha, \alpha \rangle_A}{\sqrt{\langle \alpha, \alpha \rangle_A}}. \tag{90}$$

Let $\partial_t := \overline{S}_t \setminus \underline{S}_t = \{z \colon \Lambda(z) = t\}$ and notice that $\mathbb{P}_0(\partial_t) = \mathbb{P}_1(\partial_t) = 0$ and $\mathbb{P}_0(\underline{S}_t) = \mathbb{P}_0(\overline{S}_t)$. Similarly, it holds that $\mathbb{P}_1(\underline{S}_t) = \mathbb{P}_1(\overline{S}_t)$. The function $p \mapsto \xi(p)$ is thus given by

$$\xi(p) = \mathbb{P}_1\left(\overline{S}_{\zeta^{-1}(p)}\right). \tag{91}$$

We compute ζ as

$$\zeta(t) = \mathbb{P}\left(\Lambda(\varepsilon_0) \leq t\right) = \mathbb{P}\left(\langle \varepsilon_0, \alpha \rangle_A \leq \log(t) + \frac{1}{2}\langle \alpha, \alpha \rangle_A\right) \tag{92}$$

$$= \Phi\left(\frac{\log(t) + \frac{1}{2}\langle \alpha, \alpha \rangle_A}{\sqrt{\langle \alpha, \alpha \rangle_A}}\right) \tag{93}$$

and for $p \in [0, 1]$ its inverse is

$$\zeta^{-1}(p) = \exp\left(\Phi^{-1}(p)\sqrt{\langle \alpha, \alpha \rangle_A} - \frac{1}{2}\langle \alpha, \alpha \rangle_A\right). \tag{94}$$

Thus

$$\mathbb{P}\left(\Lambda(\varepsilon_1) \leq \zeta^{-1}(p)\right) \tag{95}$$

$$= \mathbb{P}\left(\frac{\langle \varepsilon_1, \alpha \rangle_A - \langle \alpha, \alpha \rangle_A}{\sqrt{\langle \alpha, \alpha \rangle_A}} \leq \frac{\log(\zeta^{-1}(p)) - \frac{1}{2}\langle \alpha, \alpha \rangle_A}{\sqrt{\langle \alpha, \alpha \rangle_A}}\right) \tag{96}$$

$$= \Phi\left(\frac{\left(\Phi^{-1}(p)\sqrt{\langle \alpha, \alpha \rangle_A} - \frac{1}{2}\langle \alpha, \alpha \rangle_A\right) - \frac{1}{2}\langle \alpha, \alpha \rangle_A}{\sqrt{\langle \alpha, \alpha \rangle_A}}\right) \tag{97}$$

$$= \Phi\left(\Phi^{-1}(p) - \sqrt{\langle \alpha, \alpha \rangle_A}\right). \tag{98}$$

Finally, algebra shows that

$$\Phi\left(\Phi^{-1}(p_A) - \sqrt{\langle \alpha, \alpha \rangle_A}\right) + \Phi\left(\Phi^{-1}(1 - p_B) - \sqrt{\langle \alpha, \alpha \rangle_A}\right) > 1 \tag{99}$$

is equivalent to

$$\sqrt{\sum_{i=1}^m \left(\frac{\alpha_i}{\sigma_i}\right)^2} < \frac{1}{2}\left(\Phi^{-1}(p_A) - \Phi^{-1}(p_B)\right) \tag{100}$$

what concludes the proof. □

D.2 Exponential Smoothing

COROLLARY 8. *Suppose* $\mathcal{Z} = \mathbb{R}_{\geq 0}^m$, *fix some* $\lambda > 0$ *and let* $\varepsilon_{0,i} \overset{\text{iid}}{\sim} \mathrm{Exp}(1/\lambda)$, $\varepsilon_0 := (\varepsilon_{0,1}, \ldots, \varepsilon_{0,m})^T$ *and* $\varepsilon_1 := \alpha + \varepsilon_0$ *for some* $\alpha \in \mathbb{R}_{\geq 0}^m$. *Suppose that the* ε_0*-smoothed classifier* g *is* (p_A, p_B)*-confident at* $x \in \mathcal{X}$ *for some* $y_A \in \mathcal{Y}$. *Then, it holds that* $q(y_A | x; \varepsilon_1) > \max_{y \neq y_A} q(y | x; \varepsilon_1)$ *if* α *satisfies*

$$\|\alpha\|_1 < -\frac{\log(1 - p_A + p_B)}{\lambda}. \tag{101}$$

PROOF. By Theorem 1 we know that if ε_1 satisfies

$$\xi(p_A) + \xi(1 - p_B) > 1, \tag{102}$$

then it is guaranteed that

$$q(y_A | x; \varepsilon_1) > \max_{y \neq y_A} q(y | x; \varepsilon_1). \tag{103}$$

The proof is thus complete if we show that (102) reduces to (101). For that purpose, denote by f_0 and f_1 density functions of ε_0 and ε_1, respectively, and note that

$$f_1(z) = \begin{cases} \lambda \cdot \exp(-\lambda\|z - \alpha\|_1), & \min_i(z_i - \alpha_i) \geq 0, \\ 0, & \text{otherwise}, \end{cases} \tag{104}$$

$$f_0(z) = \begin{cases} \lambda \cdot \exp(-\lambda\|z\|_1), & \min_i(z_i) \geq 0, \\ 0, & \text{otherwise}, \end{cases} \tag{105}$$

and $\forall i, z_i - \alpha_i \leq z_i$ and hence $f_0(z) = 0 \Rightarrow f_1(z) = 0$. Thus

$$\Lambda(z) = \frac{f_1(z)}{f_0(z)} = \begin{cases} \exp\left(\lambda \cdot \|\alpha\|_1\right) & \min_i(z_i - \alpha_i) \geq 0, \\ 0, & \text{otherwise}. \end{cases} \tag{106}$$

Let $S_0 := \{z \in \mathbb{R}_{\geq 0}^m \colon \min_i(z_i - \alpha_i) < 0\}$ and note that due to independence

$$\mathbb{P}_0(S_0) = \mathbb{P}\left(\bigcup_{i=1}^m \{\varepsilon_{0,i} < \alpha_i\}\right) \tag{107}$$

$$= 1 - \mathbb{P}\left(\bigcap_{i=1}^m \{\varepsilon_{0,i} \geq \alpha_i\}\right) = 1 - \prod_{i=1}^m \mathbb{P}\left(\varepsilon_{0,i} \geq \alpha_i\right) \tag{108}$$

$$= 1 - \prod_{i=1}^m \left(1 - (1 - \exp(-\lambda\,\alpha_i))\right) \tag{109}$$

$$= 1 - \exp(-\lambda\|\alpha\|_1). \tag{110}$$

Let $t_\alpha := \exp(\lambda\|\alpha\|_1)$ and compute ζ as

$$\zeta(t) = \mathbb{P}(\Lambda(\varepsilon_0) \leq t) \tag{111}$$

$$= \mathbb{P}\left(\mathbb{1}\{\min_i(\varepsilon_{0,i} - \alpha_i) \geq 0\} \leq t \cdot \exp(-\lambda\|\alpha\|_1)\right) \tag{112}$$

$$= \begin{cases} 1 - \exp(-\lambda\|\alpha\|_1) & t < t_\alpha, \\ 1 & t \geq t_\alpha. \end{cases} \tag{113}$$

Recall that $\zeta^{-1}(p) := \inf\{t \geq 0 \colon \zeta(t) \geq p\}$ for $p \in [0, 1]$ and hence

$$\zeta^{-1}(p) = \begin{cases} 0 & p \leq 1 - \exp(-\lambda\|\alpha\|_1), \\ \exp(\lambda\|\alpha\|_1) & p > 1 - \exp(-\lambda\|\alpha\|_1). \end{cases} \tag{114}$$

In order to evaluate ξ we compute the lower and strict lower level sets at $t = \zeta^{-1}(p)$. Recall that $\underline{S}_t = \{z \in \mathbb{R}^m_{\geq 0} \colon \Lambda(z) < t\}$ and $\overline{S}_t = \{z \in \mathbb{R}^m_{\geq 0} \colon \Lambda(z) \leq t\}$ and consider

$$\begin{aligned} \underline{S}_{\zeta^{-1}(p)} &= \left(S_0^c \cap \{z \in \mathbb{R}^m_{\geq 0} \colon \; \exp(\lambda\|\alpha\|_1) < \zeta^{-1}(p)\}\right) \\ &\quad \cup \left(S_0 \cap \{z \in \mathbb{R}^m_{\geq 0} \mid 0 < \zeta^{-1}(p)\}\right) \end{aligned} \tag{115}$$

$$= \begin{cases} \varnothing & p \leq 1 - \exp(-\lambda\|\alpha\|_1), \\ S_0 & p > 1 - \exp(-\lambda\|\alpha\|_1) \end{cases} \tag{116}$$

and

$$\begin{aligned} \overline{S}_{\zeta^{-1}(p)} &= \left(S_0^c \cap \{z \in \mathbb{R}^m_{\geq 0} \colon \; \exp(\lambda\|\alpha\|_1) \leq \zeta^{-1}(p)\}\right) \\ &\quad \dot\cup \left(S_0 \cap \{z \in \mathbb{R}^m_{\geq 0} \colon \; 0 \leq \zeta^{-1}(p)\}\right) \end{aligned} \tag{117}$$

$$= \begin{cases} S_0 & p \leq 1 - \exp(-\lambda\|\alpha\|_1), \\ \mathbb{R}^m_+ & p > 1 - \exp(-\lambda\|\alpha\|_1). \end{cases} \tag{118}$$

Suppose that $p \leq 1 - \exp(-\lambda\|\alpha\|_1)$. Then we have that $\underline{S}_{\zeta^{-1}(p)} = \varnothing$ and $\overline{S}_{\zeta^{-1}(p)} = S_0$ and hence

$$\begin{aligned} &p \leq 1 - \exp(-\lambda\|\alpha\|_1) \\ &\Rightarrow \xi(p) = \sup\{\mathbb{P}_1(S) \colon S \subseteq S_0 \wedge \mathbb{P}_0(S) \leq p\} = 0. \end{aligned} \tag{119}$$

Condition (102) can thus be satisfied only if $p_A > 1 - \exp(-\lambda\|\alpha\|_1)$ and $1 - p_B > 1 - \exp(-\lambda\|\alpha\|_1)$. In this case $\underline{S}_{\zeta^{-1}(p)} = S_0$ and $\overline{S}_{\zeta^{-1}(p)} = \mathbb{R}^m_{\geq 0}$. For $p \in [0, 1]$ let $\mathcal{S}_p = \{S \subseteq \mathbb{R}^m_{\geq 0} \colon S_0 \subseteq S \subseteq \mathbb{R}^m_{\geq 0}, \mathbb{P}_0(S) \leq p\}$. Then

$$p > 1 - \exp(-\lambda\|\alpha\|_1) \Rightarrow \xi(p) = \sup_{S \in \mathcal{S}_p} \mathbb{P}_1(S). \tag{120}$$

We can write any $S \in \mathcal{S}_p$ as the disjoint union $S = S_0 \dot\cup T$ for some $T \subseteq \mathbb{R}^m_{\geq 0}$ such that $\mathbb{P}_0(S_0 \dot\cup T) \leq p$. Note that $\mathbb{P}_1(S_0) = 0$ and since $S_0 \cap T = \varnothing$ any $z \in T$ satisfies $0 \leq \min_i(z_i - \alpha_i) \leq \min_i z_i$ and hence $\Lambda(z) = \exp(\lambda\|\alpha\|_1)$. Thus

$$\mathbb{P}_1(S) = \mathbb{P}_1(T) = \int_T f_1(z)\,dz \tag{121}$$

$$= \int_T \exp(\lambda\|\alpha\|_1) f_0(z)\,dz = \exp(\lambda\|\alpha\|_1) \cdot \mathbb{P}_0(T). \tag{122}$$

Thus, The supremum of the left hand side over all $S \in \mathcal{S}_p$ equals the supremum of the right hand side over all $T \in \{T' \subseteq S_0^c \colon \mathbb{P}_0(T') \leq 1 - \mathbb{P}_0(S_0)\}$

$$\begin{aligned} &\sup_{S \in \mathcal{S}_p} \mathbb{P}_1(S) = \\ &\exp(\lambda\|\alpha\|_1) \cdot \sup\{\mathbb{P}_1(T') \colon T' \subseteq S_0^c, \mathbb{P}_0(T') \leq p - \mathbb{P}_0(S_0)\} \end{aligned} \tag{123}$$

$$= \exp(\lambda\|\alpha\|_1) \cdot (p - \mathbb{P}_0(S_0)). \tag{124}$$

Computing ξ at p_A thus yields

$$\xi(p_A) = \sup_{S \in \mathcal{S}_{p_A}} \mathbb{P}_1(S) = \exp(\lambda\|\alpha\|_1) \cdot (p_A - \mathbb{P}_0(S_0)) \tag{125}$$

$$= \exp(\lambda\|\alpha\|_1) \cdot (p_A - (1 - \exp(-\lambda\|\alpha\|_1))) \tag{126}$$

$$= \exp(\lambda\|\alpha\|_1) \cdot (p_A + \exp(-\lambda\|\alpha\|_1) - 1) \tag{127}$$

where the third equality follows from (110). Similarly, computing ξ at $1 - p_B$ yields

$$\xi(1 - p_B) = \sup_{S \in \mathcal{S}_{1-p_B}} \mathbb{P}_1(S) \tag{128}$$

$$= \exp(\lambda\|\alpha\|_1) \cdot (1 - p_B - \mathbb{P}_0(S_0)) \tag{129}$$

$$= \exp(\lambda\|\alpha\|_1) \cdot (1 - p_B - (1 - \exp(-\lambda\|\alpha\|_1))) \tag{130}$$

$$= \exp(\lambda\|\alpha\|_1) \cdot (-p_B + \exp(-\lambda\|\alpha\|_1)). \tag{131}$$

Finally, condition (102) is satisfied whenever α satisfies

$$\begin{aligned} \exp(\lambda\|\alpha\|_1) &\cdot (p_A + \exp(-\lambda\|\alpha\|_1) - 1) \\ &+ \exp(\lambda\|\alpha\|_1) \cdot (-p_B + \exp(-\lambda\|\alpha\|_1)) > 1 \end{aligned} \tag{132}$$

$$\Longleftrightarrow$$

$$\begin{aligned} \exp(-\lambda\|\alpha\|_1) &+ p_B - \exp(-\lambda\|\alpha\|_1) \\ &< p_A + \exp(-\lambda\|\alpha\|_1) - 1 \end{aligned} \tag{133}$$

$$\Longleftrightarrow$$

$$1 - p_A + p_B < \exp(-\lambda\|\alpha\|_1) \tag{134}$$

$$\Longleftrightarrow$$

$$\|\alpha\|_1 < -\frac{\log(1 - p_A + p_B)}{\lambda} \tag{135}$$

what completes the proof. □

D.3 Uniform Smoothing

COROLLARY 9. *Suppose $\mathcal{Z} = \mathbb{R}^m$, and $\varepsilon_0 \sim \mathcal{U}([a, b]^m)$ for some $a < b$. Set $\varepsilon_1 := \alpha + \varepsilon_0$ for $\alpha \in \mathbb{R}^m$. Suppose that the ε_0-smoothed classifier g is (p_A, p_B)-confident at $x \in \mathcal{X}$ for some $y_A \in \mathcal{Y}$. Then, it holds that $q(y_A \mid x; \varepsilon_1) > \max_{y \neq y_A} q(y \mid x; \varepsilon_1)$ if α satisfies*

$$1 - \left(\frac{p_A - p_B}{2}\right) < \prod_{i=1}^m \left(1 - \frac{|\alpha_i|}{b - a}\right)_+ \tag{136}$$

where $(x)_+ := \max\{x, 0\}$.

PROOF. By Theorem 1 we know that if ε_1 satisfies

$$\xi(p_A) + \xi(1 - p_B) > 1, \tag{137}$$

then it is guaranteed that

$$q(y_A \mid x; \varepsilon_1) > \max_{y \neq y_A} q(y \mid x; \varepsilon_1). \tag{138}$$

The proof is thus complete if we show that (137) reduces to (136). For that purpose, denote by f_0 and f_1 density functions of ε_0 and ε_1, respectively, and let $I_0 = [a, b]^m$ and $I_1 := \prod_{i=1}^m [a + \alpha_i, b + \alpha_i]$ bet the support of ε_0 and ε_1. Consider

$$f_0(z) = \begin{cases} (b - a)^{-m} & z \in I_0, \\ 0 & \text{otherwise} \end{cases} \tag{139}$$

$$f_1(z) = \begin{cases} (b - a)^{-m} & z \in I_1, \\ 0 & \text{otherwise.} \end{cases} \tag{140}$$

Let $S_0 := I_0 \setminus I_1$. Then, for any $z \in I_0 \cup I_1$

$$\Lambda(z) = \frac{f_1(z)}{f_0(z)} = \begin{cases} 0 & z \in S_0, \\ 1 & z \in I_0 \cap I_1, \\ \infty & z \in I_1 \setminus I_0. \end{cases} \tag{141}$$

Note that

$$\mathbb{P}_0(S_0) = 1 - \mathbb{P}_0(I_1) \tag{142}$$

$$= 1 - \prod_{i=1}^{m} \mathbb{P}\left(a + \alpha_i \le \varepsilon_{0,i} \le b + \alpha_i\right) \tag{143}$$

$$= 1 - \prod_{i=1}^{m} \left(1 - \frac{|\alpha_i|}{b-a}\right)_+ \tag{144}$$

where $(x)_+ = \max\{x, 0\}$. We then compute ζ for $t \ge 0$

$$\zeta(t) = \mathbb{P}\left(\Lambda(\varepsilon_0) \le t\right) = \begin{cases} \mathbb{P}_0(S_0) & t < 1, \\ \mathbb{P}_0(I_0) & t \ge 1. \end{cases} \tag{145}$$

$$= \begin{cases} 1 - \prod_{i=1}^{m} \left(1 - \frac{|\alpha_i|}{b-a}\right)_+ & t < 1, \\ 1 & t \ge 1. \end{cases} \tag{146}$$

Recall that $\zeta^{-1}(p) := \inf\{t \ge 0 \colon \zeta(t) \ge p\}$ for $p \in [0, 1]$ and hence

$$\zeta^1(p) = \begin{cases} 0 & p \le 1 - \prod_{i=1}^{m} \left(1 - \frac{|\alpha_i|}{b-a}\right)_+, \\ 1 & p > 1 - \prod_{i=1}^{m} \left(1 - \frac{|\alpha_i|}{b-a}\right)_+. \end{cases} \tag{147}$$

In order to evaluate ξ, we compute the lower and strict lower level sets at $t = \zeta^{-1}(p)$. Recall that $\underline{S}_t = \{z \in \mathbb{R}^m_{\ge 0} \colon \Lambda(z) < t\}$ and $\overline{S}_t = \{z \in \mathbb{R}^m_{\ge 0} \colon \Lambda(z) \le t\}$ and consider

$$\underline{S}_{\zeta^{-1}(p)} = \begin{cases} \varnothing & p \le 1 - \prod_{i=1}^{m} \left(1 - \frac{|\alpha_i|}{b-a}\right)_+, \\ S_0 & p > 1 - \prod_{i=1}^{m} \left(1 - \frac{|\alpha_i|}{b-a}\right)_+ \end{cases} \tag{148}$$

and

$$\overline{S}_{\zeta^{-1}(p)} = \begin{cases} S_0 & p \le 1 - \prod_{i=1}^{m} \left(1 - \frac{|\alpha_i|}{b-a}\right)_+, \\ I_0 & p > 1 - \prod_{i=1}^{m} \left(1 - \frac{|\alpha_i|}{b-a}\right)_+ \end{cases} \tag{149}$$

Suppose $p \le 1 - \prod_{i=1}^{m} \left(1 - \frac{|\alpha_i|}{b-a}\right)_+$. Then $\underline{S}_{\zeta^{-1}(p)} = \varnothing$ and $\overline{S}_{\zeta^{-1}(p)} = S_0$ and hence

$$p \le 1 - \prod_{i=1}^{m} \left(1 - \frac{|\alpha_i|}{b-a}\right)_+ \tag{150}$$
$$\Rightarrow \xi(p) = \sup\{\mathbb{P}_1(S) \colon S \subseteq S_0, \mathbb{P}_0(S) \le p\} = 0.$$

Condition (137) can thus be satisfied only if $p_A > 1 - \prod_{i=1}^{m} \left(1 - \frac{|\alpha_i|}{b-a}\right)_+$ and $1 - p_B > 1 - \prod_{i=1}^{m} \left(1 - \frac{|\alpha_i|}{b-a}\right)_+$. In this case $\underline{S}_{\zeta^{-1}(p)} = S_0$ and $\overline{S}_{\zeta^{-1}(p)} = I_0$. For $p \in [0, 1]$ let $\mathcal{S}_p = \{S \subseteq \mathbb{R}^m \colon S_0 \subseteq S \subseteq I_0, \mathbb{P}_0(S) \le p\}$. Then

$$p > 1 - \prod_{i=1}^{m} \left(1 - \frac{|\alpha_i|}{b-a}\right)_+ \Rightarrow \xi(p) = \sup_{S \in \mathcal{S}_p} \mathbb{P}_1(S). \tag{151}$$

We can write any $S \in \mathcal{S}_p$ as the disjoint union $S = S_0 \,\dot\cup\, T$ for some $T \subseteq I_0 \cap I_1$ such that $\mathbb{P}_0(S_0 \,\dot\cup\, T) \le p$. Note that $\mathbb{P}_1(S_0) = 0$ and for

any $z \in T$, we have $f_0(z) = f_1(z)$. Hence

$$\mathbb{P}_1(S) = \mathbb{P}_1(T) = \mathbb{P}_0(T) \tag{152}$$

$$\le p - \mathbb{P}_0(S_0) = p - \left(1 - \prod_{i=1}^{m} \left(1 - \frac{|\alpha_i|}{b-a}\right)_+\right). \tag{153}$$

Thus, The supremum of the left hand side over all $S \in \mathcal{S}_p$ equals the supremum of the right hand side over all $T \in \{T' \subseteq I_0 \cap I_1 \colon \mathbb{P}_0(T') \le 1 - \mathbb{P}_0(S_0)\}$

$$\sup_{S \in \mathcal{S}_p} \mathbb{P}_1(S) = \sup\{\mathbb{P}_1(T') \colon T' \subseteq I_0 \cap I_1, \\ \mathbb{P}_0(T') \le p - \mathbb{P}_0(S_0)\} \tag{154}$$

$$= p - \left(1 - \prod_{i=1}^{m} \left(1 - \frac{|\alpha_i|}{b-a}\right)_+\right). \tag{155}$$

Hence, computing ξ at p_A and $1 - p_B$ yields

$$\xi(p_A) = p_A - \left(1 - \prod_{i=1}^{m} \left(1 - \frac{|\alpha_i|}{b-a}\right)_+\right), \tag{156}$$

$$\xi(1 - p_B) = 1 - p_B - \left(1 - \prod_{i=1}^{m} \left(1 - \frac{|\alpha_i|}{b-a}\right)_+\right). \tag{157}$$

Finally, condition (137) is satisfied whenever α satisfies

$$1 - \left(1 - p_B - \left(1 - \prod_{i=1}^{m} \left(1 - \frac{|\alpha_i|}{b-a}\right)_+\right)\right) \\ < p_A - \left(1 - \prod_{i=1}^{m} \left(1 - \frac{|\alpha_i|}{b-a}\right)_+\right) \tag{158}$$

$$\Longleftrightarrow$$

$$p_B + 1 - \prod_{i=1}^{m} \left(1 - \frac{|\alpha_i|}{b-a}\right)_+ < p_A - 1 + \prod_{i=1}^{m} \left(1 - \frac{|\alpha_i|}{b-a}\right)_+ \tag{159}$$

$$\Longleftrightarrow 2 - p_A + p_B < 2 \cdot \prod_{i=1}^{m} \left(1 - \frac{|\alpha_i|}{b-a}\right)_+ \tag{160}$$

$$\Longleftrightarrow 1 - \left(\frac{p_A - p_B}{2}\right) < \prod_{i=1}^{m} \left(1 - \frac{|\alpha_i|}{b-a}\right)_+ \tag{161}$$

what concludes the proof. $\qquad\square$

D.4 Laplacian Smoothing

COROLLARY 10. *Suppose $\mathcal{Z} = \mathbb{R}$ and $\varepsilon_0 \sim \mathcal{L}(0, b)$ follows a Laplace distribution with mean 0 and scale parameter $b > 0$. Let $\varepsilon_1 := \alpha + \varepsilon_0$ for $\alpha \in \mathbb{R}$. Suppose that the ε_0-smoothed classifier g is (p_A, p_B)-confident at $x \in \mathcal{X}$ for some $y_A \in \mathcal{Y}$. Then, it holds that $q(y_A \mid x; \varepsilon_1) > \max_{y \neq y_A} q(y \mid x; \varepsilon_1)$ if α satisfies*

$$|\alpha| < \begin{cases} -b \cdot \log\left(4\,p_B(1 - p_A)\right) & (p_A = \frac{1}{2} \wedge p_B < \frac{1}{2}) \\ & \vee (p_A > \frac{1}{2} \wedge p_B = \frac{1}{2}), \\ -b \cdot \log\left(1 - p_A + p_B\right) & p_A > \frac{1}{2} \wedge p_B < \frac{1}{2}. \end{cases} \tag{162}$$

PROOF. By Theorem 1 we know that if ε_1 satisfies

$$\xi(p_A) + \xi(1 - p_B) > 1, \tag{163}$$

then it is guaranteed that

$$q(y_A \mid x; \varepsilon_1) > \max_{y \neq y_A} q(y \mid x; \varepsilon_1). \tag{164}$$

The proof is thus complete if we show that (163) reduces to (162). For that purpose denote by f_0 and f_1 density functions of ε_0 and ε_1, respectively, and consider

$$f_0(z) = \frac{1}{2b} \exp\left(-\frac{|z|}{b}\right), \quad f_1(z) = \frac{1}{2b} \exp\left(-\frac{|z-\alpha|}{b}\right). \quad (165)$$

Due to symmetry, assume without loss of generality that $\alpha \geq 0$. Then for $z \in \mathbb{R}$

$$\Lambda(z) = \frac{f_1(z)}{f_0(z)} = \exp\left(-\frac{|z-\alpha|-|z|}{b}\right) \quad (166)$$

$$= \begin{cases} \exp\left(-\frac{\alpha}{b}\right) & z < 0, \\ \exp\left(\frac{2z-\alpha}{b}\right) & 0 \leq z < \alpha, \\ \exp\left(\frac{\alpha}{b}\right) & z \geq \alpha. \end{cases} \quad (167)$$

Note that the CDFs for ε_0 and ε_1 are given by

$$F_0(z) = \begin{cases} \frac{1}{2} \exp\left(\frac{z}{b}\right) & z \leq 0, \\ 1 - \frac{1}{2} \exp\left(-\frac{z}{b}\right) & z > 0, \end{cases} \quad (168)$$

$$F_1(z) = \begin{cases} \frac{1}{2} \exp\left(\frac{z-\alpha}{b}\right) & z \leq \alpha, \\ 1 - \frac{1}{2} \exp\left(-\frac{z-\alpha}{b}\right) & z > \alpha. \end{cases} \quad (169)$$

Note that for $\exp\left(-\frac{\alpha}{b}\right) \leq t < \exp\left(\frac{\alpha}{b}\right)$ we have

$$\mathbb{P}_0\left(\exp\left(\frac{2\varepsilon_0 - \alpha}{b}\right) \leq t \wedge 0 \leq \varepsilon_0 < \alpha\right) \quad (170)$$

$$= \mathbb{P}_0\left(\exp\left(-\frac{\alpha}{b}\right) \leq \exp\left(\frac{2\varepsilon_0 - \alpha}{b}\right) \leq t\right)$$

$$= \mathbb{P}_0\left(0 \leq \varepsilon_0 \leq \frac{b\log(t) + \alpha}{2}\right) \quad (171)$$

$$= F_0\left(\frac{b\log(t) + \alpha}{2}\right) - F_0(0) \quad (172)$$

$$= \frac{1}{2} - \frac{1}{2} \exp\left(-\frac{1}{b}\left(\frac{b\log(t) + \alpha}{2}\right)\right) \quad (173)$$

$$= \frac{1}{2} - \frac{1}{2\sqrt{t}} \exp\left(-\frac{\alpha}{2b}\right). \quad (174)$$

Computing ζ yields

$$\zeta(t) = \mathbb{P}(\Lambda(\varepsilon_0) \leq t) \quad (175)$$

$$= \mathbb{P}\left(\exp\left(-\frac{\alpha}{b}\right) \leq t \wedge \varepsilon_0 < 0\right)$$

$$+ \mathbb{P}\left(\exp\left(\frac{\alpha}{b}\right) \leq t \wedge \varepsilon_0 \geq \alpha\right) \quad (176)$$

$$+ \mathbb{P}\left(\exp\left(\frac{2\varepsilon_0 - \alpha}{b}\right) \leq t \wedge 0 \leq \varepsilon_0 < \alpha\right)$$

$$= \begin{cases} 0 & t < \exp\left(-\frac{\alpha}{b}\right), \\ 1 - \frac{1}{2\sqrt{t}} \exp\left(-\frac{\alpha}{2b}\right) & \exp\left(-\frac{\alpha}{b}\right) \leq t < \exp\left(\frac{\alpha}{b}\right), \\ 1 & t \geq \exp\left(\frac{\alpha}{b}\right). \end{cases} \quad (177)$$

The inverse is then given by

$$\zeta^{-1}(p) = \begin{cases} 0 & p < \frac{1}{2}, \\ \frac{1}{4(1-p)^2} \exp\left(-\frac{\alpha}{b}\right) & \frac{1}{2} \leq p < 1 - \frac{1}{2} \exp(-\frac{\alpha}{b}), \\ \exp\left(\frac{\alpha}{b}\right) & p \geq 1 - \frac{1}{2} \exp(-\frac{\alpha}{b}). \end{cases} \quad (178)$$

In order to evaluate ξ, we compute the lower and strict lower level sets at $t = \zeta^{-1}(p)$. Recall that $\underline{S}_t = \{z \in \mathbb{R}: \Lambda(z) < t\}$ and $\overline{S}_t = \{z \in \mathbb{R}: \Lambda(z) \leq t\}$ and consider

$$\underline{S}_{\zeta^{-1}(p)} = \begin{cases} \varnothing & p \leq \frac{1}{2}, \\ \left(-\infty, b \cdot \log\left(\frac{1}{2(1-p)}\right)\right) & \frac{1}{2} < p < 1 - \frac{1}{2} \exp\left(-\frac{\alpha}{b}\right), \\ (-\infty, \alpha], & p \geq 1 - \frac{1}{2} \exp\left(-\frac{\alpha}{b}\right) \end{cases} \quad (179)$$

and

$$\overline{S}_{\zeta^{-1}(p)} = \begin{cases} \varnothing & p < \frac{1}{2}, \\ \left(-\infty, b \cdot \log\left(\frac{1}{2(1-p)}\right)\right] & \frac{1}{2} \leq p < 1 - \frac{1}{2} \exp\left(-\frac{\alpha}{b}\right), \\ \mathbb{R} & p \geq 1 - \frac{1}{2} \exp\left(-\frac{\alpha}{b}\right). \end{cases} \quad (180)$$

Suppose $p < \frac{1}{2}$. Then $\underline{S}_{\zeta^{-1}(p)} = \overline{S}_{\zeta^{-1}(p)} = \varnothing$ and hence $\xi(p) = 0$ and condition (163) cannot be satisfied. If $p = \frac{1}{2}$, then $\underline{S}_{\zeta^{-1}(p)} = \varnothing$ and $\overline{S}_{\zeta^{-1}(p)} = (-\infty, 0]$. Note that for $z \leq 0$ we have $f_1(z) = f_0(z) \exp(-\alpha/b)$ and hence for any $S \subseteq \overline{S}_{\zeta^{-1}(1/2)}$ we have $\mathbb{P}_1(S) = \exp(-\alpha/b) \cdot \mathbb{P}_0(S)$. We can thus compute ξ at $1/2$ as

$$p = \frac{1}{2}$$

$$\Rightarrow \xi(1/2) = \sup\left\{\mathbb{P}_1(S): S \subseteq (-\infty, 0], \mathbb{P}_0(S) \leq \frac{1}{2}\right\} = \frac{1}{2}. \quad (181)$$

Now suppose $1/2 < p < 1 - 1/2 \exp(-\alpha/b)$. In this case, $\underline{S}_{\zeta^{-1}(p)} = (-\infty, b \cdot \log(1/2(1-p)))$ and $\overline{S}_{\zeta^{-1}(p)} = (-\infty, b \cdot \log(1/2(1-p))]$. Since the singleton $\{b \cdot \log(1/2(1-p))\}$ has no probability mass under both $\mathbb{P}_0$ and $\mathbb{P}_1$, the function ξ is straight forward to compute: if $\frac{1}{2} < p < 1 - \frac{1}{2} \exp(-\frac{\alpha}{b})$, then

$$\xi(p) = \mathbb{P}\left(\varepsilon_1 \leq b \cdot \log\left(\frac{1}{2(1-p)}\right)\right) \quad (182)$$

$$= \frac{1}{2} \exp\left(\frac{b \cdot \log\left(\frac{1}{2(1-p)}\right) - \alpha}{b}\right) \quad (183)$$

$$= \frac{1}{4(1-p)} \exp\left(-\frac{\alpha}{b}\right). \quad (184)$$

Finally, consider the case where $p \geq 1 - 1/2 \exp(-\alpha/b)$. Then $\underline{S}_{\zeta^{-1}(p)} = (-\infty, \alpha]$ and $\overline{S}_{\zeta^{-1}(p)} = \mathbb{R}$. Any $(-\infty, \alpha] \subseteq S \subseteq \mathbb{R}$ can then be written as $S = (-\infty, \alpha] \cup T$ for some $T \subseteq (\alpha, \infty)$. Hence

$$\mathbb{P}_1(S) = \mathbb{P}(\varepsilon_1 \leq \alpha) + \mathbb{P}_1(T) = \frac{1}{2} + \exp\left(\frac{\alpha}{b}\right) \mathbb{P}_0(T), \quad (185)$$

$$\mathbb{P}_0(S) = \mathbb{P}(\varepsilon_0 \leq \alpha) + \mathbb{P}_0(T) = 1 - \frac{1}{2} \exp(-\frac{\alpha}{b}) + \mathbb{P}_0(T). \quad (186)$$

Thus, if $p \geq 1 - \frac{1}{2} \exp(-\frac{\alpha}{b})$, then

$$\xi(p) = \sup\{\mathbb{P}_1(S): (-\infty, \alpha] \subseteq S \subseteq \mathbb{R}, \mathbb{P}_0(S) \leq p\} \quad (187)$$

$$= \frac{1}{2} + \sup\Big\{\mathbb{P}_1(T): T \subseteq (\alpha, \infty),$$

$$\mathbb{P}_0(T) \leq p - 1 + \frac{1}{2} \exp\left(-\frac{\alpha}{b}\right)\Big\} \quad (188)$$

$$= \frac{1}{2} + \exp\left(\frac{\alpha}{b}\right)\left(p - 1 + \frac{1}{2}\exp\left(-\frac{\alpha}{b}\right)\right) \tag{189}$$

$$= 1 - \exp\left(\frac{\alpha}{b}\right)(1 - p). \tag{190}$$

In order to evaluate condition (163), consider

$$1 - \xi(1 - p_B) = \begin{cases} 1 & p_B > \frac{1}{2} \\ \frac{1}{2} & p_B = \frac{1}{2} \\ 1 - \frac{1}{4p_B}\exp\left(-\frac{\alpha}{b}\right) & \frac{1}{2} > p_B > \exp\left(-\frac{\alpha}{b}\right) \\ \exp\left(\frac{\alpha}{b}\right)p_B & \exp\left(-\frac{\alpha}{b}\right) \geq p_B, \end{cases} \tag{191}$$

$$\xi(p_A) = \begin{cases} 0 & p_A < \frac{1}{2} \\ \frac{1}{2} & p_A = \frac{1}{2} \\ \frac{1}{4(1-p_A)}\exp\left(-\frac{\alpha}{b}\right) & \frac{1}{2} < p_A < 1 - \frac{1}{2}\exp(-\frac{\alpha}{b}) \\ 1 - \exp\left(\frac{\alpha}{b}\right)(1 - p_A) & p_A \geq 1 - \frac{1}{2}\exp(-\frac{\alpha}{b}). \end{cases} \tag{192}$$

Note that the case $p_B > 1/2$ can be ruled out, since by assumption $p_A \geq p_B$. If $p_A = 1/2$, then we need $p_B < 1/2$. Thus, if $p_A = 1/2$, then condition (163) is satisfied if $p_B < 1/2$ and

$$\max\left\{1 - \frac{1}{4p_B}\exp\left(-\frac{\alpha}{b}\right), \exp\left(\frac{\alpha}{b}\right)\cdot p_B\right\} < \frac{1}{2} \tag{193}$$

$$\iff p_B \cdot \exp\left(\frac{\alpha}{b}\right) < \frac{1}{2} \tag{194}$$

$$\iff \alpha < -b \cdot \log(2p_B). \tag{195}$$

Now consider the case where $p_A > 1/2$. If $p_B = 1/2$, then condition (163) is satisfied if

$$\frac{1}{2} < \min\left\{\frac{1}{4(1-p_A)}\exp\left(-\frac{\alpha}{b}\right), 1 - \exp\left(\frac{\alpha}{b}\right)(1-p_A)\right\} \tag{196}$$

$$\iff \frac{1}{2} < 1 - \exp\left(\frac{\alpha}{b}\right)(1 - p_A) \tag{197}$$

$$\iff \alpha < -b \cdot \log(2(1 - p_A)). \tag{198}$$

If on the other hand, $p_A > 1/2$ and $p_B < 1/2$, condition (163) is satisfied if

$$\max\left\{1 - \frac{1}{4p_B}\exp\left(-\frac{\alpha}{b}\right), \exp\left(\frac{\alpha}{b}\right)\cdot p_B\right\} < \\ \min\left\{\frac{1}{4(1-p_A)}\exp\left(-\frac{\alpha}{b}\right), 1 - \exp\left(\frac{\alpha}{b}\right)(1-p_A)\right\} \tag{199}$$

$$\iff$$

$$p_B \cdot \exp\left(\frac{\alpha}{b}\right) < 1 - \exp\left(\frac{\alpha}{b}\right)(1 - p_A) \tag{200}$$

$$\iff$$

$$\alpha < -b \cdot \log(1 - p_A + p_B). \tag{201}$$

Finally, we get that condition (163) is satisfied, if

$$|\alpha| < \begin{cases} -b \cdot \log(4 p_B(1 - p_A)) & (p_A = \frac{1}{2} \land p_B < \frac{1}{2}) \\ & \lor (p_A > \frac{1}{2} \land p_B = \frac{1}{2}) \\ -b \cdot \log(1 - p_A + p_B) & p_A > \frac{1}{2} \land p_B < \frac{1}{2} \end{cases} \tag{202}$$

what concludes the proof. □

D.5 Folded Gaussian Smoothing

COROLLARY 11. *Suppose* $\mathcal{Z} = \mathbb{R}_{\geq 0}$, $\varepsilon_0 \sim |\mathcal{N}(0, \sigma)|$ *and* $\varepsilon_1 := \alpha + \varepsilon_0$ *for some* $\alpha > 0$. *Suppose that the* ε_0*-smoothed classifier* g *is* (p_A, p_B)*-confident at* $x \in \mathcal{X}$ *for some* $y_A \in \mathcal{Y}$. *Then, it holds that* $q(y_A|x; \varepsilon_1) > \max_{y \neq y_A} q(y|x; \varepsilon_1)$ *if* α *satisfies*

$$\alpha < \sigma \cdot \left(\Phi^{-1}\left(\frac{1 + \min\{p_A, 1 - p_B\}}{2}\right) - \Phi^{-1}\left(\frac{3}{4}\right)\right). \tag{203}$$

PROOF. By Theorem 1 we know that if ε_1 satisfies

$$\xi(p_A) + \xi(1 - p_B) > 1, \tag{204}$$

then it is guaranteed that

$$q(y_A|x; \varepsilon_1) > \max_{y \neq y_A} q(y|x; \varepsilon_1). \tag{205}$$

The proof is thus complete if we show that (204) reduces to (203). For that purpose denote by f_0 and f_1 density functions of ε_0 and ε_1, respectively, and consider

$$f_0(z) = \begin{cases} \frac{2}{\sqrt{2\pi}\sigma}\exp\left(-\frac{z^2}{2\sigma^2}\right) & z \geq 0 \\ 0 & z < 0 \end{cases} \tag{206}$$

$$f_1(z) = \begin{cases} \frac{2}{\sqrt{2\pi}\sigma}\exp\left(-\frac{(z-\alpha)^2}{2\sigma^2}\right) & z \geq \alpha \\ 0 & z < \alpha. \end{cases} \tag{207}$$

Then, for $z \geq 0$,

$$\Lambda(z) = \frac{f_1(z)}{f_0(z)} = \begin{cases} 0 & z < \alpha, \\ \exp\left(\frac{z\alpha}{\sigma^2} - \frac{\alpha^2}{2\sigma^2}\right) & z \geq \alpha. \end{cases} \tag{208}$$

Let $t_\alpha := \exp\left(\frac{\alpha^2}{2\sigma^2}\right)$ and suppose $t < t_\alpha$. Then

$$\zeta(t) = \mathbb{P}(\Lambda(\varepsilon_0) \leq t] = \mathbb{P}(\varepsilon_0 < \alpha) \tag{209}$$

$$= \int_0^\alpha \frac{2}{\sqrt{2\pi}\sigma}\exp\left(-\frac{z^2}{2\sigma^2}\right)dz \tag{210}$$

$$= 2 \cdot \int_0^{\alpha/\sigma} \frac{1}{\sqrt{2\pi}}\exp\left(-\frac{s^2}{2}\right)ds = 2 \cdot \Phi\left(\frac{\alpha}{\sigma}\right) - 1. \tag{211}$$

If $t \geq t_\alpha$, then

$$\zeta(t) = \mathbb{P}(\Lambda(\varepsilon_0) \leq t) \tag{212}$$

$$= \mathbb{P}\left(\frac{\varepsilon_0 \alpha}{\sigma^2} - \frac{\alpha^2}{2\sigma^2} \leq \log(t) \land \varepsilon_0 \geq \alpha\right) + \mathbb{P}(\varepsilon_0 < \alpha) \tag{213}$$

$$= \mathbb{P}\left(\varepsilon_0 \leq \frac{\sigma^2}{\alpha}\log(t) + \frac{1}{2}\alpha\right) \tag{214}$$

$$= 2 \cdot \Phi\left(\frac{\sigma}{\alpha}\log(t) + \frac{\alpha}{2\sigma}\right) - 1 \tag{215}$$

and hence

$$\zeta(t) = \begin{cases} 2 \cdot \Phi\left(\frac{\alpha}{\sigma}\right) - 1 & t < t_\alpha \\ 2 \cdot \Phi\left(\frac{\sigma}{\alpha}\log(t) + \frac{\alpha}{2\sigma}\right) - 1 & t \geq t_\alpha. \end{cases} \tag{216}$$

Note that $\zeta(t_\alpha) = 2 \cdot \Phi\left(\frac{\alpha}{\sigma}\right) - 1$ and let $p_\alpha := \zeta(t_\alpha)$. Recall that $\zeta^{-1}(p) := \inf\{t \geq 0 : \zeta(t) \geq p\}$, which yields

$$\zeta^{-1}(p) = \begin{cases} 0 & p \leq p_\alpha \\ \exp\left(\frac{\alpha}{\sigma}\Phi^{-1}\left(\frac{1+p}{2}\right) - \frac{\alpha^2}{2\sigma^2}\right) & p > p_\alpha. \end{cases} \tag{217}$$

In order to evaluate ξ we compute the lower and strict lower level sets at $t = \zeta^{-1}(p)$. Recall that $\underline{S}_t = \{z \in \mathbb{R}_{\geq 0} : \Lambda(z) < t\}$ and

Table 5: Comparison of certification radii with $p_A + p_B = 1$, the variance and noise dimension are set to 1 for each distribution.

Distribution	Value Space	Robust Radius
Gaussian$(0, 1)$	$(-\infty, \infty)$	$\Phi^{-1}(p_A)$
Laplace$(0, 1/\sqrt{2})$	$(-\infty, \infty)$	$-\log(2 - 2p_A)/\sqrt{2}$
Uniform$[-\sqrt{3}, -\sqrt{3}]$	$(-\infty, \infty)$	$2\sqrt{3} \cdot (p_A - 1/2)$
Exponential(1)	$[0, \infty)$	$-\log(2 - 2p_A)$
FoldedGaussian$(0, \sqrt{\frac{\pi}{\pi-2}})$	$[0, \infty)$	$\sqrt{\frac{\pi}{\pi-2}} \cdot \left(\Phi^{-1}\left(\frac{1+p_A}{2}\right) - \Phi^{-1}\left(\frac{3}{4}\right)\right)$

$\overline{S}_t = \{z \in \mathbb{R}_{\geq 0} : \Lambda(z) \leq t\}$. Let $S_0 := [0, \alpha)$ and note that if $p \leq p_\alpha$, we have $\zeta^{-1}(p) = 0$ and hence $\underline{S}_{\zeta^{-1}(p)} = \varnothing$ and $\overline{S}_{\zeta^{-1}(p)} = S_0$. If, on the other hand $p > p_\alpha$, then

$$\underline{S}_{\zeta^{-1}(p)} = \left\{z \geq 0 : \Lambda(z) < \zeta^{-1}(p)\right\} \tag{218}$$

$$= S_0 \cup \left\{z \geq \alpha : \frac{z\,\alpha}{\sigma^2} - \frac{\alpha^2}{2\sigma^2} < \frac{\alpha}{\sigma}\Phi^{-1}\left(\frac{1+p}{2}\right) - \frac{\alpha^2}{2\sigma^2}\right\} \tag{219}$$

$$= S_0 \cup \left\{z \geq \alpha : z < \sigma \cdot \Phi^{-1}\left(\frac{1+p}{2}\right)\right\} \tag{220}$$

$$= S_0 \cup \left[\alpha, \sigma \cdot \Phi^{-1}\left(\frac{1+p}{2}\right)\right) \tag{221}$$

and

$$\overline{S}_{\zeta^{-1}(p)} = \left\{z \geq 0 : \Lambda(z) \leq \zeta^{-1}(p)\right\} \tag{222}$$

$$= S_0 \cup \left\{z \geq \alpha : \frac{z\,\alpha}{\sigma^2} - \frac{\alpha^2}{2\sigma^2} \leq \frac{\alpha}{\sigma}\Phi^{-1}\left(\frac{1+p}{2}\right) - \frac{\alpha^2}{2\sigma^2}\right\} \tag{223}$$

$$= S_0 \cup \left\{z \geq \alpha : z \leq \sigma \cdot \Phi^{-1}\left(\frac{1+p}{2}\right)\right\} \tag{224}$$

$$= S_0 \cup \left[\alpha, \sigma \cdot \Phi^{-1}\left(\frac{1+p}{2}\right)\right] \tag{225}$$

$$= \underline{S}_{\zeta^{-1}(p)} \cup \left\{\sigma \cdot \Phi^{-1}\left(\frac{1+p}{2}\right)\right\}. \tag{226}$$

In other words

$$\underline{S}_{\zeta^{-1}(p)} = \begin{cases} \varnothing & p \leq p_\alpha, \\ S_0 \cup \left[\alpha, \sigma \cdot \Phi^{-1}\left(\frac{1+p}{2}\right)\right) & p > p_\alpha, \end{cases} \tag{227}$$

$$\overline{S}_{\zeta^{-1}(p)} = \begin{cases} S_0 & p \leq p_\alpha, \\ S_0 \cup \left[\alpha, \sigma \cdot \Phi^{-1}\left(\frac{1+p}{2}\right)\right] & p > p_\alpha. \end{cases} \tag{228}$$

Let $\mathcal{S}_p := \{S \subseteq \mathbb{R}_{\geq 0} : \underline{S}_{\zeta^{-1}(p)} \subseteq S \subseteq \overline{S}_{\zeta^{-1}(p)}, \mathbb{P}_0(S) \leq p\}$ and recall that $\xi(p) = \sup_{S \in \mathcal{S}_p} \mathbb{P}_1(S)$. Note that for $p \leq p_\alpha$, we have $\mathcal{S}_p = \{S \subseteq \mathbb{R}_{\geq 0} : S \subseteq S_0 \wedge \mathbb{P}_0(S) \leq p\}$ and for $S \subseteq S_0$, it holds that $\mathbb{P}_1(S) = 0$. Hence

$$p \leq p_\alpha \Rightarrow \xi(p) = \sup_{S \in \mathcal{S}_p} \mathbb{P}_1(S) = 0. \tag{229}$$

If $p > p_\alpha$, then

$$\mathcal{S}_p = \left\{S \subseteq \mathbb{R}_{\geq 0} : S_0 \cup \left[\alpha, \sigma \cdot \Phi^{-1}\left(1{+}p/2\right)\right) \subseteq S \right.$$
$$\left. \subseteq S_0 \cup \left[\alpha, \sigma \cdot \Phi^{-1}\left(1{+}p/2\right)\right], \wedge \mathbb{P}_0(S) \leq p\right\}. \tag{230}$$

Since the singleton $\left\{\sigma \cdot \Phi^{-1}\left(\frac{1+p}{2}\right)\right\}$ has no mass under both $\mathbb{P}_0$ and $\mathbb{P}_1$, we find that if $p > p_\alpha$, then

$$\xi(p) = \mathbb{P}\left(0 \leq \varepsilon_1 \leq \sigma \cdot \Phi^{-1}\left(\frac{1+p}{2}\right)\right) \tag{231}$$

$$= \mathbb{P}\left(0 \leq \varepsilon_0 \leq \sigma \cdot \Phi^{-1}\left(\frac{1+p}{2}\right) - \alpha\right) \tag{232}$$

$$= 2 \cdot \Phi\left(\Phi^{-1}\left(\frac{1+p}{2}\right) - \frac{\alpha}{\sigma}\right) - 1. \tag{233}$$

Condition (204) can thus be satisfied only if $p_B < p_A$ and

$$2 \cdot \Phi\left(\frac{\alpha}{\sigma}\right) - 1 < \min\{p_A, 1 - p_B\}$$
$$\wedge \xi(p_A) + \xi(1 - p_b) > 1 \tag{234}$$

that is equivalent to

$$\alpha < \sigma \cdot \Phi^{-1}\left(\frac{1 + \min\{p_A, 1 - p_B\}}{2}\right)$$
$$\wedge \Phi\left(\Phi^{-1}\left(\frac{1 + (1 - p_B)}{2}\right) - \frac{\alpha}{\sigma}\right) \tag{235}$$
$$+ \Phi\left(\Phi^{-1}\left(\frac{1 + p_A}{2}\right) - \frac{\alpha}{\sigma}\right) > \frac{3}{2}.$$

Thus, the following is a sufficient condition for the two inequalities in (235) and hence (204) to hold

$$\alpha < \sigma \cdot \left(\Phi^{-1}\left(\frac{1 + \min\{p_A, 1 - p_B\}}{2}\right) - \Phi^{-1}\left(\frac{3}{4}\right)\right) \tag{236}$$

what completes the proof. □

E COMPARISON OF DIFFERENT SMOOTHING DISTRIBUTIONS

Here, we provide robustness radii derived from different smoothing distributions in Table 5, each scaled to have unit variance. The figure comparison is in Figure 4 in main text.

F PROOFS FOR CERTIFYING RESOLVABLE TRANSFORMATIONS

Here, we state the proofs and technical details concerning our results for resolvable transformations. Recall the definition of resolvable transformations from the main part of this paper.

Definition 2 (restated). *A transformation $\phi \colon \mathcal{X} \times \mathcal{Z} \to \mathcal{X}$ is called resolvable if for any $\alpha \in \mathcal{Z}$ there exists a resolving function $\gamma_\alpha \colon \mathcal{Z} \to \mathcal{Z}$ that is injective, continuously differentiable, has non-vanishing Jacobian and for which*

$$\phi(\phi(x, \alpha), \beta) = \phi(x, \gamma_\alpha(\beta)) \quad x \in \mathcal{X}, \beta \in \mathcal{Z}. \tag{237}$$

We say that ϕ is additive, if $\gamma_\alpha(\beta) = \alpha + \beta$.

F.1 Proof of Corollary 1

Corollary 1 (restated). *Suppose that the transformation ϕ in Theorem 1 is resolvable with resolving function γ_α. Let $\alpha \in \mathcal{Z}$ and set $\varepsilon_1 := \gamma_\alpha(\varepsilon_0)$ in the definition of the functions ζ and ξ. Then, if α satisfies condition (6), it is guaranteed that $g(\phi(x, \alpha); \varepsilon_0) = g(x; \varepsilon_0)$.*

PROOF. Since ϕ is a resolvable transformation, by definition γ_α is injective, continuously differentiable and has non-vanishing Jacobian. By Jacobi's transformation formula (see e.g., [26]), it follows that the density of ε_1 vanishes outside the image of γ_α and is elsewhere given by

$$f_1(z) = f_0(\gamma_\alpha^{-1}(z))|\det(J_{\gamma_\alpha^{-1}(z)})| \quad \text{for any } z \in \text{Im}(\gamma_\alpha) \quad (238)$$

where $J_{\gamma_\alpha^{-1}(z)}$ is the Jacobian of $\gamma_\alpha^{-1}(z)$. Since f_1 is paramterized by α, it follows from Theorem 1 that if α satisfies (6) it is guaranteed that $\arg\max_y q(y|x, \varepsilon_1) = \arg\max_y q(y|x, \varepsilon_0)$. The statement of the corollary then follows immediately from the observation that for any $y \in \mathcal{Y}$ we have

$$q(y|x; \varepsilon_1) = \mathbb{E}(p(y|\phi(x, \varepsilon_1))) \quad (239)$$
$$= \mathbb{E}(p(y|\phi(x, \gamma_\alpha(\varepsilon_0)))) \quad (240)$$
$$= \mathbb{E}(p(y|\phi(\phi(x, \alpha), \varepsilon_0))) \quad (241)$$
$$= q(y|\phi(x, \alpha); \varepsilon_0). \quad (242)$$

$\square$

F.2 Gaussian Blur

Recall that the Gaussian blur transformation is given by a convolution with a Gaussian kernel

$$G_\alpha(k) = \frac{1}{\sqrt{2\pi\alpha}} \exp\left(-\frac{k^2}{2\alpha}\right) \quad (243)$$

where $\alpha > 0$ is the squared kernel radius. Here we show that the transformation $x \mapsto \phi_B(x) := x * G$ is additive.

Lemma 1 (restated). *The Gaussian blur transformation is additive, i.e., for any $\alpha, \beta \geq 0$, we have $\phi_B(\phi_B(x, \alpha), \beta) = \phi_B(x, \alpha + \beta)$.*

PROOF. Note that associativity of the convolution operator implies that

$$\phi_B(\phi_B(x, \alpha), \beta) = (\phi_B(x, \alpha) * G_\beta) \quad (244)$$
$$= ((x * G_\alpha) * G_\beta) \quad (245)$$
$$= (x * (G_\alpha * G_\beta)). \quad (246)$$

The claim thus follows, if we can show that $(G_\alpha * G_\beta) = G_{\alpha+\beta}$. Let $\mathcal{F}$ denote the Fourier transformation and $\mathcal{F}^{-1}$ the inverse Fourier transformation and note that by the convolution theorem $(G_\alpha * G_\beta) = \mathcal{F}^{-1}\{\mathcal{F}(G_\alpha) \cdot \mathcal{F}(G_\beta)\}$. Therefore we have to show that $\mathcal{F}(G_\alpha) \cdot \mathcal{F}(G_\beta) = \mathcal{F}(G_{\alpha+\beta})$. For that purpose, consider

$$\mathcal{F}(G_\alpha)(\omega) = \int_{-\infty}^{\infty} G_\alpha(y) \exp(-2\pi i\omega y) \, dy \quad (247)$$

$$= \int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi\alpha}} \exp\left(-\frac{y^2}{2\alpha}\right) \exp(-2\pi i\omega y) \, dy \quad (248)$$

$$= \frac{1}{\sqrt{2\pi\alpha}} \int_{-\infty}^{\infty} \exp\left(-\frac{y^2}{2\alpha}\right) (\cos(2\pi\omega y) + i\sin(2\pi\omega y)) \, dy \quad (249)$$

$$\stackrel{(i)}{=} \frac{1}{\sqrt{2\pi\alpha}} \int_{-\infty}^{\infty} \exp\left(-\frac{y^2}{2\alpha}\right) \cos(2\pi\omega y) \, dy \quad (250)$$

$$\stackrel{(ii)}{=} \exp\left(-\omega^2\pi^2 2\alpha\right), \quad (251)$$

where (i) follows from the fact that the second term is an integral of an odd function over a symmetric range and (ii) follows from $\int_{-\infty}^{\infty} \exp(-ay^2) \cos(2\pi\omega y) \, dy = \sqrt{\frac{\pi}{a}} \exp(\frac{-(\pi\omega)^2}{a})$ with $a = \frac{1}{2\alpha}$ (see p. 302, eq. 7.4.6 in [1]). This concludes our proof since

$$(\mathcal{F}(G_\alpha) \cdot \mathcal{F}(G_\beta))(\omega) = \exp\left(-\omega^2\pi^2 2\alpha\right) \cdot \exp\left(-\omega^2\pi^2 2\beta\right) \quad (252)$$

$$= \exp\left(-\omega^2\pi^2 2(\alpha + \beta)\right) \quad (253)$$

$$= \mathcal{F}(G_{\alpha+\beta})(\omega) \quad (254)$$

and hence

$$(G_\alpha * G_\beta) = \mathcal{F}^{-1}\{\mathcal{F}(G_\alpha) \cdot \mathcal{F}(G_\beta)\} \quad (255)$$

$$= \mathcal{F}^{-1}\{\mathcal{F}(G_{\alpha+\beta})\} \quad (256)$$

$$= G_{\alpha+\beta}. \quad (257)$$

$\square$

REMARK 1. *We notice that the preceding theorem naturally extends to higher dimensional Gaussian kernels of the form*

$$G_\alpha(k) = \frac{1}{(2\pi\alpha)^{\frac{m}{2}}} \exp\left(-\frac{\|k\|^2}{2\alpha}\right), \qquad k \in \mathbb{R}^m. \quad (258)$$

Consider

$$\mathcal{F}(G_\alpha)(\omega) = \int_{\mathbb{R}^m} G_\alpha(y) \exp(-2\pi i\langle\omega, y\rangle) \, dy \quad (259)$$

$$= \frac{1}{(2\pi\alpha)^{\frac{m}{2}}} \int_{\mathbb{R}^m} \exp\left(-\frac{\|y\|_2^2}{2\alpha} - 2\pi i\langle\omega, y\rangle\right) dy \quad (260)$$

$$= \prod_{j=1}^{m}\left(\frac{1}{\sqrt{2\pi\alpha}} \int_{\mathbb{R}} \exp\left(-\frac{y_j^2}{2\alpha} - 2\pi i\omega_j y_j\right) dy_j\right) \quad (261)$$

$$= \exp\left(-\|\omega\|_2^2 \pi^2 2\alpha\right) \quad (262)$$

that leads to $(G_\alpha * G_\beta) = G_{\alpha+\beta}$, and hence additivity.

F.3 Brightness and contrast

Recall that the brightness and contrast transformation is defined as

$$\phi_{BC} \colon \mathcal{X} \times \mathbb{R}^2 \to \mathcal{X}, \quad (x, \alpha) \mapsto e^{\alpha_1}(x + \alpha_2). \quad (263)$$

Lemma 2 (restated). *Let $x \in \mathcal{X}$, $k \in \mathbb{R}$, $\varepsilon_0 \sim \mathcal{N}(0, \text{diag}(\sigma^2, \tau^2))$ and $\varepsilon_1 \sim \mathcal{N}(0, \text{diag}(\sigma^2, e^{-2k}\tau^2))$. Suppose that $q(y|x; \varepsilon_0) \geq p$ for some $p \in [0, 1]$ and $y \in \mathcal{Y}$. Then*

$$q(y|x; \varepsilon_1) \geq \begin{cases} 2\Phi\left(e^k\Phi^{-1}\left(\frac{1+p}{2}\right)\right) - 1 & k \leq 0 \\ 2\left(1 - \Phi\left(e^k\Phi^{-1}(1 - \frac{p}{2})\right)\right) & k > 0. \end{cases} \quad (264)$$

PROOF. Note that $\varepsilon_0 \sim \mathcal{N}(0, \Sigma)$ and $\varepsilon_1 = A\varepsilon_0 \sim \mathcal{N}(0, A^2\Sigma)$ where

$$A = \begin{pmatrix} 1 & 0 \\ 0 & e^{-k} \end{pmatrix}, \quad \Sigma = \begin{pmatrix} \sigma^2 & 0 \\ 0 & \tau^2 \end{pmatrix} \quad (265)$$

and denote by f_0 and f_1 the probability density functions of ε_0 and ε_1, respectively, and denote by $\mathbb{P}_0$ and $\mathbb{P}_1$ the corresponding

probability measures. Recall the Definition of Lower Level sets (Definition 7: for $t \geq 0$, (strict) lower level sets are defined as

$$\underline{S_t} := \{z \in \mathcal{Z} : \Lambda(z) < t\}, \quad \overline{S_t} := \{z \in \mathcal{Z} : \Lambda(z) \leq t\},$$
$$\text{where} \quad \Lambda(z) := \frac{f_1(z)}{f_0(z)}. \tag{266}$$

Furthermore, recall that the function ζ is given by

$$t \mapsto \zeta(t) := \mathbb{P}_0\left(\overline{S}_t\right) \tag{267}$$

where $\mathbb{P}_0$ is the distribution of ε_0 and note that the generalized inverse of ζ corresponds to τ_p, i.e.,

$$\zeta^{-1}(p) = \inf\{t \geq 0 | \zeta(t) \geq p\} = \tau_p \tag{268}$$

and the function ξ is correspondingly given by

$$\xi(p) = \sup\{\mathbb{P}_1(S) | \underline{S}(\zeta^{-1}(p)) \subseteq S \subseteq \overline{S}(\zeta^{-1}(p))\}. \tag{269}$$

By assumption we know that $\mathbb{E}(p(y | \phi(x, \varepsilon_0))) = q(y | x; \varepsilon_o) \geq p$. Note that by Lemma 6, for any $p \in [0, 1]$ we have that

$$\mathbb{P}_0(\underline{S}_{\zeta^{-1}(p)}) \leq p. \tag{270}$$

Let $S \subseteq \mathcal{Z}$ be such that $\underline{S}_{\zeta^{-1}(p)} \subseteq S \subseteq \overline{S}_{\zeta^{-1}(p)}$ and $\mathbb{P}_0(S) \leq p$. Then, from part (i) of Lemma 7, it follows that $\mathbb{E}(p(y | \phi(x, \varepsilon_1))) = q(y | x; \varepsilon_1) \geq \mathbb{P}_1(S)$. Note that

$$\Lambda(z) = \frac{f_1(z)}{f_0(z)} \tag{271}$$

$$= \frac{((2\pi)^2 |A^2 \Sigma|)^{-\frac{1}{2}} \exp(-\frac{1}{2}(z^T (A^2 \Sigma)^{-1} z))}{((2\pi)^2 |\Sigma|)^{-\frac{1}{2}} \exp(-\frac{1}{2}(z^T (\Sigma)^{-1} z))} \tag{272}$$

$$= \frac{1}{|A|} \exp\left(-\frac{1}{2} z^T \left((A^2 \Sigma)^{-1} - \Sigma^{-1}\right) z\right) \tag{273}$$

$$= \exp\left(k - \frac{z_2^2}{2\tau^2} \left(e^{2k} - 1\right)\right). \tag{274}$$

Note that, if $k = 0$, then $\varepsilon_1 = \varepsilon_0$ and hence the statement holds in this case. Suppose that $k > 0$ and consider

$$\zeta(t) = \mathbb{P}_0\left(\overline{S}_t\right) = \mathbb{P}\left(\exp\left(k - \frac{\varepsilon_{0,2}^2}{2\tau^2} \left(e^{2k} - 1\right)\right) \leq t\right) \tag{275}$$

$$= 1 - \mathbb{P}\left(\left(\frac{\varepsilon_{0,2}}{\tau}\right)^2 \leq 2 \cdot \frac{k - \log(t)}{e^{2k} - 1}\right) \tag{276}$$

$$= 1 - F_{\chi^2}\left(2 \cdot \frac{k - \log(t)}{e^{2k} - 1}\right) \tag{277}$$

$$= \begin{cases} 0 & t = 0, \\ 1 - F_{\chi^2}\left(2 \cdot \frac{k - \log(t)}{e^{2k} - 1}\right) & 0 < t < e^k, \\ 1 & t \geq e^k, \end{cases} \tag{278}$$

where F_{χ^2} denotes the CDF of the χ^2-distribution with one degree of freedom. Note that for any $t \geq 0$ we have that $\mathbb{P}_0(\overline{S}_t) = \mathbb{P}_0(\underline{S}_t)$ and thus the inverse $\zeta^{-1}(p) = \inf\{t \geq 0 : \zeta(t) \geq p\}$ is given by

$$\zeta^{-1}(p) = \begin{cases} 0 & p = 0 \\ \exp\left(k - F_{\chi^2}^{-1}(1 - p) \cdot \frac{e^{2k} - 1}{2}\right) & 0 < p < 1 \\ e^k & p = 1. \end{cases} \tag{279}$$

Thus, for any $p \in [0, 1]$, we find that

$$\mathbb{P}_0(\overline{S}_{\zeta^{-1}(p)}) = \mathbb{P}_0(\underline{S}_{\zeta^{-1}(p)}) = \zeta(\zeta^{-1}(p)) = p \tag{280}$$

and

$$\mathbb{E}_0(p(y | \phi(x, \varepsilon_0))) = q(y | x; \varepsilon_0) \geq p = \mathbb{P}_0(\overline{S}_{\zeta^{-1}(p)}). \tag{281}$$

Part (i) of Lemma 7 implies that $q(y | x; \varepsilon_1) \geq \mathbb{P}_1(\overline{S}_{\zeta^{-1}(p)})$. Computing $\mathbb{P}_1(\overline{S}_{\zeta^{-1}(p)})$ yields

$$q(y | x; \varepsilon_1) \geq \mathbb{P}_1(\overline{S}_{\zeta^{-1}(p)}) \tag{282}$$

$$= 1 - \mathbb{P}\left(\left(\frac{\varepsilon_{1,2}}{\tau^2}\right)^2 \leq (k - \log(\zeta^{-1}(p))) \frac{2}{e^{2k} - 1}\right) \tag{283}$$

$$= 1 - \mathbb{P}\left(\left(\frac{\varepsilon_{0,2}}{\tau^2}\right)^2 \leq (k - \log(\zeta^{-1}(p))) \frac{2e^{2k}}{e^{2k} - 1}\right) \tag{284}$$

$$= 1 - F_{\chi^2}\left((k - \log(\zeta^{-1}(p))) \frac{2e^{2k}}{e^{2k} - 1}\right) \tag{285}$$

$$= 1 - F_{\chi^2}\left(\left(k - \left(k - \frac{e^{2k} - 1}{2} F_{\chi^2}^{-1}(1 - p)\right)\right) \frac{2e^{2k}}{e^{2k} - 1}\right) \tag{286}$$

$$= 1 - F_{\chi^2}\left(e^{2k} F_{\chi^2}^{-1}(1 - p)\right). \tag{287}$$

If, on the other hand, $k < 0$, then

$$\zeta(t) = \mathbb{P}_0\left(\overline{S}_t\right) \tag{288}$$

$$= \mathbb{P}\left(\exp\left(k + \frac{\varepsilon_{0,2}^2}{2\tau^2} \left|e^{2k} - 1\right|\right) \leq t\right) \tag{289}$$

$$= \mathbb{P}\left(\left(\frac{\varepsilon_{0,2}}{\tau}\right)^2 \leq 2 \cdot \frac{\log(t) - k}{\left|e^{2k} - 1\right|}\right) \tag{290}$$

$$= F_{\chi^2}\left(2 \cdot \frac{\log(t) - k}{\left|e^{2k} - 1\right|}\right) \tag{291}$$

$$= \begin{cases} 0 & t \leq e^k, \\ F_{\chi^2}\left(2 \cdot \frac{\log(t) - k}{\left|e^{2k} - 1\right|}\right) & t > e^k. \end{cases} \tag{292}$$

A similar computation as in the case where $k > 0$ leads to an expression for the inverse $\zeta^{-1}(p) = \inf\{t \geq 0 | \zeta(t) \geq p\}$

$$\zeta^{-1}(p) = \begin{cases} 0 & p = 0, \\ \exp\left(k + F_{\chi^2}^{-1}(p) \cdot \frac{|e^{2k} - 1|}{2}\right) & p > 0. \end{cases} \tag{293}$$

Thus, for any $p \in [0, 1]$, we find that

$$\mathbb{P}_0(\overline{S}_{\zeta^{-1}(p)}) = \mathbb{P}_0(\underline{S}_{\zeta^{-1}(p)}) = \zeta(\zeta^{-1}(p)) = p \tag{294}$$

and

$$\mathbb{E}(p(y | \phi(x, \varepsilon_0))) = q(y | x; \varepsilon_0) \geq p = \mathbb{P}_0(\overline{S}_{\zeta^{-1}(p)}). \tag{295}$$

Part (i) of Lemma 7 implies that $g_c^{\varepsilon_1}(x) \geq \mathbb{P}_1(\overline{S}_{\zeta^{-1}(p)})$. Computing $\mathbb{P}_1(\overline{S}_{\zeta^{-1}(p)})$ yields

$$q(y | x; \varepsilon_1) \geq \mathbb{P}_1(\overline{S}_{\zeta^{-1}(p)}) \tag{296}$$

$$= \mathbb{P}\left(\left(\frac{\varepsilon_{1,2}}{\tau}\right)^2 \leq 2 \cdot \frac{\log(\zeta^{-1}(p)) - k}{\left|e^{2k} - 1\right|}\right) \tag{297}$$

$$= \mathbb{P}\left(\left(\frac{\varepsilon_{0,2}}{\tau}\right)^2 \leq 2e^{2k} \cdot \frac{\log(\zeta^{-1}(p)) - k}{\left|e^{2k} - 1\right|}\right) \tag{298}$$

$$= F_{\chi^2}\left(\left(\left(k + F_{\chi^2}^{-1}(p)\frac{|e^{2k} - 1|}{2}\right) - k\right)\frac{2\,e^{2k}}{|e^{2k} - 1|}\right) \quad (299)$$

$$= F_{\chi^2}\left(e^{2k}F_{\chi^2}^{-1}(p)\right). \quad (300)$$

Finally, note the following relation between the $\chi^2(1)$ and the standard normal distribution. Let $Z \sim \mathcal{N}(0, 1)$ and denote by Φ the CDF of Z. Then, for any $z \geq 0$, $F_{\chi^2}(z) = \mathbb{P}(Z^2 \leq z) = \mathbb{P}(-\sqrt{z} \leq Z \leq \sqrt{z}) = \Phi(\sqrt{z}) - \Phi(-\sqrt{z}) = 2\Phi(\sqrt{z}) - 1$ and the inverse is thus given by $F_{\chi^2}^{-1}(p) = (\Phi^{-1}(\frac{1+p}{2}))^2$. It follows that

$$q(y|, x; \varepsilon_1) \geq \begin{cases} 2\Phi\left(e^k\Phi^{-1}\left(\frac{1+p}{2}\right)\right) - 1 & k \leq 0, \\ 2\left(1 - \Phi\left(e^k\Phi^{-1}(1 - \frac{p}{2})\right)\right) & k > 0, \end{cases} \quad (301)$$

what concludes the proof.   □

The following lemma establishes another useful property of the distribution of ε_1.

LEMMA 8. *Let* $\varepsilon_0 \sim \mathcal{N}(0, \text{diag}(\sigma^2, \tau^2))$, $\alpha = (k, b)^T \in \mathbb{R}^2$ *and* $\varepsilon_1 \sim \mathcal{N}(0, \text{diag}(\sigma^2, e^{-2k}\tau^2))$. *Then, for all* $x \in \mathcal{X}$, *it holds that* $g(\phi_{BC}(x, \alpha); \varepsilon_0) = g(x; \alpha + \varepsilon_1)$.

PROOF. Let $x \in \mathcal{X}$, and write $\varepsilon_i = (\varepsilon_{i,1}, \varepsilon_{i,2})^T$ for $i = 0, 1$. Note that

$$\phi_{BC}(\phi_{BC}(x, \alpha), \varepsilon_0) = e^{\varepsilon_{0,1}}\left(\phi_{BC}(x, \alpha) + \varepsilon_{0,2}\right) = e^{\varepsilon_{0,1}}\left(e^k\,(x + b) + \varepsilon_{0,2}\right) \quad (302)$$

$$= e^{\varepsilon_{0,1}+k}\left(x + \left(b + e^{-k}\varepsilon_{0,2}\right)\right) = \phi_{BC}(x, \alpha + \tilde{\varepsilon}_0) \quad (303)$$

where $\tilde{\varepsilon}_0 = (\varepsilon_{0,1}, e^{-k}\varepsilon_{0,2})^T$. Note that $\tilde{\varepsilon}_0$ follows a Gaussian distribution since

$$\tilde{\varepsilon}_0 = A \cdot \varepsilon_0, \quad A = \begin{pmatrix} 1 & 0 \\ 0 & e^{-k} \end{pmatrix} \quad (304)$$

and hence $\mathbb{E}(\tilde{\varepsilon}_0) = A \cdot \mathbb{E}(\varepsilon_0) = 0$ and

$$\text{Cov}(\tilde{\varepsilon}_0) = \mathbb{E}\left(\varepsilon_0\,A\,A^T\,\varepsilon_0^T\right) = A^2 \cdot \begin{pmatrix} \sigma^2 & 0 \\ 0 & \tau^2 \end{pmatrix} = \begin{pmatrix} \sigma^2 & 0 \\ 0 & e^{-2k}\tau^2 \end{pmatrix}. \quad (305)$$

The choice $\varepsilon_1 \equiv \tilde{\varepsilon}_0 \sim \mathcal{N}(0, \text{diag}(\sigma_1^2, e^{-2k}\sigma_2^2))$ shows that for any $y \in \mathcal{Y}$

$$q(y|\phi_{BC}(x, \alpha); \varepsilon_0) = \mathbb{E}(p(y|\phi(\phi(x, \alpha), \varepsilon_0)) \quad (306)$$

$$= \mathbb{E}(p(y|\phi(x, \alpha + \varepsilon_1)) \quad (307)$$

$$= q(y|x; \alpha + \varepsilon_1) \quad (308)$$

what concludes the proof.   □

These observations, together with the Gaussian robustness bound from Corollary 7 allow us to prove Lemma 3.

Lemma 3 (restated). *Let* ε_0 *and* ε_1 *be as in Lemma 2 and suppose that*

$$q(y_A|x; \varepsilon_1) \geq \tilde{p}_A > \tilde{p}_B \geq \max_{y \neq y_A} q(y|x; \varepsilon_1). \quad (309)$$

Then it is guaranteed that $y_A = g(\phi_{BC}(x, \alpha); \varepsilon_0)$ *as long as* $\alpha = (k, b)^T$ *satisfies*

$$\sqrt{\left(\frac{k}{\sigma}\right)^2 + \left(\frac{b}{e^{-k}\tau}\right)^2} < \frac{1}{2}\left(\Phi^{-1}(\tilde{p}_A) - \Phi^{-1}(\tilde{p}_B)\right). \quad (310)$$

PROOF. Since $\varepsilon_1 \sim \mathcal{N}(0, \text{diag}(\sigma^2, e^{-2k}\tau^2))$, it follows from Corollary 7 that whenever $\alpha = (k, b)^T$ satisfies

$$\sqrt{\left(\frac{k}{\sigma}\right)^2 + \left(\frac{b}{e^{-k}\tau}\right)^2} < \frac{1}{2}\left(\Phi^{-1}(\tilde{p}_A) - \Phi^{-1}(\tilde{p}_B)\right), \quad (311)$$

then it is guaranteed that $y_A = g(x; \varepsilon_1)$. The statement now directly follows from Lemma 8.   □

F.4 Gaussian Blur, Brightness, Contrast, and Translation

Recall that the composition of Gaussian Blur, with brightness, contrast and translation is defined as

$$\phi_{BTBC}(x, \alpha) := \phi_B(\phi_T(\phi_{BC}(x, \alpha_k, \alpha_b), \alpha_{Tx}, \alpha_{Ty}), \alpha_B), \quad (312)$$

where ϕ_B, ϕ_T and ϕ_{BC} are Gaussian blur, translation, and brightness and contrast transformations respectively as defined before and $\alpha := (\alpha_k, \alpha_b, \alpha_{Tx}, \alpha_{Ty}, \alpha_B)^T \in \mathbb{R}^4 \times \mathbb{R}_{\geq 0}$ is the transformation parameter. It is easy to see that this transformation composition satisfies the following properties:

- **(P1)** For arbitrary $\alpha^{(1)}, \alpha^{(2)} \in \mathbb{R}^4 \times \mathbb{R}_{\geq 0}$,

$$\phi_{BTBC}(\phi_{BTBC}(x, \alpha^{(1)}), \alpha^{(2)}) = \phi_{BTBC}(x, \alpha) \quad (313)$$

where

$$\alpha = \left(\alpha_k^{(1)} + \alpha_k^{(2)}, \alpha_b^{(1)} + \alpha_b^{(2)}/e^{\alpha_k^{(1)}},\right.$$
$$\left.\alpha_{Tx}^{(1)} + \alpha_{Tx}^{(2)}, \alpha_{Tx}^{(1)} + \alpha_{Tx}^{(2)}, \alpha_B^{(1)} + \alpha_B^{(2)}\right). \quad (314)$$

- **(P2)** For an arbitrary $\alpha \in \mathbb{R}^4 \times \mathbb{R}_{\geq 0}$, define the parameterized operators:

$$\phi_B^\alpha := \phi_B(\cdot; \alpha_B), \quad \phi_T^\alpha := \phi_T(\cdot; \alpha_{Tx}, \alpha_{Ty}),$$
$$\phi_{BC}^\alpha := \phi_{BC}(\cdot; \alpha_k, \alpha_b) \quad (315)$$

and let $\phi_1^\alpha, \phi_2^\alpha, \phi_3^\alpha$ be an arbitrary permutation of the above three operators. Then, we have that

$$\phi_{BTBC}(x, \alpha) = \phi_1^\alpha \circ \phi_2^\alpha \circ \phi_3^\alpha(x). \quad (316)$$

The property **(P1)** states that ϕ_{BTBC} is almost additive where the exception happens only on the brightness dimension (α_b). The brightness dimension is subject to the same contrast effect implied and proved in Lemma 8 (in main paper). The property **(P2)** states that all the three transformations ϕ_B, ϕ_T, and ϕ_{BC} are commutative. The reason is that: (1) ϕ_{BC} is a per-pixel color shift and independent of ϕ_B and ϕ_T; (2) ϕ_B, Gaussian blur, relies on relative position of pixels and the translation with reflection padding, ϕ_T, does not change it.

Based on these two properties, we prove the key results as follows.

Corollary 3 (restated). *Let $x \in \mathcal{X}$, $k \in \mathbb{R}$ and let $\varepsilon_0 := (\varepsilon_0^a, \varepsilon_0^b)^T$ be a random variable defined as*

$$\varepsilon_0^a \sim \mathcal{N}(0, \mathrm{diag}(\sigma_k^2, \sigma_b^2, \sigma_T^2, \sigma_T^2)) \text{ and } \varepsilon_0^b \sim \mathrm{Exp}(\lambda_B). \tag{317}$$

Similarly, let $\varepsilon_1 := (\varepsilon_1^a, \varepsilon_1^b)$ be a random variable with

$$\varepsilon_1^a \sim \mathcal{N}(0, \mathrm{diag}(\sigma_k^2, e^{-2k}\sigma_b^2, \sigma_T^2, \sigma_T^2)) \text{ and } \varepsilon_1^b \sim \mathrm{Exp}(\lambda_B). \tag{318}$$

For either random variable (denoted as ε), recall that $q(y|x; \varepsilon) := \mathbb{E}(p(y|\phi_{BTBC}(x, \varepsilon)))$. Suppose that $q(y|x; \varepsilon_0) \geq p$ for some $p \in [0, 1]$ and $y \in \mathcal{Y}$. Then $q(y|x; \varepsilon_1)$ satisfies Eq. (11).

PROOF. According to the commutative property **(P2)**, we can view $q(y|x; \varepsilon)$ as

$$q(y|x, \varepsilon) = \mathbb{E}_\varepsilon p(y|\phi_{BTBC}(x, \varepsilon)) \tag{319}$$

$$= \mathbb{E}_{\varepsilon_k, \varepsilon_b} \underbrace{\mathbb{E}_{\varepsilon_{Tx}, \varepsilon_{Ty}, \varepsilon_B} p(y|\phi_{BC}(\phi_T(\phi_B(x, \varepsilon_B), \varepsilon_{Tx}, \varepsilon_{Ty}), \varepsilon_k, \varepsilon_b))}_{=:q'(y|x; \varepsilon_k, \varepsilon_b)}. \tag{320}$$

Notice that $q'(y|x; \varepsilon_k, \varepsilon_b)$ is a deterministic value in $[0, 1]$. Its value is dependent on the distribtuion of $\varepsilon_{Tx}, \varepsilon_{Ty}, \varepsilon_B$ and the underlying base classifier. Luckily, the random variables ε_0 and ε_1 have the same distribution over the components $\varepsilon_{Tx}, \varepsilon_{Ty}$ and ε_B. Thus, they share the same q' and we write $q(y|x; \varepsilon_0)$ and $q(y|x; \varepsilon_1)$ as

$$q(y|x; \varepsilon_0) = \mathbb{E}_{(\varepsilon_k, \varepsilon_b) \sim \mathcal{N}(0, \mathrm{diag}(\sigma_k^2, \sigma_b^2))} q'(y|x; \varepsilon_k, \varepsilon_b), \tag{321}$$

$$q(y|x; \varepsilon_1) = \mathbb{E}_{(\varepsilon_k, \varepsilon_b) \sim \mathcal{N}(0, \mathrm{diag}(\sigma_k^2, e^{-2k}\sigma_b^2))} q'(y|x; \varepsilon_k, \varepsilon_b). \tag{322}$$

Now, we directly apply Lemma 2 and the desired lower bound for $q(y|x; \varepsilon_1)$ follows. □

Lemma B.1 (restated). *Let ε_0 and ε_1 be as in Corollary 3 and suppose that*

$$q(y_A|x; \varepsilon_1) \geq \tilde{p}_A > \tilde{p}_B \geq \max_{y \neq y_A} q(y|x; \varepsilon_1). \tag{323}$$

Then it is guaranteed that $y_A = g(\phi_{BTBC}(x, \alpha); \varepsilon_0)$ as long as $p'_A > p'_B$,

$$p'_A = \begin{cases} 0, & \text{if } \tilde{p}_A \leq 1 - \exp(-\lambda_B \alpha_B), \\ \Phi\left(\Phi^{-1}\left(1 - (1 - \tilde{p}_A)\exp(\lambda_B \alpha_B)\right) \right. \\ \left. - \sqrt{\alpha_k^2/\sigma_k^2 + \alpha_b^2/(e^{-2\alpha_k}\sigma_b^2) + (\alpha_{Tx}^2 + \alpha_{Ty}^2)/\sigma_T^2}\right), & \text{otherwise} \end{cases} \tag{324}$$

and

$$p'_B = \begin{cases} 1, & \text{if } \tilde{p}_B \geq \exp(-\lambda_B \alpha_B), \\ 1 - \Phi\left(\Phi^{-1}\left(1 - \tilde{p}_B \exp(\lambda_B \alpha_B)\right) \right. \\ \left. - \sqrt{\alpha_k^2/\sigma_k^2 + \alpha_b^2/(e^{-2\alpha_k}\sigma_b^2) + (\alpha_{Tx}^2 + \alpha_{Ty}^2)/\sigma_T^2}\right). & \text{otherwise} \end{cases} \tag{325}$$

PROOF. We notice that for any $y \in \mathcal{Y}$,

$$q(y|\phi_{BTBC}(x, \alpha); \varepsilon_0)$$

$$= \mathbb{E}_{\varepsilon_0} p(y|\phi_{BTBC}(\phi_{BTBC}(x, \alpha), \varepsilon_0)) \tag{326}$$

$$\stackrel{(a.)}{=} \mathbb{E}_{\varepsilon_0} p\left(y\middle|x; \alpha + \right. \tag{327}$$

$$\left. ((\varepsilon_0)_k, (\varepsilon_0)_b/e^{\alpha_k}, (\varepsilon_0)_{Tx}, (\varepsilon_0)_{Ty}, (\varepsilon_0)_B)^T\right)$$

$$\stackrel{(b.)}{=} E_{\varepsilon_1} p(y|x; \alpha + \varepsilon_1). \tag{328}$$

The step $(a.)$ uses the property **(P1)** of transformation ϕ_{BTBC}, and the step $(b.)$ follows the definition of ε_1 in Corollary 3 (we define $k := \alpha_k$ hereinafter for simplicity). Thus, $g(\phi_{BTBC}(x, \alpha); \varepsilon_0) = g(x; \alpha + \varepsilon_1)$, and the robustness condition is equivalent to $g(x; \alpha + \varepsilon_1) = g(x; \varepsilon_1) = y_A$.

According to Theorem 1, to prove the robustness, we only need to show that $\xi(\tilde{p}_A) + \xi(1 - \tilde{p}_B) > 1$ given $p'_A > p'_B$. Note that in the definition of ξ, the density functions f_0 and f_1 are for distributions of $\varepsilon_1 \sim \mathbb{P}_0$ and $(\alpha + \varepsilon_1) \sim \mathbb{P}_1$ respectively.

In the proof below, we will compute the closed-form solution of $\xi(p)$ for any $0 \leq p \leq 1$, and show that $\xi(\tilde{p}_A) + \xi(1 - \tilde{p}_B) > 1$ given $p'_A > p'_B$. To begin with, we write down f_0 and f_1.

$$f_0(z) = \frac{\lambda_B}{(2\pi)^2 \sigma_k \sigma_b \sigma_T^2} \exp\left(-\lambda_B z_B - (z_{Tx}^2 + z_{Ty}^2)/2\sigma_T^2 \right. \\ \left. - z_k^2/2\sigma_k^2 - z_b^2/2e^{-2k}\sigma_b^2\right), \tag{329}$$

$$f_1(z) = \begin{cases} \frac{\lambda_B \exp(\lambda_B \alpha_B)}{(2\pi)^2 \sigma_k \sigma_b \sigma_T^2} \exp\left(-\lambda_B z_B \right. \\ \quad - (z_{Tx} - \alpha_{Tx})^2/2\sigma_T^2 - (z_{Ty} - \alpha_{Ty})^2/2\sigma_T^2 \\ \left. \quad - (z_k - \alpha_k)^2/2\sigma_k^2 - (z_b - \alpha_b)^2/2e^{-2k}\sigma_b^2\right), & \text{if } z_B \geq \alpha_B, \\ 0, & \text{otherwise,} \end{cases} \tag{330}$$

where $z = (z_k, z_b, z_{Tx}, z_{Ty}, z_B)^T \in \mathbb{R}^4 \times R_{\geq 0}$. As a result, function $\Lambda = f_1/f_0$ in Theorem 1 writes as

$$\Lambda(z) = \begin{cases} \exp\left(\lambda_B \alpha_B - \frac{\alpha_{Tx}^2}{2\sigma_T^2} - \frac{\alpha_{Ty}^2}{2\sigma_T^2} - \frac{\alpha_k^2}{2\sigma_k^2} - \frac{\alpha_b^2}{2e^{-2k}\sigma_b^2} \right. \\ \left. + \frac{\alpha_{Tx}z_{Tx}}{\sigma_T^2} + \frac{\alpha_{Ty}z_{Ty}}{\sigma_T^2} + \frac{\alpha_k z_k}{\sigma_k^2} + \frac{\alpha_b z_b}{e^{-2k}\sigma_b^2}\right), & \text{if } z_B \geq \alpha_B, \\ 0 & \text{otherwise.} \end{cases} \tag{331}$$

It turns out that for any $t > 0$,

$$\underline{S}_t = \{f_1/f_0 < t\}$$

$$= \left\{(\hat{z}_k \sigma_k, \hat{z}_b e^{-k}\sigma_b, \hat{z}_{Tx}\sigma_T, \hat{z}_{Ty}\sigma_T)^T \right.$$

$$\left. \middle| \hat{\alpha}_{Tx}\hat{z}_{Tx} + \hat{\alpha}_{Ty}\hat{z}_{Ty} + \hat{\alpha}_k \hat{z}_k + \hat{\alpha}_b \hat{z}_b \right.$$

$$\left. < \ln t + \hat{\alpha}_{Tx}^2/2 + \hat{\alpha}_{Ty}^2/2 + \hat{\alpha}_k^2/2 + \hat{\alpha}_b^2/2 - \lambda_B \alpha_B\right\} \times [\alpha_B, +\infty)$$

$$\cup \mathbb{R}^4 \times [0, \alpha_B), \tag{332}$$

$$\overline{S}_t = \{f_1/f_0 \leq t\}$$

$$= \left\{(\hat{z}_k \sigma_k, \hat{z}_b e^{-k}\sigma_b, \hat{z}_{Tx}\sigma_T, \hat{z}_{Ty}\sigma_T)^T \right.$$

$$\left. \middle| \hat{\alpha}_{Tx}\hat{z}_{Tx} + \hat{\alpha}_{Ty}\hat{z}_{Ty} + \hat{\alpha}_k \hat{z}_k + \hat{\alpha}_b \hat{z}_b \right.$$

$$\left. \leq \ln t + \hat{\alpha}_{Tx}^2/2 + \hat{\alpha}_{Ty}^2/2 + \hat{\alpha}_k^2/2 + \hat{\alpha}_b^2/2 - \lambda_B \alpha_B\right\} \times [\alpha_B, +\infty)$$

$$\cup \mathbb{R}^4 \times [0, \alpha_B), \tag{333}$$

where $\hat{\alpha}_{Tx} = \alpha_{Tx}/\sigma_T$, $\hat{\alpha}_{Ty} = \alpha_{Ty}/\sigma_T$, $\hat{\alpha}_k = \alpha_k/\sigma_k$, $\hat{\alpha}_b = \alpha_b/(e^{-k}\sigma_b)$. When $t = 0$, $\underline{S}_t = \emptyset$ and $\overline{S}_t = \mathbb{R}^4 \times [0, \alpha_B]$. Then, the probability integration shows that

$$\tau_p = \inf\{t \geq 0 : \mathbb{P}_0(\overline{S}_t) \geq p\}$$

$$= \begin{cases} 0, & \text{if } p \leq 1 - \exp(-\lambda_B\alpha_B), \\ \exp\left(\lambda_B\alpha_B + \|\hat{\alpha}_{:-1}\|\Phi^{-1}\left(1 - \exp(\lambda_B\alpha_B)(1-p)\right) - 1/2\|\hat{\alpha}_{:-1}\|^2\right), \\ & \text{otherwise,} \end{cases}$$
$$(334)$$

where $\|\hat{\alpha}_{:-1}\| = \sqrt{\hat{\alpha}_{Tx}^2 + \hat{\alpha}_{Ty}^2 + \hat{\alpha}_k^2 + \hat{\alpha}_b^2}$. Now we are ready to compute $\xi(p) = \sup\{\mathbb{P}_1(S) : \underline{S}_{\tau_p} \subset S \subset \overline{S}_{\tau_p}\}$. When $p \leq 1 - \exp(-\lambda_B\alpha_B)$, we have $S \subset \mathbb{R}^4 \times [0, \alpha_B)$ and $\mathbb{P}_1(S) = 0$ because $\mathbb{P}_1$ has zero mass for any $z_B < \alpha_B$ (see (330)). When $p > 1 - \exp(-\lambda_B\alpha_B)$, $\tau_p > 0$. Again, from probability integration, we get

$$\mathbb{P}_1(\underline{S}_{\tau_p}) = \mathbb{P}_1(\overline{S}_{\tau_p}) = \Phi\left(\frac{\ln \tau_p - \lambda_B\alpha_B}{\|\hat{\alpha}_{:-1}\|} - \frac{1}{2}\|\hat{\alpha}_{:-1}\|\right). \quad (335)$$

We inject the closed-form solution of τ_p in (334) and yield

$$\xi(p) = \mathbb{P}_1(S) = \Phi\left(\Phi^{-1}\left(1 - (1-p)\exp(\lambda_B\alpha_B)\right) - \|\hat{\alpha}_{:-1}\|\right) \quad (336)$$

for any S satisfying $\underline{S}_{\tau_p} \subset S \subset \overline{S}_{\tau_p}$. We summarize the above equations and write down the closed-form solution of $\xi(p)$ as such:

$$\xi(p) = \begin{cases} 0, & \text{if } p \leq 1 - \exp(-\lambda_B\alpha_B), \\ \Phi\left(\Phi^{-1}\left(1 - (1-p)\exp(\lambda_B\alpha_B)\right) - \|\hat{\alpha}_{:-1}\|\right), & \text{otherwise.} \end{cases}$$
$$(337)$$

We can easily observe that p'_A in lemma statement (324) is indeed $\xi(\tilde{p}_A)$, and p'_B (325) is indeed $1 - \xi(1 - \tilde{p}_B)$. Therefore,

$$\xi(\tilde{p}_A) + \xi(1 - \tilde{p}_B) > 1 \iff p'_A > p'_B \quad (338)$$

and using Theorem 1 concludes the proof. □

Finally, we simplify the statement of Lemma B.1 with slight relaxation.

Corollary 4 (restated). *Let ε_0 and ε_1 be as in Corollary 3 and suppose that*

$$q(y_A|x; \varepsilon_1) \geq \tilde{p}_A. \quad (339)$$

Then it is guaranteed that $y_A = g(\phi_{BTBC}(x, \alpha); \varepsilon_0)$ as long as

$$\tilde{p}_A > 1 - \exp(-\lambda_B\alpha_B)\left(1 - \Phi\left(\sqrt{\frac{\alpha_k^2}{\sigma_k^2} + \frac{\alpha_b^2}{e^{-2\alpha_k}\sigma_b^2} + \frac{\alpha_{Tx}^2 + \alpha_{Ty}^2}{\sigma_T^2}}\right)\right).$$
$$(340)$$

PROOF. Since $q(y_A|x; \varepsilon_1) \geq \tilde{p}_A$, according to the complement rule, $\max_{y \neq y_A} q(y|x; \varepsilon_1) < 1 - \tilde{p}_A =: \tilde{p}_B$. Inject the $\tilde{p}_A$ and $\tilde{p}_B$ into Lemma B.1 we find that $p'_A + p'_B = 1$ always hold. Therefore, $p'_A > 0.5$ guarantees that $p'_A > p'_B$ and thus the robustness. Indeed, from simple algebra, $p'_A > 0.5 \iff (340)$. □

G PROOFS FOR DIFFERENTIALLY RESOLVABLE TRANSFORMATIONS

Here we provide proofs and technical details for theoretical results about certifying differentially resolvable transformations. First, let us recall the definition of differentially resolvable transformations.

Definition 3 (restated). *Let $\phi: X \times \mathcal{Z}_\phi \to X$ be a transformation with noise space $\mathcal{Z}_\phi$ and let $\psi: X \times \mathcal{Z}_\psi \to X$ be a resolvable transformation with noise space $\mathcal{Z}_\psi$. We say that ϕ can be resolved by ψ if for any $x \in X$ there exists function $\delta_x: \mathcal{Z}_\psi \times \mathcal{Z}_\phi \to \mathcal{Z}_\psi$ such that for any $\beta \in \mathcal{Z}_\phi$*

$$\phi(x, \alpha) = \psi(\phi(x, \beta), \delta_x(\alpha, \beta)). \quad (341)$$

Theorem 2 (restated). *Let $\phi: X \times \mathcal{Z}_\phi \to X$ be a transformation that is resolved by $\psi: X \times \mathcal{Z}_\psi \to X$. Let $\varepsilon \sim \mathbb{P}_\varepsilon$ be a $\mathcal{Z}_\psi$-valued random variable and suppose that the smoothed classifier $g: X \to \mathcal{Y}$ given by $q(y|x; \varepsilon) = \mathbb{E}(p(y|\psi(x, \varepsilon)))$ predicts $g(x; \varepsilon) = y_A = \arg\max_y q(y|x; \varepsilon)$. Let $\mathcal{S} \subseteq \mathcal{Z}_\phi$ and $\{\alpha_i\}_{i=1}^N \subseteq \mathcal{S}$ be a set of transformation parameters such that for any i, the class probabilities satisfy*

$$q(y_A|\phi(x, \alpha_i); \varepsilon) \geq p_A^{(i)} \geq p_B^{(i)} \geq \max_{y \neq y_A} q(y|\phi(x, \alpha_i); \varepsilon). \quad (342)$$

Then there exists a set $\Delta^* \subseteq \mathcal{Z}_\psi$ with the property that, if for any $\alpha \in \mathcal{S}$, $\exists\alpha_i$ with $\delta_x(\alpha, \alpha_i) \in \Delta^*$, then it is guaranteed that

$$q(y_A|\phi(x, \alpha); \varepsilon) > \max_{y \neq y_A} q(y|\phi(x, \alpha); \varepsilon). \quad (343)$$

PROOF. We prove the theorem by explicitly constructing a region Δ^* with the desired property by applying Theorem 1. For that purpose let $\delta \in \mathcal{Z}_\psi$ and denote by $\gamma_\delta: \mathcal{Z}_\psi \to \mathcal{Z}_\psi$ the resolving function of ψ, i.e.,

$$\psi(\psi(x, \delta), \delta') = \psi(x, \gamma_\delta(\delta')). \quad (344)$$

Let $\mathbb{P}_\gamma$ be the distribution of the random variable $\gamma := \gamma_\delta(\varepsilon)$ with density function f_γ and let

$$\underline{S}_t = \{z \in \mathcal{Z}_\psi: \Lambda(z) < t\}, \quad \overline{S}_t = \{z \in \mathcal{Z}_\psi: \Lambda(z) \leq t\} \quad (345)$$

$$\text{where} \quad \Lambda(z) = \frac{f_\gamma(z)}{f_\varepsilon(z)}. \quad (346)$$

Furthermore, recall the definition of the function $\zeta: \mathbb{R}_{\geq 0} \to [0, 1]$ that is given by $t \mapsto \zeta(t) := \mathbb{P}_\varepsilon(\overline{S}_t)$ with generalized inverse $\zeta^{-1}(p) := \inf\{t \geq 0: \zeta(t) \geq p\}$. For $t \geq 0$ and the function $\xi: [0, 1] \to [0, 1]$ is given by by

$$\xi(p) := \sup\{\mathbb{P}_\gamma(S): \underline{S}_{\zeta^{-1}(p)} \subseteq S \subseteq \overline{S}_\zeta^{-1}(p), \mathbb{P}_\varepsilon(S) \leq p\}. \quad (347)$$

By assumption, for every $i = 1, \ldots, n$, the ε-smoothed classifier g is $(p_A^{(i)}, p_B^{(i)})$-confident at $\phi(x, \alpha_i)$. Identify $\Delta_i \subseteq \mathcal{Z}_\psi$ with the set of perturbations that satisfy the robustness condition (6) in Theorem 1, i.e.,

$$\Delta_i \equiv \{\delta \in \mathcal{Z}_\psi: 1 - \xi(1 - p_B^{(i)}) < \xi(p_A^{(i)})\}. \quad (348)$$

Thus, by Theorem 1, we have that for any $\delta \in \Delta_i$

$$q(y_A|\psi(\phi(x, \alpha_i), \delta); \varepsilon) > \max_{y \neq y_A} q(y|\psi(\phi(x, \alpha_i), \delta); \varepsilon). \quad (349)$$

Finally, note that for the set

$$\Delta^* \equiv \bigcap_{i=1}^N \Delta_i \quad (350)$$

it holds that, if for $\alpha \in \mathcal{S}$ there exists α_i with $\delta_x(\alpha, \alpha_i) \in \Delta^*$, then in particular $\delta_x(\alpha, \alpha_i) \in \Delta_i$ and hence, by Theorem 1 it is guaranteed that

$$q(y_A | \phi(x, \alpha); \varepsilon) = q(y_A | \psi(\phi(x, \alpha_i), \delta_x(\alpha, \alpha_i)); \varepsilon) \qquad (351)$$

$$> \max_{y \neq y_A} q(y | \psi(\phi(x, \alpha_i), \delta_x(\alpha, \alpha_i)); \varepsilon) \qquad (352)$$

$$= \max_{y \neq y_A} q(y | \phi(x, \alpha); \varepsilon) \qquad (353)$$

what concludes the proof. □

Corollary 2 (restated). *Let $\psi(x, \delta) = x + \delta$ and let $\varepsilon \sim \mathcal{N}(0, \sigma^2 \mathbb{1}_d)$. Furthermore, let ϕ be a transformation with parameters in $\mathcal{Z}_\phi \subseteq \mathbb{R}^m$ and let $\mathcal{S} \subseteq \mathcal{Z}_\phi$ and $\{\alpha_i\}_{i=1}^N \subseteq \mathcal{S}$. Let $y_A \in \mathcal{Y}$ and suppose that for any i, the ε-smoothed classifier defined by $q(y | x; \varepsilon) := \mathbb{E}(p(y | x + \varepsilon))$ has class probabilities that satisfy*

$$q(y_A | \phi(x, \alpha_i); \varepsilon) \geq p_A^{(i)} \geq p_B^{(i)} \geq \max_{y \neq y_A} q(y | \phi(x, \alpha_i); \varepsilon). \quad (354)$$

Then it is guaranteed that $\forall \alpha \in \mathcal{S}: y_A = \arg\max_y q(y | \phi(x, \alpha); \varepsilon)$ if the maximum interpolation error

$$M_\mathcal{S} := \max_{\alpha \in \mathcal{S}} \min_{1 \leq i \leq N} \|\phi(x, \alpha) - \phi(x, \alpha_i)\|_2 \qquad (355)$$

satisfies
$$M_\mathcal{S} < R := \frac{\sigma}{2} \min_{1 \leq i \leq N} \left(\Phi^{-1}\left(p_A^{(i)}\right) - \Phi^{-1}\left(p_B^{(i)}\right) \right). \qquad (356)$$

PROOF. Since the resolvable transformation ψ is given by $\psi(x, \delta) = x + \delta$ we can write

$$\phi(x, \alpha) = \phi(x, \alpha_i) + \underbrace{(\phi(x, \alpha) - \phi(x, \alpha_i))}_{=:\delta_x(\alpha, \alpha_i)}. \qquad (357)$$

Furthermore, by assumption $\varepsilon \sim \mathcal{N}(0, \sigma^2 \mathbb{1}_d)$ and $g(\cdot; \varepsilon)$ is $(p_A^{(i)}, p_B^{(i)})$-confident at $\phi(x, \alpha_i)$ for y_A and for all i. Thus, by Corollary 7, if δ satisfies

$$\|\delta\|_2 < R_i := \frac{\sigma}{2} \left(\Phi^{-1}\left(p_A^{(i)}\right) - \Phi^{-1}\left(p_B^{(i)}\right) \right) \qquad (358)$$

then it is guaranteed that $y_A = \arg\max_y q(y | \phi(x, \alpha_i) + \delta; \varepsilon)$. Let $\Delta_i := B_{R_i}(0)$ and notice that $R \equiv \min_i R_i$ and thus

$$\bigcap_{i=1}^N B_{R_i}(0) = B_R(0) = \Delta^*. \qquad (359)$$

To see that Δ^* has the desired property, consider

$$\forall \alpha \in \mathcal{S} \ \exists \alpha_i: \ \delta_x(\alpha, \alpha_i) \in \Delta^* \qquad (360)$$

$$\iff \forall \alpha \in \mathcal{S} \ \exists \alpha_i: \ \|\phi(x, \alpha) - \phi(x, \alpha_i)\|_2 < R. \qquad (361)$$

Since $R \leq R_i$ it follows that for $\delta_i = \phi(x, \alpha) - \phi(x, \alpha_i)$ it is guaranteed that

$$y_A = \arg\max_y q(y | \phi(x, \alpha_i) + \delta_i; \varepsilon) \qquad (362)$$

$$= \arg\max_y q(y | \phi(x, \alpha); \varepsilon). \qquad (363)$$

Thus, the set Δ^* has the desired property. In particular, since

$$\forall \alpha \in \mathcal{S} \ \exists \alpha_i: \ \|\phi(x, \alpha) - \phi(x, \alpha_i)\|_2 < R \qquad (364)$$

$$\iff \max_{\alpha \in \mathcal{S}} \min_{1 \leq i \leq N} \|\phi(x, \alpha) - \phi(x, \alpha_i)\|_2 < R \qquad (365)$$

the statement follows. □

Corollary 5 (restated). *Let $\psi_B(x, \delta, b) = x + \delta + b \cdot \mathbb{1}_d$ and let $\varepsilon \sim \mathcal{N}(0, \sigma^2 \mathbb{1}_d)$, $\varepsilon_b \sim \mathcal{N}(0, \sigma_b^2)$. Furthermore, let ϕ be a transformation with parameters in $\mathcal{Z}_\phi \subseteq \mathbb{R}^m$ and let $\mathcal{S} \subseteq \mathcal{Z}_\phi$ and $\{\alpha_i\}_{i=1}^N \subseteq \mathcal{S}$. Let $y_A \in \mathcal{Y}$ and suppose that for any i, the $(\varepsilon, \varepsilon_b)$-smoothed classifier $q(y | x; \varepsilon, \varepsilon_b) := \mathbb{E}(p(y | \psi_B(x, \varepsilon, \varepsilon_b))$ satisfies*

$$q(y_A | x; \varepsilon, \varepsilon_b) \geq p_A^{(i)} > p_B^{(i)} \geq \max_{y \neq y_A} q(y | x; \varepsilon, \varepsilon_b). \qquad (366)$$

for each i. Let

$$R := \frac{\sigma}{2} \min_{1 \leq i \leq N} \left(\Phi^{-1}\left(p_A^{(i)}\right) - \Phi^{-1}\left(p_B^{(i)}\right) \right) \qquad (367)$$

Then, $\forall \alpha \in \mathcal{S}$ and $\forall b \in [-b_0, b_0]$ it is guaranteed that $y_A = \arg\max_y q(y | \phi(x, \alpha) + b \cdot \mathbb{1}_d; \varepsilon, \varepsilon_b)$ as long as

$$R > \sqrt{M_\mathcal{S}^2 + \frac{\sigma^2}{\sigma_b^2} b_0^2}, \qquad (368)$$

where $M_\mathcal{S}$ is defined as in Corollary 2.

PROOF. Since the resolvable transformation ψ_B is given by

$$\psi_B(x, \delta, b) = x + \delta + b \cdot \mathbb{1}_d, \qquad (369)$$

we can write

$$\phi(x, \alpha) + b \cdot \mathbb{1}_d = \phi(x, \alpha_i) + \underbrace{(\phi(x, \alpha) - \phi(x, \alpha_i)) + b \cdot \mathbb{1}_d}_{=:\delta_x((\alpha, b), (\alpha_i, 0))}. \qquad (370)$$

Furthermore, by assumption $\varepsilon \sim \mathcal{N}(0, \sigma^2 \mathbb{1}_d)$, $\varepsilon_b \sim \mathcal{N}(0, \sigma_b^2)$ and $g(\cdot; \varepsilon, \varepsilon_b)$ is $(p_A^{(i)}, p_B^{(i)})$-confident at $\phi(x, \alpha_i)$ for y_A and all i. Thus, by Corollary 7, if δ and b satisfy

$$\sqrt{\frac{\|\delta\|_2^2}{\sigma^2} + \frac{b^2}{\sigma_b^2}} < \frac{1}{2} \left(\Phi^{-1}(p_A^{(i)}) - \Phi^{-1}(p_B^{(i)}) \right), \qquad (371)$$

then it is guaranteed that

$$y_A = \arg\max_y q(y | \phi(x, \alpha_i) + \delta + b \cdot \mathbb{1}_d; \varepsilon, \varepsilon_b). \qquad (372)$$

Let

$$R_i := \frac{\sigma}{2} \left(\Phi^{-1}(p_A^{(i)}) - \Phi^{-1}(p_B^{(i)}) \right) \qquad (373)$$

and note that without loss of generality we can assume that $R_i > \sigma/\sigma_b b_0$, because otherwise the robustness condition is violated. Rearranging terms in (371) leads to the condition

$$\|\delta\|_2 < \sqrt{R_i^2 - \frac{\sigma^2}{\sigma_b^2} b^2} \qquad (374)$$

that can be turned into a sufficient robustness condition holding for any $b \in [-b_0, b_0]$ simultaneously

$$\|\delta\|_2 < \sqrt{R_i^2 - \frac{\sigma^2}{\sigma_b^2} b_0^2} \qquad (375)$$

Note that, without loss of generality For each i let Δ_i be the set defined as

$$\Delta_i := \left\{ \delta + b \cdot \mathbb{1}_d \in \mathbb{R}^d : \|\delta\|_2 < \sqrt{R_i^2 - \frac{\sigma^2}{\sigma_b^2} b_0^2}, \ |b| \leq b_0 \right\} \quad (376)$$

and note that

$$\Delta^* := \bigcap_{i=1}^N \Delta_i \qquad (377)$$

$$= \left\{ \delta + b \cdot \mathbb{1}_d \in \mathbb{R}^d : \|\delta\|_2 < \sqrt{R^2 - \frac{\sigma^2}{\sigma_b^2} b_0^2}, \, |b| \le b_0 \right\} \quad (378)$$

with $R := \min_i R_i$. Clearly, if $\forall \alpha \in \mathcal{S}, \forall b \in [-b_0, b_0] \exists i$ such that

$$\delta_x((\alpha, b), (\alpha_i, 0)) \in \Delta^* \quad (379)$$

then it is guaranteed that

$$y_A = \arg\max_y q(y \mid \phi(x, \alpha_i) + \delta_x((\alpha, b), (\alpha_i, 0))) \quad (380)$$

$$= \arg\max_y q(y \mid \phi(x, \alpha) + b \cdot \mathbb{1}_d). \quad (381)$$

We can thus reformulate the robustness condition as

$$\forall \alpha \in \mathcal{S}, \forall b \in [-b_0, b_0] \exists i \text{ s.t. } \delta_x((\alpha, b), (\alpha_i, 0)) \in \Delta^* \quad (382)$$

$$\iff$$

$$\forall \alpha \in \mathcal{S}, \forall b \in [-b_0, b_0] \exists i \text{ s.t.}$$

$$\|\phi(x, \alpha) - \phi(x, \alpha_i)\|_2 < \sqrt{R^2 - \frac{\sigma^2}{\sigma_b^2} b_0^2} \quad (383)$$

$$\iff$$

$$\max_{\alpha \in \mathcal{S}} \min_{1 \le i \le N} \|\phi(x, \alpha) - \phi(x, \alpha_i)\|_2 < \sqrt{R^2 - \frac{\sigma^2}{\sigma_b^2} b_0^2} \quad (384)$$

that, written in terms of the maximum ℓ_2 interpolation error $M_{\mathcal{S}}$, is equivalent to

$$R > \sqrt{M_{\mathcal{S}}^2 + \frac{\sigma^2}{\sigma_b^2} b_0^2} \quad (385)$$

what concludes the proof. □

Corollary 6 (restated). *Under the same setting as in Corollary 5, for $\forall \alpha \in \mathcal{S}, \forall b \in [-b_0, b_0]$ and $\forall \delta \in \mathbb{R}^d$ such that $\|\delta\|_2 \le r$, it is guaranteed that $y_A = \arg\max_k q(y \mid \phi(x, \alpha) + b \cdot \mathbb{1}_d + \delta; \varepsilon, \varepsilon_d)$ as long as*

$$R > \sqrt{(M_{\mathcal{S}} + r)^2 + \frac{\sigma^2}{\sigma_b^2} b_0^2}, \quad (386)$$

where $M_{\mathcal{S}}$ is defined as in Corollary 2.

PROOF. Note that we can write the transformed input as

$$\phi(x, \alpha) + b \cdot \mathbb{1}_d + \delta$$
$$= \phi(x, \alpha_i) + \underbrace{(\phi(x, \alpha) - \phi(x, \alpha_i) + \delta) + b \cdot \mathbb{1}_d}_{=: \delta_x((\alpha, b, \delta), (\alpha_i, 0, 0))}. \quad (387)$$

Since we use the same smoothing protocol as in Corollary 5, the general proof idea is similar to Corollary 5 — we use the same resolvable transformation ψ_B and define the same set Δ_i, namely

$$\Delta_i := \left\{ \delta' + b \cdot \mathbb{1}_d + \delta \in \mathbb{R}^d : \right.$$
$$\left. \|\delta' + \delta\|_2 < \sqrt{R_i^2 - \frac{\sigma^2}{\sigma_b^2} b_0^2}, \, |b| \le b_0, \, \|\delta\|_2 \le r \right\}. \quad (388)$$

and set

$$\Delta^* := \bigcap_{i=1}^{N} \Delta_i \quad (389)$$

$$= \left\{ \delta' + b \cdot \mathbb{1}_d + \delta \in \mathbb{R}^d : \right.$$
$$\left. \|\delta' + \delta\|_2 < \sqrt{R^2 - \frac{\sigma^2}{\sigma_b^2} b_0^2}, \, |b| \le b_0, \, \|\delta\|_2 \le r \right\} \quad (390)$$

with $R := \min_i R_i$. Clearly, if $\forall \alpha \in \mathcal{S}, \forall b \in [-b_0, b_0], \|\delta\|_2 \le r \exists i$ such that

$$\delta_x((\alpha, b, \delta), (\alpha_i, 0)) \in \Delta^* \quad (391)$$

then it is guaranteed that

$$y_A = \arg\max_y q(y \mid \phi(x, \alpha_i) + \delta_x((\alpha, b), (\alpha_i, 0))) \quad (392)$$

$$= \arg\max_y q(y \mid \phi(x, \alpha) + b \cdot \mathbb{1}_d). \quad (393)$$

We can thus reformulate the robustness condition as

$$\forall \alpha \in \mathcal{S}, \forall b \in [-b_0, b_0], \|\delta\|_2 \le r \exists i$$
$$\text{s.t. } \delta_x((\alpha, b, \delta), (\alpha_i, 0, 0)) \in \Delta^* \quad (394)$$

$$\iff$$

$$\forall \alpha \in \mathcal{S}, \forall b \in [-b_0, b_0], \|\delta\|_2 \le r \exists i \text{ s.t.}$$

$$\|\phi(x, \alpha) - \phi(x, \alpha_i) + \delta\|_2 < \sqrt{R^2 - \frac{\sigma^2}{\sigma_b^2} b_0^2} \quad (395)$$

$$\iff$$

$$\max_{\alpha \in \mathcal{S}} \min_{1 \le i \le N} \|\phi(x, \alpha) - \phi(x, \alpha_i) + \delta\|_2 < \sqrt{R^2 - \frac{\sigma^2}{\sigma_b^2} b_0^2}. \quad (396)$$

Note that by the triangle inequality have

$$\max_{\alpha \in \mathcal{S}} \min_{1 \le i \le N} \|\phi(x, \alpha) - \phi(x, \alpha_i) + \delta\|_2 \le M_{\mathcal{S}} + \|\delta\|_2 \quad (397)$$

$$\le M_{\mathcal{S}} + r \quad (398)$$

and thus, robustness is implied by

$$R > \sqrt{(M_{\mathcal{S}} + r)^2 + \frac{\sigma^2}{\sigma_b^2} b_0^2} \quad (399)$$

what concludes the proof.

□

H TRANSFORMATION DETAILS

In this section, we provide detailed definitions of rotation and scaling transformation.

H.1 Bilinear Interpolation

Let $\Omega_K := \{0, \ldots, K-1\}$ and $\Omega := [0, W-1] \times [0, H-1]$. We define bilinear interpolation to be the map $Q : \mathbb{R}^{K \times W \times H} \to L^2(\Omega_K \times \mathbb{R}^2, \mathbb{R})$, $x \mapsto Q(x) =: Q_x$ where Q_x is given by

$$(k, i, j) \mapsto Q_x(k, i, j) := \begin{cases} 0 & (i, j) \notin \Omega \\ x_{k,i,j} & (i, j) \in \Omega \cap \mathbb{N}^2 \\ \tilde{x}_{k,i,j} & (i, j) \in \Omega \setminus \mathbb{N}^2. \end{cases} \quad (400)$$

and where

$$\tilde{x}_{k,i,j} := (1 - (i - \lfloor i \rfloor)) \cdot \left((1 - (j - \lfloor j \rfloor)) \cdot x_{k, \lfloor i \rfloor, \lfloor j \rfloor} \right.$$
$$+ (j - \lfloor j \rfloor) \cdot x_{k, \lfloor i \rfloor, \lfloor j \rfloor + 1} \Big)$$
$$+ (i - \lfloor i \rfloor) \cdot \left((1 - (j - \lfloor j \rfloor)) \cdot x_{k, \lfloor i \rfloor + 1, \lfloor j \rfloor} \right.$$
$$+ (j - \lfloor j \rfloor) \cdot x_{k, \lfloor i \rfloor + 1, \lfloor j \rfloor + 1} \Big). \quad (401)$$

H.2 Rotation

The rotation transformation is denoted as $\phi_R \colon \mathbb{R}^{K \times W \times H} \times \mathbb{R} \to \mathbb{R}^{K \times W \times H}$ and acts on an image in three steps that we will highlight in greater detail. First, it rotates the image by α degrees counterclockwise. After rotation, pixel values are determined using bilinear interpolation (400). Finally, we apply black padding to all pixels (i, j) whose ℓ_2-distance to the center pixel is larger than half of the length of the shorter side, and denote this operation by P. Let c_W and c_H be the center pixels

$$c_W := \frac{W - 1}{2}, \qquad c_H := \frac{H - 1}{2}. \tag{402}$$

and

$$d_{i,j} = \sqrt{(i - c_W)^2 + (j - c_H)^2},$$
$$g_{i,j} = \operatorname{arctan2}(j - c_H, i - c_W). \tag{403}$$

We write $\tilde{\phi}_R$ for the rotation transformation before black padding and decompose ϕ_R as $\phi_R = P \circ \tilde{\phi}_R$, where $\tilde{\phi}_R \colon \mathbb{R}^{K \times W \times H} \times \mathbb{R} \to \mathbb{R}^{K \times W \times H}$ is defined by

$$\tilde{\phi}_R(x, \alpha)_{k,i,j} := Q_x(k, c_W + d_{i,j} \cos(g_{i,j} - \alpha),$$
$$c_H + d_{i,j} \sin(g_{i,j} - \alpha)) \tag{404}$$

and $P \colon \mathbb{R}^{K \times W \times H} \to \mathbb{R}^{K \times W \times H}$ by

$$f \mapsto P(f)_{k,i,j} = \begin{cases} f(k, i, j) & d_{i,j} < \min\{c_W, c_H\} \\ 0 & \text{otherwise} \end{cases}. \tag{405}$$

The rotation transformation in practice may use different padding mechanisms. For example, the rotation in the physical world may fill in boundary pixels with real elements captured by the camera. We remark that our **TSS** against the transformation ϕ_R implies the defense against rotation with *any other* padding mechanisms, because we first apply black-padding P to any such rotated input and then feed into **TSS** models so that **TSS** models always receive black-padded inputs.

H.3 Scaling

The scaling transformation is denoted as $\phi_S \colon \mathbb{R}^{K \times W \times H} \times \mathbb{R} \to \mathbb{R}^{K \times W \times H}$. Similar as for rotations, ϕ_S acts on an image in three steps. First, it stretches height and width by a fixed ratio $\alpha \in \mathbb{R}$. Second, we determine missing pixel values with bilinear interpolation. Finally, we apply black padding to regions with missing pixel values if the image is scaled by a factor smaller than 1. Let c_W and c_H be the center pixels

$$c_W := \frac{W - 1}{2}, \qquad c_H := \frac{H - 1}{2}. \tag{406}$$

We notice that black padding is naturally applied during bilinear interpolation in cases where the scaling factor is smaller than 1 (that is, when we make images smaller). We can thus write the scaling operation as $\phi_S \colon \mathbb{R}^{K \times W \times H} \times \mathbb{R}_{>0} \to \mathbb{R}^{K \times W \times H}$, $(x, \alpha) \mapsto \phi(x, \alpha)$ where

$$\phi_S(x, \alpha)_{k,i,j} := Q_x\left(k, c_W + \frac{i - c_W}{\alpha}, c_H + \frac{j - c_H}{\alpha}\right). \tag{407}$$

When the scaling transformation in practice uses different padding mechanisms, we can simply apply black padding to the outer pixels during preprocessing. For example, if we know the semantic attacker could choose 0.7 as the smallest scaling ratio, we can apply black padding to all pixels that are out of canvas after 0.7 scaling. Therefore, we overwrite all different padding mechanisms and ensure the generalizability. As a trade-off, the classifier has a narrower reception field that affects the clean accuracy.

I PROOFS FOR INTERPOLATION BOUND COMPUTATION

In this section we state the proofs for the theoretical results governing our approach to certifying rotations and scaling transformations using randomized smoothing. We first define the maximum ℓ_2 interpolation error. First, let us recall the following definitions from the main part of this paper.

DEFINITION 9 (ℓ_2 INTERPOLATION ERROR). *Let* $x \in \mathcal{X}$, $\phi \colon \mathcal{X} \times \mathcal{Z} \to \mathcal{X}$ *a transformation,* $\mathcal{S} = [a, b]$, $N \in \mathbb{N}$ *and suppose* $\{\alpha_i\}_{i=1}^N \subseteq \mathcal{S}$. *The maximum* ℓ_2 *interpolation error is defined as*

$$M_{\mathcal{S}} := \max_{a \le \alpha \le b} \min_{1 \le i \le N} \|\phi(x, \alpha) - \phi(x, \alpha_i)\|_2. \tag{408}$$

Definition 4 (restated). *For pixels* $(i, j) \in \Omega$, *we define the grid pixel generator* G_{ij} *as*

$$G_{ij} := \{(i, j), (i + 1, j), (i, j + 1), (i + 1, j + 1)\}. \tag{409}$$

Definition 5 (restated). *We define the operator that extracts the channel-wise maximum pixel wise on a grid* $S \subseteq \Omega$ *as the map* $\overline{m} \colon \mathbb{R}^{K \times W \times H} \times \{0, \ldots, K - 1\} \times 2^{\Omega} \to \mathbb{R}$ *with*

$$\overline{m}(x, k, S) := \max_{(i,j) \in S}\left(\max_{(r,s) \in G_{ij}} x_{k,r,s}\right) \tag{410}$$

Definition 6 (restated). *We define the operator that extracts the channel-wise maximum change in color on a grid* $S \subseteq \Omega$ *as the map* $m_\Delta \colon \mathbb{R}^{K \times W \times H} \times \{0, \ldots, K - 1\} \times 2^{\Omega} \to \mathbb{R}$ *with*

$$m_\Delta(x, k, S) := \max_{(i,j) \in S}\left(\max_{(r,s) \in G_{ij}} x_{k,r,s} - \min_{(r,s) \in G_{ij}} x_{k,r,s}\right) \tag{411}$$

The following auxiliary lemma is used for both rotation and scaling:

LEMMA 9. *Let* $x \in \mathbb{R}^{K \times W \times H}$, $-\infty < t_1 < t_2 < \infty$ *and suppose* $\rho \colon [t_1, t_2] \to [0, W - 1] \times [0, H - 1]$ *is a curve of class* C^1. *Let*

$$\psi_k \colon [t_1, t_2] \to \mathbb{R}, \quad \psi_k(t) := Q_x(k, \rho_1(t), \rho_2(t)) \tag{412}$$

where $k \in \Omega_K$ *and* Q_x *denotes bilinear interpolation. Then* ψ_k *is* L_k-*Lipschitz continuous with constant*

$$L_k = \max_{t \in [t_1, t_2]}\left(\sqrt{2}\,\|\dot{\rho}(t)\|_2 \cdot m_\Delta(x, k, \lfloor \rho(t) \rfloor)\right) \tag{413}$$

PROOF. Note that the function $t \mapsto \lfloor \rho(t) \rfloor$ is piecewise constant and let $t_1 =: u_1 < u_2 < \ldots < u_{N_0} := t_2$ such that $\lfloor \rho(t) \rfloor$ is constant on $[u_i, u_{i+1}]$ for all $1 \le i \le N_0 - 1$ and $\dot{\bigcup}_{i=1}^{N_0} [u_i, u_{i+1}] = [t_1, t_2]$. We notice that ψ_k is a continuous real-valued function since it is the composition of the continuous Q_x and C^1-curve ρ. L_k-Lipschitz continuity on $[t_1, t_2]$ thus follows if we show that ψ_k is L_k-Lipschitz on each interval in the partition. For that purpose, let $1 \le i \le N_0$ be arbitrary and fix some $t \in [u_i, u_{i+1}]$. Let $(w, h) := \lfloor \rho(t) \rfloor$ and $\gamma(t) := \rho(t) - \lfloor \rho(t) \rfloor$ and notice that $\gamma(t) \in [0, 1]^2$. Let

$$V_1 := x_{k,w,h}, V_2 := x_{k,w,h+1}, \tag{414}$$
$$V_3 := x_{k,w+1,h}, V_4 := x_{k,w+1,h+1}, \tag{415}$$

Then, for any $u \in [u_i, u_{i+1})$

$$\psi_k(u) = Q_x(k, \rho_1(u), \rho_2(u)) \tag{416}$$

$$\begin{aligned} = (1 - \gamma_1(u)) \cdot ((1 - \gamma_2(u)) \cdot V_1 + \gamma_2(u) \cdot V_2) \\ + \gamma_1(u) \cdot ((1 - \gamma_2(u) \cdot V_3 + \gamma_2(u) \cdot V_4). \end{aligned} \tag{417}$$

Let $m_\Delta := m_\Delta(x, k \lfloor \rho(t) \rfloor)$ and notice that by definition

$$m_\Delta = \max_i V_i - \min_i V_i \tag{418}$$

and in particular

$$\left| V_i - V_j \right| \le m_\Delta \quad \forall i, j. \tag{419}$$

Since V_i is constant for each i and γ is differentiable, ψ_k is differentiable on $[u_i, u_{i+1})$ and hence

$$\dot{\psi}_k(u) = (\dot{\gamma}_1(u)\gamma_2(u) + \gamma_1(u)\dot{\gamma}_2(u))(V_1 - V_2 - V_3 + V_4) \tag{420}$$

$$+ \dot{\gamma}_1(u)(V_3 - V_1) + \dot{\gamma}_2(u)(V_2 - V_1). \tag{421}$$

Note that the derivative $\dot{\psi}_k$ is linear in γ_1 and γ_2 and hence its extreme values are bounded when evaluated at extreme values of γ, that is $(\gamma_1, \gamma_2) \in \{0, 1\}^2$. We treat each case separately:

- $\gamma_1 = \gamma_2 = 0$. Then,

$$\left| \dot{\psi}_k \right| \le |\dot{\gamma}_1(V_3 - V_1) + \dot{\gamma}_2(V_2 - V_1)| \tag{422}$$

$$\le |\dot{\gamma}_1| \cdot |V_3 - V_1| + |\dot{\gamma}_2| \cdot |V_2 - V_1| \tag{423}$$

$$\le m_\Delta(|\dot{\gamma}_1| + |\dot{\gamma}_2|) \tag{424}$$

- $\gamma_1 = \gamma_2 = 1$. Then,

$$\left| \dot{\psi}_k \right| \le |\dot{\gamma}_1(V_4 - V_2) + \dot{\gamma}_2(V_4 - V_3)| \tag{425}$$

$$\le |\dot{\gamma}_1| \cdot |V_4 - V_2| + |\dot{\gamma}_2| \cdot |V_4 - V_3| \tag{426}$$

$$\le m_\Delta(|\dot{\gamma}_1| + |\dot{\gamma}_2|) \tag{427}$$

- $\gamma_1 = 0, \gamma_2 = 1$. Then,

$$\left| \dot{\psi}_k \right| \le |\dot{\gamma}_1(V_4 - V_2) + \dot{\gamma}_2(V_2 - V_1)| \tag{428}$$

$$\le |\dot{\gamma}_1| \cdot |V_4 - V_2| + |\dot{\gamma}_2| \cdot |V_2 - V_1| \tag{429}$$

$$\le m_\Delta(|\dot{\gamma}_1| + |\dot{\gamma}_2|) \tag{430}$$

- $\gamma_1 = 1, \gamma_2 = 0$. Then,

$$\left| \dot{\psi}_k \right| \le |\dot{\gamma}_1(V_3 - V_1) + \dot{\gamma}_2(V_4 - V_3)| \tag{431}$$

$$\le |\dot{\gamma}_1| \cdot |V_3 - V_1| + |\dot{\gamma}_2| \cdot |V_4 - V_3| \tag{432}$$

$$\le m_\Delta(|\dot{\gamma}_1| + |\dot{\gamma}_2|) \tag{433}$$

Hence, for any $u \in [u_i, u_{i+1})$, the modulus of the derivative is bounded by $m_\Delta(|\dot{\gamma}_1| + |\dot{\gamma}_2|)$. We can further bound this by observing the following connection between ℓ_1 and ℓ_2 distance

$$\forall x \in \mathbb{R}^n : \quad \|x\|_1 = |\langle |x|, \mathbf{1} \rangle| \le \|x\|_2 \|\mathbf{1}\|_2 = \sqrt{n} \|x\|_2 \tag{434}$$

and hence $\forall u \in [u_i, u_{i+1})$

$$|\dot{\psi}_k(u)| \le m_\Delta \|\dot{\gamma}(u)\|_1 \tag{435}$$

$$\le m_\Delta \sqrt{2} \|\dot{\gamma}(u)\|_2 \tag{436}$$

$$= m_\Delta \sqrt{2} \|\dot{\rho}(u)\|_2 . \tag{437}$$

Since ψ_k is differentiable on $[u_i, u_{i+1})$, its Lipschitz constant is bounded by the maximum absolute value of its derivative. Hence

$$\begin{aligned} \max_{u \in [u_i, u_{i+1})} m_\Delta \sqrt{2} \|\dot{\rho}(u)\|_2 \\ = \max_{u \in [u_i, u_{i+1})} m_\Delta(x, k, \lfloor \rho(u) \rfloor) \sqrt{2} \|\dot{\rho}(u)\|_2 \end{aligned} \tag{438}$$

$$\le \max_{u \in [t_1, t_2)} m_\Delta(x, k, \lfloor \rho(u) \rfloor) \sqrt{2} \|\dot{\rho}(u)\|_2 = L_k \tag{439}$$

is a Lipschitz constant for ψ_k on $[u_i, u_{i+1})$. Note that L_k does not depend on i. Furthermore, i was chosen arbitrarily and hence L_k is a Lipschitz constant for ψ_k on $[t_1, t_2)$ and due to continuity on $[t_1, t_2]$, concluding the proof. □

I.1 Rotation

Lemma 4 (restated). *Let $x \in \mathbb{R}^{K \times W \times H}$ be a K-channel image and let $\phi_R = P \circ I \circ \tilde{\phi}_R$ be the rotation transformation. Then, a global Lipschitz constant L for the functions $\{g_i\}_{i=1}^N$ is given by*

$$L_r = \max_{1 \le i \le N-1} \sum_{k=0}^{K-1} \sum_{r,s \in V} 2d_{r,s} \cdot m_\Delta(x, k, \mathcal{P}_{r,s}^{(i)}) \cdot \overline{m}(x, k, \mathcal{P}_{r,s}^{(i)}) \tag{440}$$

where $V = \left\{ (r, s) \in \mathbb{N}^2 | d_{r,s} < \frac{1}{2}(\min\{W, H\} - 1) \right\}$. The set $\mathcal{P}_{r,s}^{(i)}$ is given by all integer grid pixels that are covered by the trajectory of source pixels of (r, s) when rotating from angle α_i to α_{i+1}.

PROOF. Recall that ϕ_R acts on images $x \in \mathbb{R}^{K \times W \times H}$ and that g_i is defined as

$$\begin{aligned} g_i(\alpha) &= \|\phi_R(x, \alpha) - \phi_R(x, \alpha_i)\|_2^2 \\ &= \sum_{k=0}^{K-1} \sum_{r=0}^{W-1} \sum_{s=0}^{H-1} \left(\phi_R(x, \alpha)_{k,r,s} - \phi_R(x, \alpha_i)_{k,r,s} \right)^2 \end{aligned} \tag{441}$$

Let c_W and c_H denote the center pixels

$$c_W := \frac{W-1}{2}, \qquad c_H := \frac{H-1}{2}. \tag{442}$$

and recall the following quantities from the definition of ϕ_R (Appendix H.2):

$$\begin{aligned} d_{r,s} &= \sqrt{(r - c_W)^2 + (s - c_H)^2}, \\ g_{r,s} &= \arctan 2 (s - c_H, r - c_W) \end{aligned} \tag{443}$$

Note that

$$d_{r,s} \ge \min\{c_W, c_H\} \implies \phi_R(x, \alpha)_{k,r,s} = 0. \tag{444}$$

We thus only need to consider pixels that lie inside the centered disk. We call the collection of such pixels *valid* pixels, denoted by V:

$$V := \left\{ (r, s) \in \mathbb{N}^2 \mid d_{r,s} < \min\{c_W, c_H\} \right\}. \tag{445}$$

Let $f_1^{r,s} : \mathbb{R} \to \mathbb{R}$ and $f_2^{r,s} : \mathbb{R} \to \mathbb{R}$ be functions defined as

$$\begin{aligned} f_1^{r,s}(\alpha) &= c_W + d_{r,s} \cos(g_{r,s} - \alpha), \\ f_2^{r,s}(\alpha) &= c_H + d_{r,s} \sin(g_{r,s} - \alpha). \end{aligned} \tag{446}$$

Then for any valid pixel $(r, s) \in V$, the value of the rotated image $\phi_R(x, \alpha)$ is given by

$$\phi_R(x, \alpha)_{k,r,s} = Q_x(k, f_1^{r,s}(\alpha), f_2^{r,s}(\alpha)) \tag{447}$$

where Q_x denotes bilinear interpolation. We define the shorthand

$$g_i^{k,r,s}(\alpha) := \left(\phi_R(x, \alpha)_{k,r,s} - \phi_R(x, \alpha_i)_{k,r,s} \right)^2 \tag{448}$$

and denote by $L_i^{k,r,s}$ and $L_{i+1}^{k,r,s}$ the Lipschitz constants of $g_i^{k,r,s}$ and $g_{i+1}^{k,r,s}$ on $[\alpha_i, \alpha_{i+1}]$. We can write (441) as

$$g_i(\alpha) = \sum_{k=0}^{K-1} \sum_{(r,s) \in V} g_i^{k,r,s}(\alpha),$$
$$g_{i+1}(\alpha) = \sum_{k=0}^{K-1} \sum_{(r,s) \in V} g_{i+1}^{k,r,s}(\alpha) \tag{449}$$

and note that Lipschitz constants of g_i and g_{i+1} on $[\alpha_i, \alpha_{i+1}]$ are given by

$$\max_{c,d \in [\alpha_i, \alpha_{i+1}]} \frac{|g_i(c) - g_i(d)|}{|c-d|} \le \left(\sum_{k=0}^{K-1} \sum_{(r,s) \in V} L_i^{k,r,s} \right) =: L_i \tag{450}$$

$$\max_{c,d \in [\alpha_i, \alpha_{i+1}]} \frac{|g_{i+1}(c) - g_{i+1}(d)|}{|c-d|} \le \left(\sum_{k=0}^{K-1} \sum_{(r,s) \in V} L_{i+1}^{k,r,s} \right) =: L_{i+1} \tag{451}$$

We can hence determine L according to equation (26) as

$$L = \max_i \{ \max \{L_i, L_{i+1}\} \}. \tag{452}$$

Without loss of generality, consider $L_i^{k,r,s}$ and note that

$$\max_{c,d \in [\alpha_i, \alpha_{i+1}]} \left| \frac{g_i^{k,r,s}(c) - g_i^{k,r,s}(d)}{c-d} \right| \tag{453}$$

$$= \max_{c,d \in [\alpha_i, \alpha_{i+1}]} \left| \frac{\phi_R(x,c)_{k,r,s} - \phi_R(x,d)_{k,r,s}}{c-d} \right| \tag{454}$$
$$\cdot \left| \phi_R(x,c)_{k,r,s} + \phi_R(x,d)_{k,r,s} - 2\phi_R(x,\alpha_i)_{k,r,s} \right|$$

$$\le \max_{c,d \in [\alpha_i, \alpha_{i+1}]} \underbrace{\left| \frac{\phi_R(x,c)_{k,r,s} - \phi_R(x,d)_{k,r,s}}{c-d} \right|}_{(I)}$$

$$\cdot 2 \max_{\theta \in [\alpha_i, \alpha_{i+1}]} \underbrace{\left| \phi_R(x,\theta)_{k,r,s} - \phi_R(x,\alpha_i)_{k,r,s} \right|}_{(II)}. \tag{455}$$

To compute a Lipschitz constant for $g_i^{k,r,s}$ on the interval $[\alpha_i, \alpha_{i+1}]$ we thus only need to compute a Lipschitz constant for $\phi_R(x, \cdot)$ on $[\alpha_i, \alpha_{i+1}]$ and an upper bound on (II). For that purpose, note that ϕ_R takes only positive values and consider

$$(II) \le \max_{\theta \in [\alpha_i, \alpha_{i+1}]} \{ \phi_R(x,\theta)_{k,r,s}, \phi_R(x,\alpha_i)_{k,r,s} \} \tag{456}$$

$$= \max_{\theta \in [\alpha_i, \alpha_{i+1}]} \phi_R(x,\theta)_{k,r,s} \tag{457}$$

Notice that now both $L_i^{k,r,s}$ and $L_{i+1}^{k,r,s}$ share the same upper bound. Recall (447), i.e.,

$$\phi_R(x,\theta)_{k,r,s} = Q_x(k, f_1^{r,s}(\theta), f_2^{r,s}(\theta)). \tag{458}$$

Now, we upper bound (456) by finding all integer grid pixels that are covered by the trajectory $(f_1^{r,s}(\theta), f_2^{r,s}(\theta))$. Specifically, let

$$\mathcal{P}_{r,s}^{(i)} := \bigcup_{\theta \in [\alpha_i, \alpha_{i+1}]} \left(\lfloor f_1^{r,s}(\theta) \rfloor, \lfloor f_2^{r,s}(\theta) \rfloor \right). \tag{459}$$

Since ϕ_R is interpolated from integer pixels, we can consider the maximum over $\mathcal{P}_{r,s}^{(i)}$ in order to upper bound (456):

$$\max_{\theta \in [\alpha_i, \alpha_{i+1}]} \phi_R(x,\theta)_{k,r,s} =$$

$$\max_{\theta \in [\alpha_i, \alpha_{i+1}]} Q_x(k, f_1^{r,s}(\theta), f_2^{r,s}(\theta)) \tag{460}$$

$$\le \max_{(i,j) \in \mathcal{P}_{r,s}} \max \{ x(k,i,j), x(k,i+1,j), \tag{461}$$
$$x(k,i,j+1), x(k,i+1,j+1) \}$$

$$= \bar{m}(x, k, \mathcal{P}_{r,s}^{(i)}). \tag{462}$$

We now have to find an upper bound of (I), that is, a Lipschitz constant of $\phi_R(x, \cdot)_{k,r,s}$ on the interval $[\alpha_i, \alpha_{i+1}]$. For that purpose, consider the following. Note that the curve $\rho \colon [\alpha_i, \alpha_{i+1}] \to \mathbb{R}^2$, $\rho(t) := (f_1^{r,s}(t), f_2^{r,s}(t))$ is of class C^1 and

$$\frac{df_1^{r,s}(t)}{dt} = \frac{d}{dt} \left(c_W + d_{r,s} \cos(g_{r,s} - t) \right) \tag{463}$$
$$= d_{r,s} \sin(g_{r,s} - t) \tag{464}$$

$$\frac{df_2^{r,s}(t)}{dt} = \frac{d}{dt} \left(c_H + d_{r,s} \sin(g_{r,s} - t) \right) \tag{465}$$
$$= -d_{r,s} \cos(g_{r,s} - t) \tag{466}$$

and hence

$$\|\dot{\rho}(t)\|_2 = \sqrt{\left(\frac{df_1^{r,s}(t)}{dt} \right)^2 + \left(\frac{df_2^{r,s}(t)}{dt} \right)^2} = \sqrt{2} \, d_{r,s}. \tag{467}$$

By Lemma 9 a Lipschitz constant for the function $\phi_R(x, \cdot)_{k,r,s}$ is thus given by

$$\max_{c,d \in [\alpha_i, \alpha_{i+1}]} \left| \frac{\phi_R(x,c)_{k,r,s} - \phi_R(x,d)_{k,r,s}}{c-d} \right|$$

$$\le 2 \, d_{r,s} \cdot m_\Delta(x, k, \mathcal{P}_{r,s}^{(i)}). \tag{468}$$

We can thus upper bound (I) and (II) in (455) yielding a Lipschitz constant for $g_i^{k,r,s}$ and $g_{i+1}^{k,r,s}$ on $[\alpha_i, \alpha_{i+1}]$

$$\max_{c,d \in [\alpha_i, \alpha_{i+1}]} \left| \frac{g_i^{k,r,s}(c) - g_i^{k,r,s}(d)}{c-d} \right|$$

$$\le 2 \, d_{r,s} \cdot m_\Delta(x, k, \mathcal{P}_{r,s}^{(i)}) \cdot \bar{m}(x, k, \mathcal{P}_{r,s}^{(i)}) \tag{469}$$

$$= L_i^{k,r,s} (= L_{i+1}^{k,r,s}). \tag{470}$$

Finally, we can compute L_r as

$$L = \max_{1 \le i \le N-1} \sum_{k=0}^{K-1} \sum_{(r,s) \in V} L_i^{k,r,s}$$

$$= \max_{1 \le i \le N-1} \sum_{k=0}^{K-1} \sum_{r,s \in V} 2 d_{r,s} \cdot m_\Delta(x, k, \mathcal{P}_{r,s}^{(i)}) \cdot \overline{m}(x, k, \mathcal{P}_{r,s}^{(i)}))$$

$$\tag{471}$$

what concludes the proof. □

I.2 Scaling

Lemma 5 (restated). *Let $x \in \mathbb{R}^{K \times W \times H}$ be a K-channel image and let ϕ_S be the scaling transformation. Then, a global Lipschitz constant L for the functions $\{g_i\}_{i=1}^N$ is given by*

$$L_s = \max_{1 \leq i \leq N-1} \sum_{k=0}^{K-1} \sum_{r,s \in \Omega \cap \mathbb{N}^2} \frac{\sqrt{2}d_{r,s}}{a^2} \cdot m_\Delta(x, k, \mathcal{P}_{r,s}^{(i)}) \cdot \overline{m}(x, k, \mathcal{P}_{r,s}^{(i)}) \tag{472}$$

where $\Omega = [0, W-1] \times [0, H-1]$ and a is the lower boundary value in $\mathcal{S} = [a, b]$. The set $\mathcal{P}_{r,s}^{(i)}$ is given by all integer grid pixels that are covered by the trajectory of source pixels of (r, s) when scaling with factors from α_{i+1} to α_i.

Proof. Recall the Definition of the Scaling transformation ϕ_S given by $\phi_S \colon \mathbb{R}^{K \times W \times H} \times \mathbb{R} \to \mathbb{R}^{K \times W \times H}$, where

$$\phi_S(x, \alpha)_{k,r,s} := Q_x\left(k, c_W + \frac{r - c_W}{s}, c_H + \frac{s - c_H}{s}\right). \tag{473}$$

Recall that the set Ω is given by $\Omega = [0, W-1] \times [0, H-1] = \{1, \ldots, K\}$ and let

$$\Omega_{\mathbb{N}} := \Omega \cap \mathbb{N}^2 \tag{474}$$

be the set of integers in Ω. Let $f_1^r \colon [a, b] \to \mathbb{R}$ and $f_2^{r,s} \colon [a, b] \to \mathbb{R}$ be functions defined as

$$\begin{aligned} f_1^r(\alpha) &:= c_W + \frac{r - c_W}{\alpha}, \\ f_2^s(\alpha) &:= c_H + \frac{s - c_H}{\alpha}. \end{aligned} \tag{475}$$

Then, the value of the scaled image $\phi_S(x, \alpha)$ is given by

$$\phi_S(x, \alpha)_{k,r,s} = Q_x(k, f_1^r(\alpha), f_2^s(\alpha)) \tag{476}$$

where Q_x denotes bilinear interpolation. Let

$$\psi_k \colon [a, b] \to \mathbb{R}, \ \alpha \mapsto Q_x(k, f_1^r(\alpha), f_2^s(\alpha)). \tag{477}$$

We notice that, in contrast to rotations, ψ_k is *not* continuous at every $\alpha \in \mathbb{R}_{>0}$. Namely, when considering scaling factors in $(0, 1)$, bilinear interpolation applies black padding to some $(r, s) \in \Omega$ resulting in discontinuities of ψ_k. To see this, consider the following. The interval $[\alpha_{i+1}, \alpha_i]$ contains a discontinuity of ψ_k, if

$$\begin{cases} \alpha_{i+1} < \dfrac{r - c_W}{c_W} < \alpha_i, & r > c_W, \\ \alpha_{i+1} < \dfrac{c_W - r}{c_W} < \alpha_i, & r < c_W, \end{cases} \tag{478}$$

because then $\exists \alpha_0 \in [\alpha_{i+1}, \alpha_i]$ such that $f_1^r(\alpha_0) \in \{0, W-1\} \subseteq \Omega$ and hence

$$\phi_S(x, \alpha_0)_{k,r,s} \neq 0 \tag{479}$$

but, for $r > c_W$,

$$\phi_S(x, \alpha_0 + \varepsilon)_{k,r,s} = 0 \quad \forall \varepsilon > 0 \tag{480}$$

or, when $r < c_W$,

$$\phi_S(x, \alpha_0 - \varepsilon)_{k,r,s} = 0 \quad \forall \varepsilon > 0. \tag{481}$$

A similar reasoning leads to a discontinuity in the s-coordinates. We can thus define the set of discontinuities of ψ_k as

$$\mathcal{D} := \left(\bigcup_{r=0}^{W-1} \mathcal{D}_1^r\right) \cup \left(\bigcup_{s=0}^{H-1} \mathcal{D}_2^s\right) \tag{482}$$

where

$$\begin{aligned} \mathcal{D}_1^r &:= \{\alpha_0 \in [a, b] \mid f_1^r(\alpha_0) \in \{0, W-1\}\} \\ \mathcal{D}_2^s &:= \{\alpha_0 \in [a, b] \mid f_2^s(\alpha_0) \in \{0, H-1\}\}. \end{aligned} \tag{483}$$

We notice that $|\mathcal{D}| \leq H + W$ and hence for large enough N, each interval $[\alpha_i, \alpha_{i+1}]$ contains at most 1 discontinuity.

Due to these continuities, we need to modify the general upper bound M of the interpolation error M_S Recall that for $a < b$ and $\{\alpha_i\}_{i=1}^N$, the maximum L_2-sampling error $M_{a,b}$ is given by

$$M_S := \max_{a \leq \alpha \leq b} \min_{1 \leq i \leq N} \|\phi_S(x, \alpha) - \phi_S(x, \alpha_i)\|_2. \tag{484}$$

In order to compute an upper bound on (484) for scaling, we are interested in finding $M \geq 0$ such that

$$M_S^2 \leq M \tag{485}$$

For scaling, similar as in the case for rotations, we sample α_i uniformly from $[a, b]$:

$$\alpha_i = a + \frac{b - a}{N - 1}(i - 1) \text{ for } 1 \leq i \leq N. \tag{486}$$

and note that $\alpha_1 = b$ and $\alpha_N = a$. For $1 \leq i \leq N$ Let g_i be the functions $g_i \colon [a, b] \to \mathbb{R}_{\geq 0}$ defined by

$$g_i(\alpha) := \|\phi_S(x, \alpha) - \phi_S(x, \alpha_i)\|_2^2. \tag{487}$$

Note that $\forall \alpha \in [a, b], \exists i$ such that $\alpha \in [\alpha_{i+1}, \alpha_i]$. Suppose that N is large enough such that $\forall i \colon |\mathcal{D} \cap [\alpha_{i+1}, \alpha_i]| \leq 1$ and denote the discontinuity in interval $[\alpha_{i+1}, \alpha_i]$ by t_i if it exists. Let

$$M_i := \begin{cases} \max\limits_{\alpha_i \leq \alpha \leq \alpha_{i+1}} \min\{g_i(\alpha), g_{i+1}(\alpha)\} & [\alpha_i, \alpha_{i+1}] \cap \mathcal{D} = \varnothing \\ \max\left\{\max\limits_{\alpha_i \leq \alpha \leq t_i} g_{i+1}(\alpha), \ \max\limits_{t_i \leq \alpha \leq \alpha_{i+1}} g_i(\alpha)\right\} & [\alpha_i, \alpha_{i+1}] \cap \mathcal{D} = \{t_i\} \end{cases} \tag{488}$$

Similarly as in the case for rotations, we find

$$M_S^2 \leq \max_{1 \leq i \leq N-1} M_i. \tag{489}$$

For simplicity, we assume for the sequel that $\mathcal{D} = \varnothing$. The case where discontinuities exist can be treated analogously. We further divide each interval $[\alpha_i, \alpha_{i+1}]$ by sampling $n \in \mathbb{N}$ points $\{\gamma_{i,j}\}_{j=1}^n$ according to

$$\gamma_{i,j} := \alpha_i + \frac{\alpha_{i+1} - \alpha_i}{n - 1}(j - 1) \text{ for } 1 \leq j \leq n \tag{490}$$

and define

$$m_{i,j} := \max_{\gamma_{i,j} \leq \gamma \leq \gamma_{i,j+1}} \min\{g_i(\gamma), g_{i+1}(\gamma)\}. \tag{491}$$

We can thus upper bound each M_i by

$$M_i \leq \max_{1 \leq j \leq n-1} m_{i,j}. \tag{492}$$

In order to find an upper bound on M_S^2, we thus need to find an upper bound on $m_{i,j}$ and can proceed analogously to rotations. Namely, setting

$$M := \max_{1 \leq i \leq N-1} \left\{ \max_{1 \leq j \leq n-1} \left\{ \frac{1}{2} \cdot \left(\min\left\{g_i(\gamma_{i,j}) + g_i(\gamma_{i,j+1}), \right. \right. \right. \right. \\ \left. \left. \left. \left. g_{i+1}(\gamma_{i,j}) + g_{i+1}(\gamma_{i,j+1})\right\}\right) + L \cdot \frac{\gamma_{i,j+1} - \gamma_{i,j}}{2} \right\} \right\} \tag{493}$$

yields a computable upper bound of the maximum ℓ_2 interpolation error. Computing a Lipschitz constant for g_i and g_{i+1} is also analogous to rotations. The difference lies only in computing a Lipschitz constant for ϕ_S what we will explain in greater detail.

Recall that Lemma 9 provides a Lipschitz constant for the function $t \mapsto \psi_k(t) := Q_x(k, \rho_1(t), \rho_2(t))$ where ρ is a differentiable

Algorithm 1 Interpolation Error M Computation for Rotation Transformation.

Input: clean input image x;
interval of rotation angle to certify $[a, b]$;
number of first-level samples N;
number of second-level samples n
Output: rotation angle samples $\{\alpha_i\}_{i=1}^N$;
upper bound M of squared ℓ_2-interpolation error
$$M_S^2 = \arg\max_{\alpha \in [a,b]} \min_{1 \le i \le N} \|\tilde{\phi}_R(x, \alpha) - \tilde{\phi}_R(x, \alpha_i)\|_2^2.$$
/* Compute Lipschitz constant L_r (32) */
$\alpha_1 \leftarrow a$
for $i = 1, \ldots, N - 1$ **do**
  $\alpha_{i+1} \leftarrow a + (b - a) \cdot \frac{i}{N-1}$ (21)
  for all $(r, s) \in V$ **do**
    /* V and $\mathcal{P}_{r,s}^{(i)}$ are defined in Lemma 4 */
    Compute trajectory covered grid pixels $\mathcal{P}_{r,s}^{(i)}$
    for $k = 0, \ldots, K - 1$ **do**
      Compute $2d_{r,s} \cdot m_\Delta(x, k, \mathcal{P}_{r,s}^{(i)}) \cdot \bar{m}(x, k, \mathcal{P}_{r,s}^{(i)})$ (32)
    end for
  end for
  $L_{r,i} \leftarrow \sum_{k=0}^{K-1} \sum_{(r,s) \in V} 2d_{r,s} \cdot m_\Delta(x, k, \mathcal{P}_{r,s}^{(i)}) \cdot \bar{m}(x, k, \mathcal{P}_{r,s}^{(i)})$.
end for
$L_r \leftarrow \max_{1 \le i \le N-1} L_{r,i}$ (32)
/* Compute interpolation error bound M (24) from stratified sampling */
for $i = 1, \ldots, N - 1$ **do**
  for $j = 1, \ldots, n$ **do**
    /* Second-level sampling */
    $\gamma_{i,j} \leftarrow \alpha_i + (\alpha_{i+1} - \alpha_i) \cdot \frac{j-1}{n-1}$ (25)
  end for
  $M_i \leftarrow 0$
  for $j = 1, \ldots, n - 1$ **do**
    Compute $g_i(\gamma_{i,j}), g_i(\gamma_{i,j+1}), g_{i+1}(\gamma_{i,j})$, and $g_{i+1}(\gamma_{i,j+1})$ (22)
    $M_i \leftarrow \max\{M_i, \min\{g_i(\gamma_{i,j}) + g_i(\gamma_{i,j+1}),$
        $g_{i+1}(\gamma_{i,j}) + g_{i+1}(\gamma_{i,j+1})\}\}$
  end for
  $M_i \leftarrow \frac{1}{2} M_i + L \cdot \frac{b-a}{(N-1)(n-1)}$ (27)
end for
Return: $M \leftarrow \max_{1 \le i \le N-1} M_i$ (24)

---

curve with values in $\mathbb{R}^2$. Namely, a Lipschitz constant for ψ_k is given by

$$L_k = \max_{t \in [t_1, t_2]} \left(\sqrt{2} \|\dot{\rho}(t)\|_2 \cdot m_\Delta(x, k, \lfloor \rho(t) \rfloor) \right). \tag{494}$$

Consider the curve

$$\rho(t) := (f_1^r(t), f_2^s(t)), \quad t > 0 \tag{495}$$

and note that it is differentiable with derivatives

$$\frac{df_1^r(t)}{dt} = \frac{d}{dt}\left(c_W + \frac{r - c_W}{t}\right) = \frac{c_W - r}{t^2} \tag{496}$$

$$\frac{df_2^s(t)}{dt} = \frac{d}{dt}\left(c_H + \frac{s - c_H}{t}\right) = \frac{c_H - s}{t^2} \tag{497}$$

and

$$\|\dot{\rho}(t)\|_2 = \frac{1}{t^2}\sqrt{(c_W - r)^2 + (c_H - s)^2}. \tag{498}$$

A Lipschitz constant for $\phi_S(x, \cdot)_{k,r,s}$ is thus given by

$$L_k^{r,s} = \max_{t \in [t_1, t_2]} \left(\frac{\sqrt{(c_W - r)^2 + (c_H - s)^2}}{t^2} \cdot \sqrt{2} m_\Delta(x, k, \lfloor \rho(t) \rfloor) \right) \tag{499}$$

$$\le \frac{\sqrt{(c_W - r)^2 + (c_H - s)^2}}{t_1^2} \cdot \sqrt{2} \cdot m_\Delta(x, k, \mathcal{P}_{r,s}) \tag{500}$$

$$\le \frac{\sqrt{(c_W - r)^2 + (c_H - s)^2}}{a^2} \cdot \sqrt{2} \cdot m_\Delta(x, k, \mathcal{P}_{r,s}) \tag{501}$$

where

$$\mathcal{P}_{r,s} = \bigcup_{\alpha \in [t_1, t_2]} \{(\lfloor f_1^r(t) \rfloor, \lfloor f_2^s(t) \rfloor)\}. \tag{502}$$

Finally, setting

$$L_i^{k,r,s} := L_k^{r,s} \cdot \bar{m}(x, k, \mathcal{P}_{r,s}^{(i)}) \tag{503}$$

and

$$\begin{aligned} L_s &= \max_{1 \le i \le N-1} \sum_{k=0}^{K-1} \sum_{(r,s) \in \Omega_{\mathbb{N}}} L_i^{k,r,s} \\ &= \max_{1 \le i \le N-1} \sum_{k=0}^{K-1} \sum_{r,s \in \Omega \cap \mathbb{N}^2} \frac{\sqrt{2} d_{r,s}}{a^2} \\ &\quad \cdot m_\Delta(x, k, \mathcal{P}_{r,s}^{(i)}) \cdot \overline{m}(x, k, \mathcal{P}_{r,s}^{(i)}) \end{aligned} \tag{504}$$

yields the desired Lipschitz constant. □

J ALGORITHM DESCRIPTION FOR DIFFERENTIALLY RESOLVABLE TRANSFORMATIONS

Algorithm 1 presents a pseudo-code for interpolation error M computation, taking rotation transformation as the example. It corresponds to the description in Section 6.2. Algorithm 2 presents a pseudo-code for progressive sampling. It corresponds to the description in Section 6.4.2. We remark that in practice, we sample in mini-batches with batch size B. The error tolerance T is set to M_S (20) if certifying rotation or scaling, is set to $\sqrt{M_S^2 + \sigma^2/\sigma_b^2 \cdot b_0^2}$ (50) if certifying the composition of rotation or scaling with brightness change within $[-b_0, b_0]$; and is set to $\sqrt{(M_S + r)^2 + \sigma^2/\sigma_b^2 \cdot b_0^2}$ (51) if certifying the composition of rotation or scaling, brightness change $[-b_0, b_0]$, and ℓ_2 bounded perturbations within r. The two algorithms jointly constitute our pipeline **TSS-DR** for certifying against differentially resolvable transformations as shown in Figure 5a.

K OMITTED EXPERIMENT DETAILS

Here we provide all omitted details about experiment setup, implementation, discussion about baselines, evaluation protocols, results, findings, and analyses.

K.1 Model Preparation

As previous work shows, an undefended model is very vulnerable even under simple random semantic attacks. Therefore, to obtain nontrivial certified robustness, we require the model itself to be trained to be robust against semantic transformations. We apply data augmentation training [8] combined with Consistency regularization [27] to train the base classifiers. The data augmentation

Algorithm 2 Progressive Sampling for Certification.

Input: clean input image x with true class k_A; first-level parameter samples $\{\alpha_i\}_{i=1}^N$; perturbation random variable ε with variance σ^2; ℓ_2 error tolerance T; batch size B; sampling size limit n_s; confidence level p.
Output: with probability $1 - p$, whether $g(\cdot; \varepsilon)$ is certifiably robust at $\phi(x, \alpha)$.
for $i = 1, \ldots, N$ **do**
    $x^{(i)} \leftarrow \phi(x, \alpha_i)$
    $j \leftarrow 0$
    while $j \leq n_s$ **do**
        Sample B instances of $\phi(x^{(i)}, \varepsilon)$, and use them to update empirical mean $\hat{q}(y_A \,|\, x^{(i)}; \varepsilon)$.
        $j \leftarrow j + B$.
        /* Lower confidence interval bound with these j samples */
        $\underline{p_A}^{(i)} = \text{LowerConfBound}(\hat{q}(y_A \,|\, x^{(i)}; \varepsilon), j, 1 - p/N)$.
        if $\underline{R_i} = \sigma \Phi^{-1}\left(\underline{p_A}^{(i)}\right) > T$ **then**
            /* Already get the certification that $R_i > T$, break */
            Break
        end if
    end while
    if $\underline{R_i} = \sigma \Phi^{-1}\left(\underline{p_A}^{(i)}\right) \leq T$ **then**
        /* Cannot ensure that $R_i > T$. So cannot ensure that $R = \min R_i > T$. Early halt */
        Return: false
    end if
end for
Return: true

---

training randomly transforms the input by the specified transformation using parameters drawn from the specified smoothing distribution/strategy. The Consistency regularization further enhances the consistency of the base classifiers' prediction among the drawn parameters. Then, the base classifiers are used to construct smoothed classifiers by the specified smoothing distribution/strategy, and we compute its robustness certification with our approach.

On relatively small datasets MNIST and CIFAR-10, the models are trained from scratch. On MNIST, we use a convolutional neural network (CNN) composed of four convolutional layers and three fully connected layers. On CIFAR-10, we use the neural network ResNet-110, a 110-layer ResNet model [21]. These model structures are the same as in the literature [8, 44, 65] for direct comparison. On MNIST, we train 100 epochs; on CIFAR-10, we train 150 epochs. The batch sizes (B) are 400 and 256 on MNIST and CIFAR-10, respectively. The learning rate on both datasets is initialized to 0.01, and after every 50 epochs, the learning rate is multiplied by 0.1. For resolvable transformations, the data augmentation usually uses the same smoothing distribution/strategy as we will use to construct the smoothed classifier. In particular, for brightness and contrast transformation, we empirically observe that a larger variance during inference time helps to improve the certified accuracy under large attack radius, and for the composition of Gaussian blur, brightness, contrast, and translation, we additionally add small additive Gaussian noise to improve its ability to defend against other unforeseen attacks as we will discuss in Appendix K.8. For differentially resolvable transformations, since Gaussian noise is required in constructing the smoothed classifier, the data augmentation jointly adds Gaussian noise and the transformation to certify against. The detailed hyperparameters such as distribution type and variance

are listed in Table 7. The weight of Consistency regularization is set to 10 throughout the training.

On the large ImageNet dataset, we finetune the existing trained models. For resolvable transformations, we finetune from ResNet-50 model in `torchvision` library [42]. For differentially resolvable transformations, since the base classifier should also be robust under Gaussian noise, we finetune from Resnet-50 model in [44] that achieves state-of-the-art robustness under Gaussian noise. In either case, we follow the same data augmentation scheme as on MNIST and CIFAR-10, and we finetune for two epochs with batch size (B) 128, learning rate 0.001, and Consistency regularization weight 10. During certification (e.g., Algorithm 2), we use the same batch sizes as during training on these datasets.

The channel-wise normalization is used for all models on these three datasets as in [8, 44]. On all three datasets, in each training epoch, we feed in the whole training dataset without random shuffle.

We remark that since our approach focuses on robustness certification and the smoothing strategy to improve certified robustness, we did not fully explore the potential of improving certified robustness from the training side, nor did we file-tune the training hyperparameters. Therefore, though we already achieved the state of the art by our effective robustness certification and smoothing strategies, we believe the results could be further improved by more effective training approaches.

K.2 Implementation Details

We implement the whole approach along with the training scripts in a tool based on `PyTorch`. For resolvable transformations, we extend the smoothing module from Cohen et al [8] to accommodate various smoothing strategies and smoothing distributions. The predict and certify modules are kept the same. For differentially resolvable transformations, since the stratified sampling requires $N \times n$ times of transformation to compute the interpolation error bound (where N is the number of first-level samples and n the number of second-level samples), we implement a fast `C` module and integrate it to our `Python`-based tool. It empirically achieves $3 - 5x$ speed gain compared with `OpenCV`[39]-based transformation. For Lipschitz upper bound computation, since the loop in `Python` is slow, we reformulate the computation by loop-free tensor computations using `numpy`. It empirically achieves $20-40x$ speed gain compared to the plain loop-based implementation. The full code implementation of our **TSS** tool along with all trained models are publicly available at https://github.com/AI-secure/semantic-randomized-smoothing.

K.3 Details on Attacks

We use the following three attacks to evaluate the empirical accuracy of both **TSS** models and vanilla models: Random Attack, Random+ Attack, and PGD Attack. The Random Attack is used in previous work [3, 14] but does not consider the intrinsic characteristics of semantic transformations. Thus, we propose Random+ Attack and PGD Attack as the alternatives since they are adaptively designed for our smoothed **TSS** models and also consider the intrinsic characteristics of these transformations.

K.3.1 Random Attack. The random attack is used to evaluate the empirical robust accuracy, which is an upper bound of the certified
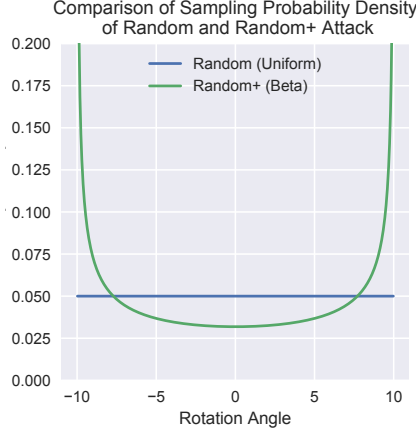
Figure 9: Comparison of probability density of Random and Random+ attack when attacking the rotation transformation with rotation angle between $-10°$ and $+10°$.

robust accuracy. The random attack reads in the clean input, and uniformly samples N parameters from the pre-defined transformation parameter space to transform the input following uniform distribution. If the model gives a wrong prediction on any of these N transformed inputs, we treat this sample as being successfully attacked; otherwise, the sample counts toward the empirical robust accuracy. We denote by N the "number of initial starts". In the main experiments, we set $N = 100$, and in the following ablation study (Appendix K.7), we also compare the behaviors of the three attacks under $N = 10/20/50$.

For transformations with a hyper-rectangle parameter space, including brightness, contrast, scaling, rotation, Gaussian blur, and their compositions, we uniformly sample transformation parameters for each coordinate. For transformations with discrete parameter space, such as translation, we draw the parameter with equal probability. When the transformation is composed with ℓ_p-bounded perturbations, we additionally generate the perturbation vector using FGSM attack [50], where the precise gradient is used for vanilla models, and the empirical mean gradient over 100 samples is used for smoothed **TSS** models.

K.3.2 Adaptive Attack: Random+. The Random+ attack follows the same procedure as the Random attack. The only difference is that, instead of using uniform distribution for sampling transformation parameters, we use the Beta distribution Beta$(0.5, 0.5)$.

Formally, suppose the transformation space is $[a, b]$. In Random attack, we generate the attack parameter ε randomly as follows:

$$\varepsilon' \sim \mathrm{Unif}(0, 1), \quad \varepsilon \leftarrow a + (b - a) \times \varepsilon'. \tag{505}$$

In Random+ attack, we generate the attack parameter δ randomly as follows:

$$\delta' \sim \mathrm{Beta}(0.5, 0.5), \quad \delta \leftarrow a + (b - a) \times \delta'. \tag{506}$$

We choose the Beta distribution because, intuitively, an adversarial example would be more likely to exist at the boundary, i.e., closer to a or b. For example, suppose the rotation attacker has permitted angles in $[-r, r]$, then the adversarial samples may be more likely to have large rotation angle. As shown in Figure 9, the Beta distribution

helps to assign more mass when the parameter becomes closer to the boundary. Choosing other Beta distribution hyperparameters could control the trade-off on sampling weights over the boundary or over center, and we empirically find Beta$(0.5, 0.5)$ already works very well as shown in experiments (Appendix K.7).

K.3.3 Adaptive Attack: PGD. We propose the semantic transformation version of PGD attack as follows: (1) Initialize the transformation parameter following the same process as in Random+ Attack; (2) Suppose the current parameter is $(\alpha_1, \cdots, \alpha_z)$. The attack slightly perturbs each coordinate from α_i to $\alpha_i \pm \tau_i$ respectively and obtain $2z$ perturbed candidates. (3) The attack clips each coordinate to be within the specified range, and then choose the candidate that yields the largest gain of cross-entropy loss (for vanilla models) or empirical mean cross-entropy loss (for **TSS** models) to update the current parameter. Then go to step (2). The loop repeats for 10 iterations for each sample obtained in step (1). The perturbed step size $\tau_i = l_i/10$ where l_i is the length of the specified interval on the i-th coordinate. Finally, if the transformation is composed with ℓ_p-bounded perturbations, we additionally generate the perturbation vector in the same way as in Random attack.

For each Step (1) sample we would have an output and thus there are N outputs. If any of these outputs fool the target model, we treat this sample as being successfully attacked; otherwise, the sample counts toward the empirical robust accuracy. Note that the translation transformation has a discrete parameter space, thus PGD attack is not applicable.

We refer to the attack as the semantic transformation version of PGD attack because: (1) It involves multiple initial starts; (2) It leverages the local landscape information to maximize the loss function iteratively. (3) It clips (i.e., "projects") the parameters to be within the perturbation range. Compared to the classical PGD attack under ℓ_p-norm constraint, we use coordinate-wise perturbations to probe the local landscape to circumvent the hardness of obtaining the gradient with respect to transformation parameters.

K.4 Details on Baseline Approaches

DeepG [3] is based on linear relaxations. The code is open-sourced, and we utilize it to provide a direct comparison. The code provides trained models on MNIST and CIFAR-10, while on ImageNet the method is too slow and memory-consuming to run. On both MNIST and CIFAR-10, we use the provided trained models from the code. In terms of computation time, since our approach uses far less than 1000 s for certification per input on MNIST and CIFAR-10, we tune the hyperparameters to let the code spend roughly 1000 s for certification.

Interval [49] is based on interval bound propagation. We also utilize the open-sourced code to provide a direct comparison. The settings are the same as in DeepG.

VeriVis [41] provides an enumeration-based solution when the number of possible transformation parameters or the number of possible transformed images is finite. In our evaluation, only translation satisfies this property. Therefore, as the baseline, we implement the enumeration-based robustness certification algorithm for our trained robust models.

Semanify-NN [37] proposes to insert a preprocessing layer to reduce the verification against semantic transformations to the

Table 6: Benign accuracy of our **TSS** models corresponding to those in Table 2. Certified robust accuracy shown as reference.

Transformation	Dataset	Attack Radius	Certified Robust Acc.	Benign Acc.
Gaussian Blur	MNIST	Squared Radius $\alpha \leq 36$	90.6%	96.8%
	CIFAR-10	Squared Radius $\alpha \leq 16$	63.6%	76.2%
	ImageNet	Squared Radius $\alpha \leq 36$	51.6%	59.2%
Translation (Reflection Pad.)	MNIST	$\sqrt{\Delta x^2 + \Delta y^2} \leq 8$	99.6%	99.6%
	CIFAR-10	$\sqrt{\Delta x^2 + \Delta y^2} \leq 20$	80.8%	87.0%
	ImageNet	$\sqrt{\Delta x^2 + \Delta y^2} \leq 100$	50.0%	73.0%
Brightness	MNIST	$b \pm 50\%$	98.2%	98.2%
	CIFAR-10	$b \pm 40\%$	87.0%	87.8%
	ImageNet	$b \pm 40\%$	70.0%	72.2%
Contrast and Brightness	MNIST	$c \pm 50\%, b \pm 50\%$	97.6%	98.0%
	CIFAR-10	$c \pm 40\%, b \pm 40\%$	82.4%	86.8%
	ImageNet	$c \pm 40\%, b \pm 40\%$	61.4%	72.2%
Gaussian Blur, Translation, Brightness, and Contrast	MNIST	$\alpha \leq 1, c, b \pm 10\%, \sqrt{\Delta x^2 + \Delta y^2} \leq 5$	90.2%	98.2%
	CIFAR-10	$\alpha \leq 1, c, b \pm 10\%, \sqrt{\Delta x^2 + \Delta y^2} \leq 5$	58.2%	77.6%
	ImageNet	$\alpha \leq 10, c, b \pm 20\%, \sqrt{\Delta x^2 + \Delta y^2} \leq 10$	32.8%	61.6%
Rotation	MNIST	$r \pm 50°$	97.4%	99.4%
	CIFAR-10	$r \pm 10°$	70.6%	83.2%
		$r \pm 30°$	63.6%	82.6%
	ImageNet	$r \pm 30°$	30.4%	46.2%
Scaling	MNIST	$s \pm 30\%$	99.0%	99.4%
	CIFAR-10	$s \pm 30\%$	58.8%	79.8%
	ImageNet	$s \pm 30\%$	26.4%	50.8%
Rotation and Brightness	MNIST	$r \pm 50°, b \pm 20\%$	97.0%	99.4%
	CIFAR-10	$r \pm 10°, b \pm 10\%$	70.2%	83.0%
		$r \pm 30°, b \pm 20\%$	61.4%	82.6%
	ImageNet	$r \pm 30°, b \pm 20\%$	26.8%	45.8%
Scaling and Brightness	MNIST	$s \pm 50\%, b \pm 50\%$	96.6%	99.4%
	CIFAR-10	$s \pm 30\%, b \pm 30\%$	54.2%	79.6%
	ImageNet	$s \pm 30\%, b \pm 30\%$	23.4%	50.8%
Rotation, Brightness, and ℓ_2	MNIST	$r \pm 50°, b \pm 20\%, \|\delta\|_2 \leq .05$	96.6%	99.4%
	CIFAR-10	$r \pm 10°, b \pm 10\%, \|\delta\|_2 \leq .05$	64.2%	83.0%
		$r \pm 30°, b \pm 20\%, \|\delta\|_2 \leq .05$	55.2%	82.6%
	ImageNet	$r \pm 30°, b \pm 20\%, \|\delta\|_2 \leq .05$	26.6%	45.8%
Scaling, Brightness, and ℓ_2	MNIST	$s \pm 50\%, b \pm 50\%, \|\delta\|_2 \leq .05$	96.4%	99.4%
	CIFAR-10	$s \pm 30\%, b \pm 30\%, \|\delta\|_2 \leq .05$	51.2%	79.6%
	ImageNet	$s \pm 30\%, b \pm 30\%, \|\delta\|_2 \leq .05$	22.6%	50.8%

verification against classical ℓ_p noises. To our knowledge, the code has not been open-sourced yet. Therefore, we directly compare with the numbers reported in their paper. Since they report the average of certified robust magnitude, we apply Markov's inequality to obtain an upper bound of their certified robust accuracy. For example, they report 46.24 degrees as the average certified robust rotation angles. It means that $\mathbb{P}[r \geq 50°] \leq \mathbb{E}[r]/50 = 92.48\%$, i.e., the certified robust accuracy is no larger than 92.48% when fixing the rotation angle to be $50°$.

DistSPT [14] combines randomized smoothing and interval bound propagation to provide certified robustness against semantic transformations. Concretely, the approach leverages interval bound propagation to compute the upper bound of interpolation error and then applies randomized smoothing. On small datasets such as MNIST and CIFAR-10, the approach is able to provide nontrivial robustness certification. Though the certified robust accuracy is inferior than **TSS** as reflected in Table 2. We use their reported numbers in [14, Table 4] for DistSPT^x for comparison, since the certification goal and evaluation protocol are the same as ours. On ImageNet, as described in [14, Section 7.4], the interval bound propagation is computationally expensive and loose. Therefore, they use sampling to estimate the interpolation error, which makes the robustness certification no longer hold against arbitrary attacks but just a certain random attack ("worst-of-10" attack).

IndivSPT [14] provides a different certification goal from the above approaches. At a high level, the approach uses a transformed image as the input where the transformation parameter is within predefined threshold. Then the approach certifies whether the prediction for the transformed image and the prediction for the original image are the same. In contrast, **TSS** and other baseline approaches take original image as the input and certifies whether there exists no transformed image that can mislead the model. Due to different certification goals, **TSS** is not comparable with IndivSPT.

K.5 Benign Accuracy

Table 6 shows the benign accuracy of our models corresponding to Table 2. For comparison, the vanilla trained models have benign accuracy 98.6% on MNIST, 88.6% on CIFAR-10, and 74.4% on ImageNet.

We observe that though the trade-off between accuracy and (certified) robustness is widely reported both theoretically [36, 65, 67] and empirically (e.g., [8, 27, 66]) in classical ℓ_p threat model, it does not always exist in our semantic defense setting. Specifically, for resolvable transformations, we do not observe an apparent loss of benign accuracy in our certifiably robust models; while for differentially resolvable transformations (those involving scaling and rotation), there is no loss on MNIST, slight losses on CIFAR-10, and apparent losses on ImageNet. When there does exist a trade-off between benign accuracy and certified robust accuracy, we show that smoothing variance levels control it in Appendix K.10.

K.6 Smoothing Distributions and Running Time Statistics

In Table 7, we present the smoothing distributions with concrete parameters and average certification computing time per sample for results in main table (Table 2). In the table, α is for squared kernel radius for Gaussian blur; Δx and Δy are for translation displacement on horizontal and vertical direction; b and c are for brightness shift and contrast change respectively as in $x \mapsto (1 + c)x + b$; r is for rotation angle; s is for size scaling ratio; ε is for additive noise vector; and $\|\delta\|_2$ for ℓ_2 norm of permitted additional perturbations. Specifically, "Training Distribution" stands for the distributions for data augmentation during training the base classifiers; and "Smoothing Distribution" stands for the distributions for constructing the smoothed classifiers for certification.

We select these distributions according to the principles in Appendix K.1.

K.7 Comparison of Random Attack and Adaptive Attacks: Detail

In Table 2, we compare the empirical robust accuracy of vanilla models and **TSS** models under random attacks and two adaptive attacks: Random+ and PGD. However, due to space limits, we omit the empirical accuracy of each adaptive attack and we just present the minimum empirical accuracy among them. In Table 8, Table 9, and Table 10, for all transformations on MNIST, CIFAR-10, and ImageNet respectively, we present the detailed empirical accuracy of each attack under different number of initial starts $N = 10/20/50/100$. Note that the main table (Table 2) shows the empirical accuracy with $N = 100$ for all attacks.

Table 7: Detailed smoothing distributions and running time statistics for our **TSS**. $\mathcal{N}(\mu, \Sigma)$ is the normal distribution, $\exp(\lambda)$ is the exponential distribution, $\mathcal{U}([a,b])$ is the uniform distribution. Random variable ε is the elementwise noise as in Corollary 2. "Cert." means certification.

Transformation	Dataset	Attack Radius	Training Distribution	Smoothing Distribution	Avg. Cert. Time per Sample
Gaussian Blur	MNIST	Squared Radius $\alpha \le 36$	$\alpha \sim \text{Exp}(1/10)$		7.9 s
	CIFAR-10	Squared Radius $\alpha \le 16$	$\alpha \sim \text{Exp}(1/5)$		30.9 s
	ImageNet	Squared Radius $\alpha \le 36$	$\alpha \sim \text{Exp}(1/10)$		45.7 s
Translation (Reflection Pad.)	MNIST	$\sqrt{\Delta x^2 + \Delta y^2} \le 8$	$(\Delta x, \Delta y) \sim \mathcal{N}(0, 10^2 I)$		10.2 s
	CIFAR-10	$\sqrt{\Delta x^2 + \Delta y^2} \le 20$	$(\Delta x, \Delta y) \sim \mathcal{N}(0, 15^2 I)$		39.4 s
	ImageNet	$\sqrt{\Delta x^2 + \Delta y^2} \le 100$	$(\Delta x, \Delta y) \sim \mathcal{N}(0, 30^2 I)$		161.9 s
Brightness	MNIST	$b \pm 50\%$	$b \sim \mathcal{N}(0, 0.6^2)$		2.1 s
	CIFAR-10	$b \pm 40\%$	$b \sim \mathcal{N}(0, 0.3^2)$		4.4 s
	ImageNet	$b \pm 40\%$	$b \sim \mathcal{N}(0, 0.4^2)$		45.1 s
Contrast and Brightness	MNIST	$c \pm 50\%, b \pm 50\%$	$(c, b) \sim \mathcal{N}(0, 0.6^2 I)$	$(c, b) \sim \mathcal{N}(0, 1.0^2 I)$	9.8 s
	CIFAR-10	$c \pm 40\%, b \pm 40\%$	$(c, b) \sim \mathcal{N}(0, 0.4^2 I)$	$(c, b) \sim \mathcal{N}(0, 0.6^2 I)$	45.0 s
	ImageNet	$c \pm 40\%, b \pm 40\%$	$(c, b) \sim \mathcal{N}(0, 0.4^2 I)$		325.6 s
Gaussian Blur, Translation, Brightness, and Contrast	MNIST	$\alpha \le 5, c, b \pm 10\%,$ $\sqrt{\Delta x^2 + \Delta y^2} \le 5$	$\alpha \sim \text{Exp}(1/10)$ $(\Delta x, \Delta y) \sim \mathcal{N}(0, 10^2 I)$ $(c, b) \sim \mathcal{N}(0, 0.3^2 I)$ $\varepsilon \sim \mathcal{N}(0, 0.05^2 I)$	$\alpha \sim \text{Exp}(1/10)$ $(\Delta x, \Delta y) \sim \mathcal{N}(0, 10^2 I)$ $(c, b) \sim \mathcal{N}(0, 0.3^2 I)$	12.9 s
	CIFAR-10	$\alpha \le 1, c, b \pm 10\%,$ $\sqrt{\Delta x^2 + \Delta y^2} \le 5$	$\alpha \sim \text{Exp}(1)$ $(\Delta x, \Delta y) \sim \mathcal{N}(0, 10^2 I)$ $(c, b) \sim \mathcal{N}(0, 0.3^2 I)$ $\varepsilon \sim \mathcal{N}(0, 0.01^2 I)$	$\alpha \sim \text{Exp}(1)$ $(\Delta x, \Delta y) \sim \mathcal{N}(0, 10^2 I)$ $(c, b) \sim \mathcal{N}(0, 0.3^2 I)$	43.1 s
	ImageNet	$\alpha \le 10, c, b \pm 20\%,$ $\sqrt{\Delta x^2 + \Delta y^2} \le 10$	$\alpha \sim \text{Exp}(1/5)$ $(\Delta x, \Delta y) \sim \mathcal{N}(0, 20^2 I)$ $(c, b) \sim \mathcal{N}(0, 0.4^2 I)$ $\varepsilon \sim \mathcal{N}(0, 0.01^2 I)$	$\alpha \sim \text{Exp}(1/5)$ $(\Delta x, \Delta y) \sim \mathcal{N}(0, 20^2 I)$ $(c, b) \sim \mathcal{N}(0, 0.4^2 I)$	238.1 s
Rotation	MNIST	$r \pm 50°$	Same as Rotation and Brightness	$\varepsilon \sim \mathcal{N}(0, 0.12^2 I)$	20.1 s
	CIFAR-10	$r \pm 10°$		$\varepsilon \sim \mathcal{N}(0, 0.05^2 I)$	52.8 s
		$r \pm 30°$		$\varepsilon \sim \mathcal{N}(0, 0.05^2 I)$	141.0 s
	ImageNet	$r \pm 30°$		$\varepsilon \sim \mathcal{N}(0, 0.5^2 I)$	2358.1 s
Scaling	MNIST	$s \pm 30\%$	Same as Scaling and Brightness	$\varepsilon \sim \mathcal{N}(0, 0.12^2 I)$	17.7 s
	CIFAR-10	$s \pm 30\%$		$\varepsilon \sim \mathcal{N}(0, 0.12^2 I)$	42.2 s
	ImageNet	$s \pm 30\%$		$\varepsilon \sim \mathcal{N}(0, 0.5^2 I)$	1201.2 s
Rotation and Brightness	MNIST	$r \pm 50°, b \pm 20\%$	$r \sim \mathcal{U}([-55, 55])$ $\varepsilon \sim \mathcal{N}(0, 0.12^2 I)$ $b \sim \mathcal{N}(0, 0.2^2)$	$\varepsilon \sim \mathcal{N}(0, 0.12^2 I)$ $b \sim \mathcal{N}(0, 0.2^2)$	31.4 s
	CIFAR-10	$r \pm 10°, b \pm 10\%$	$r \sim \mathcal{U}([-12.5, 12.5])$ $\varepsilon \sim \mathcal{N}(0, 0.05^2 I)$ $b \sim \mathcal{N}(0, 0.2^2)$	$\varepsilon \sim \mathcal{N}(0, 0.05^2 I)$ $b \sim \mathcal{N}(0, 0.2^2)$	62.3 s
		$r \pm 30°, b \pm 20\%$	$r \sim \mathcal{U}([-35, 35])$ $\varepsilon \sim \mathcal{N}(0, 0.05^2 I)$ $b \sim \mathcal{N}(0, 0.2^2)$	$\varepsilon \sim \mathcal{N}(0, 0.05^2 I)$ $b \sim \mathcal{N}(0, 0.2^2)$	157.0 s
	ImageNet	$r \pm 30°, b \pm 20\%$	$r \sim \mathcal{U}([-35, 35])$ $\varepsilon \sim \mathcal{N}(0, 0.5^2 I)$ $b \sim \mathcal{N}(0, 0.2^2)$	$\varepsilon \sim \mathcal{N}(0, 0.5^2 I)$ $b \sim \mathcal{N}(0, 0.2^2)$	2475.6 s
Scaling and Brightness	MNIST	$s \pm 50\%, b \pm 50\%$	$s \sim \mathcal{U}([0.45, 1.55])$ $\varepsilon \sim \mathcal{N}(0, 0.12^2 I)$ $b \sim \mathcal{N}(0, 0.5^2)$	$\varepsilon \sim \mathcal{N}(0, 0.12^2 I)$ $b \sim \mathcal{N}(0, 0.5^2)$	74.9 s
	CIFAR-10	$s \pm 30\%, b \pm 30\%$	$s \sim \mathcal{U}([0.65, 1.35])$ $\varepsilon \sim \mathcal{N}(0, 0.12^2 I)$ $b \sim \mathcal{N}(0, 0.3^2)$	$\varepsilon \sim \mathcal{N}(0, 0.12^2 I)$ $b \sim \mathcal{N}(0, 0.3^2)$	44.5 s
	ImageNet	$s \pm 30\%, b \pm 30\%$	$s \sim \mathcal{U}([0.65, 1.35])$ $\varepsilon \sim \mathcal{N}(0, 0.5^2 I)$ $b \sim \mathcal{N}(0, 0.3^2)$	$\varepsilon \sim \mathcal{N}(0, 0.5^2 I)$ $b \sim \mathcal{N}(0, 0.3^2)$	1401.6 s
Rotation, Brightness, and ℓ_2	MNIST	$r \pm 50°, b \pm 20\%, \|\delta\|_2 \le .05$	Same as Rotation and Brightness	Same as Rotation and Brightness	35.1 s
	CIFAR-10	$r \pm 10°, b \pm 10\%, \|\delta\|_2 \le .05$			132.5 s
		$r \pm 30°, b \pm 20\%, \|\delta\|_2 \le .05$			520.2 s
	ImageNet	$r \pm 30°, b \pm 20\%, \|\delta\|_2 \le .05$			3463.8 s
Scaling, Brightness, and ℓ_2	MNIST	$s \pm 50\%, b \pm 50\%, \|\delta\|_2 \le .05$	Same as Scaling and Brightness	Same as Scaling and Brightness	75.1 s
	CIFAR-10	$s \pm 30\%, b \pm 30\%, \|\delta\|_2 \le .05$			50.0 s
	ImageNet	$s \pm 30\%, b \pm 30\%, \|\delta\|_2 \le .05$			1657.7 s

Table 8: Comparison between empirical robust accuracy against random and adaptive attacks and certified robust accuracy on **MNIST**. The attack radii are consistent with Table 2. The most powerful attack in each setting is highlighted in bold font. The adaptive attacks are shown in gray rows. Note that PGD attack cannot apply to translation transformation because the parameter space is discrete.

Transformation	Attack Radius	Model	Attack	Empirical Robust Accuracy Initial Starts $N=$				Certified Robust Accuracy
				10	20	50	100	
Gaussian Blur	Squared Radius $\alpha \leq 36$	TSS	Random	93.2%	92.2%	92.0%	91.4%	90.6%
			Random+	92.4%	92.2%	**91.2%**	**91.2%**	
			PGD	**91.6%**	**91.6%**	91.6%	91.6%	
		Vanilla	Random	14.0%	12.4%	12.2%	12.2%	-
			Random+	12.4%	12.4%	12.2%	12.2%	
			PGD	12.2%	12.2%	12.2%	12.2%	
Translation (Reflection Pad.)	$\sqrt{\Delta x^2 + \Delta y^2} \leq 8$	TSS	Random	99.6%	99.6%	99.6%	99.6%	99.6%
			Random+	99.6%	99.6%	99.6%	99.6%	
			PGD	-	-	-	-	
		Vanilla	Random	0.0%	0.0%	0.0%	0.0%	-
			Random+	0.0%	0.0%	0.0%	0.0%	
			PGD	-	-	-	-	
Brightness	$b \pm 50\%$	TSS	Random	98.2%	98.2%	98.2%	98.2%	98.2%
			Random+	98.2%	98.2%	98.2%	98.2%	
			PGD	98.2%	98.2%	98.2%	98.2%	
		Vanilla	Random	97.2%	96.6%	96.6%	96.6%	-
			Random+	96.8%	96.6%	96.6%	96.6%	
			PGD	96.6%	96.6%	96.6%	96.6%	
Contrast and Brightness	$c \pm 50\%$, $b \pm 50\%$	TSS	Random	98.0%	98.0%	98.0%	98.0%	97.6%
			Random+	98.0%	98.0%	98.0%	98.0%	
			PGD	98.0%	98.0%	98.0%	98.0%	
		Vanilla	Random	96.8%	95.8%	95.0%	94.6%	-
			Random+	95.8%	94.4%	93.8%	93.6%	
			PGD	93.6%	93.4%	93.2%	93.2%	
Gaussian Blur, Translation, Contrast, and Brightness	$\alpha \leq 5$, $c \pm 10\%$, $b \pm 10\%$, $\sqrt{\Delta x^2 + \Delta y^2} \leq 5$	TSS	Random	97.6%	97.6%	97.6%	97.2%	90.2%
			Random+	97.6%	**97.2%**	**97.0%**	**97.0%**	
			PGD	**97.4%**	**97.4%**	**97.2%**	**97.0%**	
		Vanilla	Random	10.0%	4.4%	1.4%	**0.4%**	-
			Random+	**6.8%**	2.4%	1.2%	**0.4%**	
			PGD	7.0%	**1.4%**	0.8%	**0.4%**	
Rotation	$r \pm 50°$	TSS	Random	98.6%	**98.4%**	98.2%	98.4%	97.4%
			Random+	98.6%	98.6%	98.4%	**98.2%**	
			PGD	**98.2%**	**98.4%**	98.4%	**98.2%**	
		Vanilla	Random	27.2%	17.4%	13.8%	12.2%	-
			Random+	**15.4%**	**13.0%**	**11.0%**	**11.0%**	
			PGD	16.4%	15.6%	15.4%	15.2%	
Scaling	$s \pm 30\%$	TSS	Random	99.2%	99.2%	99.2%	99.2%	97.2%
			Random+	99.2%	99.2%	99.2%	99.2%	
			PGD	99.2%	99.2%	99.2%	99.2%	
		Vanilla	Random	92.0%	91.4%	90.2%	90.2%	-
			Random+	**90.0%**	**89.4%**	**89.2%**	**89.2%**	
			PGD	90.4%	90.2%	90.2%	90.2%	
Rotation and Brightness	$r \pm 50\%$, $b \pm 20\%$	TSS	Random	98.8%	98.4%	98.2%	98.2%	97.0%
			Random+	98.6%	98.2%	**98.0%**	98.2%	
			PGD	**98.2%**	**98.0%**	**98.0%**	**98.0%**	
		Vanilla	Random	28.8%	17.8%	12.6%	11.0%	-
			Random+	16.6%	**11.6%**	**10.4%**	**10.4%**	
			PGD	**13.4%**	13.6%	13.0%	12.6%	
Scaling and Brightness	$s \pm 50\%$, $b \pm 50\%$	TSS	Random	98.6%	98.6%	98.4%	**97.8%**	96.6%
			Random+	98.4%	98.0%	97.8%	**97.8%**	
			PGD	**98.2%**	**97.8%**	**97.8%**	**97.8%**	
		Vanilla	Random	57.4%	46.0%	31.0%	24.8%	-
			Random+	40.4%	28.0%	**19.8%**	**15.6%**	
			PGD	**29.0%**	**25.2%**	25.0%	24.0%	
Rotation, Brightness, and ℓ_2	$r \pm 50\%$, $b \pm 20\%$, $\|\delta\|_2 \leq .05$	TSS	Random	98.2%	97.8%	**97.6%**	97.6%	96.6%
			Random+	98.4%	98.0%	97.8%	97.6%	
			PGD	**97.8%**	**97.6%**	**97.6%**	**97.4%**	
		Vanilla	Random	27.6%	17.2%	11.4%	10.8%	-
			Random+	15.2%	**11.2%**	**9.4%**	**9.0%**	
			PGD	**13.4%**	12.8%	12.0%	11.8%	
Scaling, Brightness, and ℓ_2	$s \pm 50\%$, $b \pm 50\%$, $\|\delta\|_2 \leq .05$	TSS	Random	98.4%	98.4%	**97.6%**	**97.6%**	96.4%
			Random+	97.8%	97.8%	97.6%	97.6%	
			PGD	**97.8%**	**97.6%**	97.6%	97.6%	
		Vanilla	Random	50.4%	38.2%	28.2%	22.2%	-
			Random+	34.4%	23.2%	**13.4%**	**12.2%**	
			PGD	**23.4%**	**22.0%**	21.6%	20.8%	

Table 9: Comparison between empirical robust accuracy against random and adaptive attacks and certified robust accuracy on **CIFAR-10**. The attack radii are consistent with Table 2. The most powerful attack in each setting is highlighted in bold font. The adaptive attacks are shown in gray rows. Note that the PGD attack cannot apply to translation transformation because the parameter space is discrete.

Transformation	Attack Radius	Model	Attack	Empirical Robust Accuracy Initial Starts $N=$				Certified Robust Accuracy
				10	20	50	100	
Gaussian Blur	Squared Radius $\alpha \leq 16$	TSS	Random	66.4%	66.4%	**65.8%**	**65.8%**	63.6%
			Random+	66.8%	66.0%	**65.8%**	**65.8%**	
			PGD	**65.8%**	**65.8%**	**65.8%**	**65.8%**	
		Vanilla	Random	4.8%	4.2%	3.4%	3.4%	-
			Random+	4.6%	4.0%	3.6%	3.4%	
			PGD	**3.4%**	**3.4%**	**3.4%**	**3.4%**	
Translation (Reflection Pad.)	$\sqrt{\Delta x^2 + \Delta y^2} \leq 20$	TSS	Random	86.2%	**86.0%**	86.2%	86.2%	80.8%
			Random+	86.4%	**86.0%**	**86.0%**	**86.0%**	
			PGD	-	-	-	-	
		Vanilla	Random	8.0%	7.0%	4.4%	4.2%	-
			Random+	8.2%	7.2%	**4.2%**	**4.2%**	
			PGD	-	-	-	-	
Brightness	$b \pm 40\%$	TSS	Random	87.2%	87.2%	87.4%	**87.2%**	87.0%
			Random+	**87.0%**	**87.0%**	**87.0%**	87.4%	
			PGD	87.4%	87.4%	87.4%	87.4%	
		Vanilla	Random	57.8%	51.2%	45.8%	44.4%	-
			Random+	**49.8%**	**44.2%**	**42.8%**	**42.6%**	
			PGD	52.4%	51.0%	50.8%	50.8%	
Contrast and Brightness	$c \pm 40\%$, $b \pm 40\%$	TSS	Random	86.2%	86.2%	86.2%	86.0%	82.4%
			Random+	**85.8%**	86.2%	86.0%	**85.8%**	
			PGD	**85.8%**	**85.8%**	**85.8%**	**85.8%**	
		Vanilla	Random	48.0%	40.0%	27.2%	21.0%	-
			Random+	32.0%	23.2%	14.8%	9.6%	
			PGD	**17.0%**	**13.0%**	**12.2%**	11.8%	
Gaussian Blur, Translation, Contrast, and Brightness	$\alpha \leq 1$, $c \pm 10\%$, $b \pm 10\%$, $\sqrt{\Delta x^2 + \Delta y^2} \leq 5$	TSS	Random	71.0%	69.2%	68.0%	67.6%	58.2%
			Random+	70.6%	69.8%	68.4%	67.8%	
			PGD	**69.8%**	**69.8%**	**69.0%**	**68.0%**	
		Vanilla	Random	21.2%	16.6%	12.0%	9.6%	-
			Random+	18.6%	14.2%	9.0%	7.2%	
			PGD	**12.8%**	**9.8%**	**6.8%**	**5.6%**	
Rotation	$r \pm 10°$	TSS	Random	78.0%	77.0%	76.8%	76.6%	70.6%
			Random+	77.4%	**76.8%**	**76.4%**	**76.4%**	
			PGD	**76.8%**	**76.8%**	76.8%	76.6%	
		Vanilla	Random	69.2%	68.0%	**65.6%**	65.6%	-
			Random+	68.4%	67.2%	66.0%	65.6%	
			PGD	**66.4%**	**66.0%**	**65.6%**	**65.4%**	
Rotation	$r \pm 30°$	TSS	Random	71.8%	70.2%	69.8%	**69.2%**	63.6%
			Random+	71.0%	**69.4%**	**69.2%**	69.4%	
			PGD	**70.4%**	70.0%	70.0%	69.8%	
		Vanilla	Random	31.6%	27.4%	22.6%	21.6%	-
			Random+	32.2%	27.2%	23.8%	**21.4%**	
			PGD	**25.2%**	**23.8%**	**23.2%**	23.2%	
Scaling	$s \pm 30\%$	TSS	Random	69.6%	67.8%	67.8%	67.2%	58.8%
			Random+	69.2%	68.4%	67.4%	**67.0%**	
			PGD	**67.8%**	**67.6%**	**67.2%**	**67.0%**	
		Vanilla	Random	60.0%	54.6%	52.8%	51.6%	-
			Random+	56.6%	53.8%	52.2%	**51.2%**	
			PGD	**53.2%**	**52.4%**	**52.0%**	52.0%	
Rotation and Brightness	$r \pm 10°$, $b \pm 10\%$	TSS	Random	77.2%	76.8%	77.0%	76.6%	70.6%
			Random+	77.2%	**76.6%**	**76.4%**	**76.0%**	
			PGD	**76.6%**	**76.6%**	**76.4%**	76.4%	
		Vanilla	Random	67.2%	64.8%	60.6%	59.4%	-
			Random+	66.0%	63.0%	59.4%	57.8%	
			PGD	**57.8%**	**57.6%**	**57.0%**	**56.8%**	
Rotation and Brightness	$r \pm 30°$, $b \pm 20\%$	TSS	Random	72.0%	70.2%	68.8%	68.4%	61.4%
			Random+	70.6%	68.8%	**68.0%**	68.2%	
			PGD	**69.2%**	**68.6%**	68.6%	68.6%	
		Vanilla	Random	26.6%	20.2%	15.8%	13.0%	-
			Random+	18.8%	16.0%	11.6%	9.4%	
			PGD	**12.2%**	**10.4%**	**9.2%**	**9.0%**	
Scaling and Brightness	$s \pm 30\%$, $b \pm 30\%$	TSS	Random	68.6%	68.6%	67.4%	67.2%	54.2%
			Random+	68.4%	68.0%	67.0%	**66.8%**	
			PGD	**67.4%**	**67.4%**	**66.8%**	**66.8%**	
		Vanilla	Random	39.2%	30.6%	20.0%	17.4%	-
			Random+	30.4%	19.4%	15.4%	**11.6%**	
			PGD	**16.0%**	**14.4%**	**13.0%**	13.0%	
Rotation, Brightness, and ℓ_2	$r \pm 10°$, $b \pm 10\%$, $\|\delta\|_2 \leq .05$	TSS	Random	74.2%	72.8%	71.8%	71.6%	64.2%
			Random+	72.8%	72.2%	71.8%	**71.2%**	
			PGD	**71.6%**	**71.6%**	**71.6%**	71.6%	
		Vanilla	Random	40.4%	35.8%	34.4%	31.8%	-
			Random+	36.4%	34.6%	**30.8%**	**29.6%**	
			PGD	**36.0%**	**35.0%**	34.6%	34.6%	
Rotation, Brightness, and ℓ_2	$r \pm 30°$, $b \pm 20\%$, $\|\delta\|_2 \leq .05$	TSS	Random	67.6%	66.2%	64.8%	65.2%	55.2%
			Random+	65.6%	65.6%	65.2%	64.4%	
			PGD	**65.2%**	**64.6%**	**64.0%**	**64.0%**	
		Vanilla	Random	7.6%	5.4%	2.6%	0.8%	-
			Random+	3.8%	2.4%	1.2%	**0.4%**	
			PGD	**1.2%**	**0.6%**	**0.6%**	0.6%	
Scaling, Brightness, and ℓ_2	$s \pm 30\%$, $b \pm 30\%$, $\|\delta\|_2 \leq .05$	TSS	Random	67.6%	66.8%	65.2%	65.0%	51.2%
			Random+	66.0%	66.2%	64.6%	64.4%	
			PGD	**64.2%**	**62.2%**	**61.8%**	**61.8%**	
		Vanilla	Random	15.6%	11.4%	5.8%	4.4%	-
			Random+	8.2%	5.0%	**2.0%**	**2.0%**	
			PGD	**3.8%**	**2.8%**	2.8%	2.6%	

Table 10: Comparison between empirical robust accuracy against random and adaptive attacks and certified robust accuracy on **ImageNet**. The attack radii are consistent with Table 2. The most powerful attack in each setting is highlighted in bold font. The adaptive attacks are shown in gray rows. Note that PGD attack cannot apply to translation transformation because the parameter space is discrete.

Transformation	Attack Radius	Model	Attack	Empirical Robust Accuracy Initial Starts $N =$ 10	20	50	100	Certified Robust Accuracy
Gaussian Blur	Squared Radius $\alpha \leq 36$	TSS	Random	53.2%	**52.8%**	**52.8%**	52.8%	
			Random+	53.2%	**52.8%**	**52.8%**	52.8%	51.6%
			PGD	**52.8%**	**52.8%**	**52.8%**	**52.6%**	
		Vanilla	Random	9.6%	8.6%	8.4%	8.4%	
			Random+	8.8%	**8.2%**	**8.2%**	**8.2%**	-
			PGD	**8.4%**	**8.2%**	**8.2%**	**8.2%**	
Translation (Reflection Pad.)	$\sqrt{\Delta x^2 + \Delta y^2} \leq 100$	TSS	Random	70.0%	69.6%	**69.2%**	**69.2%**	
			Random+	**69.4%**	**69.2%**	**69.2%**	**69.2%**	50.0%
			PGD	-	-	-	-	
		Vanilla	Random	**55.8%**	**53.4%**	**48.8%**	46.6%	
			Random+	57.2%	54.6%	50.6%	**46.2%**	-
			PGD	-	-	-	-	
Brightness	$b \pm 40\%$	TSS	Random	70.8%	**70.4%**	**70.4%**	**70.4%**	
			Random+	**70.4%**	**70.4%**	**70.4%**	**70.4%**	70.0%
			PGD	**70.4%**	**70.4%**	**70.4%**	**70.4%**	
		Vanilla	Random	31.6%	26.6%	21.6%	19.6%	
			Random+	22.8%	**19.8%**	**18.4%**	**18.4%**	-
			PGD	**22.0%**	22.4%	21.8%	21.8%	
Contrast and Brightness	$c \pm 40\%$, $b \pm 40\%$	TSS	Random	70.4%	69.2%	**68.4%**	**68.4%**	
			Random+	69.2%	68.8%	**68.4%**	**68.4%**	61.4%
			PGD	**68.4%**	**68.4%**	**68.4%**	**68.4%**	
		Vanilla	Random	20.8%	10.4%	3.6%	1.2%	
			Random+	8.0%	2.0%	0.4%	**0.0%**	-
			PGD	**1.8%**	**0.2%**	**0.0%**	**0.0%**	
Gaussian Blur, Translation, Contrast, and Brightness	$\alpha \leq 10$, $c \pm 20\%$, $b \pm 20\%$, $\sqrt{\Delta x^2 + \Delta y^2} \leq 10$	TSS	Random	51.8%	50.2%	49.2%	48.8%	
			Random+	51.4%	**49.6%**	**48.0%**	48.2%	32.8%
			PGD	**49.6%**	**49.6%**	48.2%	**47.4%**	
		Vanilla	Random	20.6%	17.4%	12.0%	9.4%	
			Random+	15.2%	12.8%	7.8%	6.6%	-
			PGD	**11.2%**	**8.0%**	**6.0%**	**4.0%**	
Rotation	$r \pm 30\%$	TSS	Random	40.2%	**38.4%**	38.4%	**37.8%**	
			Random+	**39.0%**	38.6%	**38.0%**	**37.8%**	30.4%
			PGD	40.4%	39.8%	39.8%	39.4%	
		Vanilla	Random	47.8%	44.4%	41.4%	40.0%	
			Random+	45.0%	43.6%	40.6%	38.8%	-
			PGD	**39.6%**	**38.4%**	**37.8%**	**37.0%**	
Scaling	$s \pm 30\%$	TSS	Random	40.2%	38.0%	37.4%	37.4%	
			Random+	**38.8%**	**37.2%**	**36.8%**	**36.4%**	26.4%
			PGD	39.0%	37.8%	37.6%	37.8%	
		Vanilla	Random	55.2%	53.0%	51.2%	50.0%	
			Random+	55.6%	52.8%	50.6%	50.0%	-
			PGD	**50.6%**	**49.8%**	**49.4%**	**49.8%**	
Rotation and Brightness	$r \pm 30°$, $b \pm 20\%$	TSS	Random	**38.8%**	**38.0%**	37.2%	37.4%	
			Random+	39.0%	38.2%	**37.0%**	**36.8%**	26.8%
			PGD	39.6%	39.4%	38.6%	38.8%	
		Vanilla	Random	40.4%	35.4%	29.2%	22.4%	
			Random+	35.2%	31.2%	25.2%	**21.2%**	-
			PGD	**25.0%**	**23.2%**	**22.2%**	21.4%	
Scaling and Brightness	$s \pm 30\%$, $b \pm 30\%$	TSS	Random	40.2%	38.0%	**36.4%**	36.4%	
			Random+	38.0%	**37.0%**	36.6%	**36.0%**	23.4%
			PGD	**37.0%**	**37.0%**	36.6%	36.6%	
		Vanilla	Random	34.4%	26.2%	19.4%	16.0%	
			Random+	21.0%	**15.0%**	**12.4%**	**8.8%**	-
			PGD	**17.6%**	15.2%	13.8%	13.4%	
Rotation, Brightness, and ℓ_2	$r \pm 30°$, $b \pm 20\%$, $\|\delta\|_2 \leq .05$	TSS	Random	39.4%	38.2%	37.8%	37.0%	
			Random+	**38.2%**	**37.8%**	**36.6%**	**36.4%**	26.6%
			PGD	38.8%	38.8%	38.4%	38.0%	
		Vanilla	Random	26.0%	23.2%	19.8%	17.6%	
			Random+	21.4%	18.4%	16.0%	14.4%	-
			PGD	**16.6%**	**14.6%**	**14.2%**	**14.0%**	
Scaling, Brightness, and ℓ_2	$s \pm 30\%$, $b \pm 30\%$, $\|\delta\|_2 \leq .05$	TSS	Random	40.2%	38.2%	37.2%	36.0%	
			Random+	38.0%	**36.4%**	**35.8%**	**35.6%**	22.6%
			PGD	**36.8%**	**36.4%**	36.4%	36.0%	
		Vanilla	Random	24.4%	17.2%	11.4%	7.4%	
			Random+	13.8%	**8.4%**	**5.8%**	**4.8%**	-
			PGD	**9.8%**	8.8%	7.4%	7.4%	

From these three tables, we cross-validate the findings shown in the main paper: the adaptive attack decreases the empirical accuracy of **TSS** models slightly, while it decreases that of vanilla models more. Moreover, when comparing these three attacks, we find that with a small number of initial starts (e.g., $N = 10$), the PGD attack is typically the most powerful. However, with a large number of

initial starts (e.g., $N = 100$), Random+ attack sometimes becomes better. We conjecture that the optimization goal of PGD attack—maximization of cross-entropy loss—might be sub-optimal in terms of increasing the misclassification rate. Thus, with a small number of initial starts, PGD is better than Random/Random+ attack due to the iterative ascending. However, with a large number of initial starts, both PGD and Random+ attack can sufficiently explore the adversarial region, and PGD may be misled by the optimization goal to a benign region. It would be an interesting future work to study these intriguing properties of semantic attacks.

Table 11: Comparison of Empirical Accuracy for each corruption evaluated from the highest severity level (5) of CIFAR-10-C and ImageNet-C.

Corruption		CIFAR-10			ImageNet		
Category	Type	Vanilla	AugMix [23]	TSS	Vanilla	AugMix [23]	TSS
Weather	Snow	68.2%	**75.6%**	69.4%	16.0%	**22.6%**	13.8%
	Fog	63.4%	**65.4%**	62.0%	**24.0%**	22.2%	18.0%
	Frost	59.2%	67.8%	**73.8%**	21.6%	**24.8%**	22.6%
	Brightness	**82.4%**	**82.4%**	71.8%	**56.8%**	56.6%	35.8%
Blur	Zoom Blur	52.6%	70.8%	**75.2%**	21.4%	**31.0%**	20.4%
	Glass Blur	46.6%	50.2%	**72.2%**	8.0%	**14.0%**	13.8%
	Motion Blur	54.8%	68.6%	**70.2%**	14.2%	**25.2%**	11.4%
	Defocus Blur	49.0%	72.2%	**75.6%**	14.0%	22.6%	**25.6%**
Noise	Impulse Noise	29.8%	**51.0%**	46.2%	4.0%	9.8%	**12.0%**
	Gaussian Noise	34.8%	56.4%	**62.8%**	4.4%	9.6%	**12.8%**
	Shot Noise	43.0%	**63.4%**	62.6%	4.0%	13.0%	**14.0%**
Digital	Pixelate	42.0%	59.0%	**76.0%**	19.6%	39.2%	**55.6%**
	Elastic Transform	71.4%	65.2%	**74.4%**	14.8%	**23.6%**	**23.6%**
	Contrast	23.8%	26.0%	**49.8%**	4.2%	**11.6%**	5.0%
	JPEG Compression	70.8%	**73.0%**	71.8%	33.6%	**45.4%**	31.6%
Extra	Saturate	79.6%	**83.4%**	63.6%	41.6%	**43.4%**	25.8%
	Spatter	72.8%	**82.0%**	69.0%	22.4%	**30.6%**	17.6%
	Speckle Noise	45.2%	**64.0%**	58.8%	11.4%	**27.4%**	23.6%
	Gaussian Blur	34.6%	67.4%	**75.8%**	11.2%	15.2%	**33.0%**
	Average	53.89%	65.46%	**67.42%**	18.27%	**25.68%**	21.89%

K.8 Empirical Robustness against Unforeseen Attacks: Evaluation Protocol and Result Breakdown

In the main text (Section 7.2.4), we briefly show that **TSS** is generalizable to defend against unforeseen physical attacks by evaluating on CIFAR-10-C and ImageNet-C. Here, we first introduce the detailed evaluation protocol, then a breakdown of the result table (Table 3) in the main text and show empirical accuracy on each type of corruption.

K.8.1 Evaluated Models. On either CIFAR-10-C and ImageNet-C, we choose three models for evaluation: the vanilla model, the AugMix [23] trained model, and our **TSS** model for defending the composition of Gaussian blur, translation, brightness, and contrast. The vanilla models and **TSS** models are the same models as used in main experiments. The AugMix is the state-of-the-art empirical defense on the CIFAR-10-C and ImageNet-C dataset according to [9]. For AugMix, on CIFAR-10-C, since the model weights are not released, we use the official implementation of Aug-Mix (https://github.com/google-research/augmix) and extend the code to support our used model architecture (ResNet-110) for a fair comparison. We run the code with the suggested hyperparameters and achieve similar performance as reported in their paper. On ImageNet-C, we directly use the officially released model weights. The model has the same architecture (ResNet-50) as ours so the comparison is naturally fair. Note that all these models are trained only on the clean CIFAR-10 or ImageNet training set. We do not
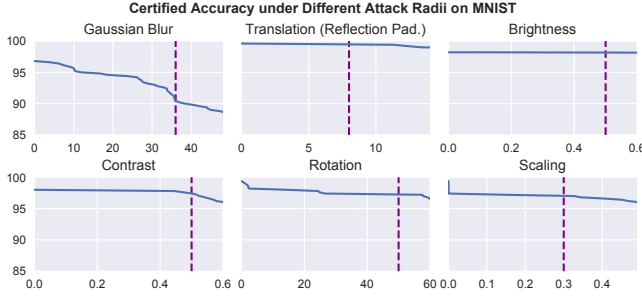
Figure 10: The blue curves show the certified robust accuracy on **MNIST**. The predefined certified radii are shown as purple vertical dotted lines. **No significant degradation after exceeding the predefined radii**. For Contrast subfigure, we allow additional 50% brightness change.
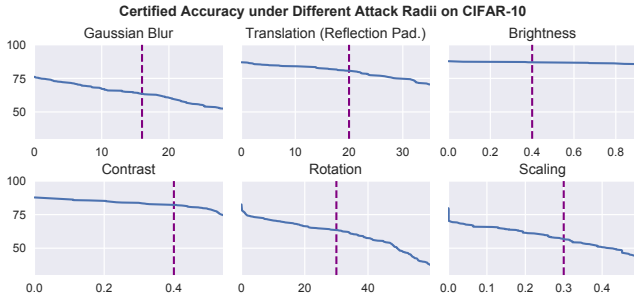

Figure 11: The blue curves show the certified robust accuracy on **CIFAR-10**. The predefined certified radii are shown as purple vertical dotted lines. **No significant degradation after exceeding the predefined certified radii**. For Contrast subfigure, we allow additional 40% brightness change.

Table 12: Empirical and certified robust accuracy on **MNIST** when attack radii go beyond the predefined one. The predefined certified radii are consistent with Table 2.

* For Contrast we allow additional ±50% brightness change since a single Contrast attack is not powerful enough.

				Beyond Predefined Radii		
Gaussian Blur		Radii:	36	40	44	48
	TSS	Empirical	91.2%	90.8%	90.4%	89.2%
		Certified	90.6%	90.0%	89.6%	88.8%
	Vanilla	Empirical	12.2%	11.8%	11.2%	11.2%
Translation (Reflection Pad.)		Radii:	8	10	12	14
	TSS	Empirical	99.6%	99.6%	99.6%	99.6%
		Certified	99.6%	99.6%	99.4%	99.0%
	Vanilla	Empirical	0.0%	0.0%	0.0%	0.0%
Brightness		Radii:	50%	52%	55%	60%
	TSS	Empirical	98.2%	98.2%	98.2%	98.2%
		Certified	98.2%	98.2%	98.2%	98.2%
	Vanilla	Empirical	96.6%	96.2%	95.6%	94.4%
Contrast*		Radii:	50%	52%	55%	60%
	TSS	Empirical	98.0%	98.0%	98.0%	98.0%
		Certified	97.6%	97.2%	96.8%	96.2%
	Vanilla	Empirical	93.2%	93.2%	93.2%	93.0%
Rotation		Radii:	50°	52°	55°	60°
	TSS	Empirical	98.2%	98.2%	98.2%	97.8%
		Certified	97.4%	97.4%	97.4%	96.6%
	Vanilla	Empirical	11.0%	9.8%	8.4%	7.2%
Scaling		Radii:	30%	35%	40%	50%
	TSS	Empirical	99.2%	98.8%	98.8%	98.6%
		Certified	97.2%	96.8%	96.8%	96.0%
	Vanilla	Empirical	89.2%	82.6%	72.8%	45.4%

include the evaluation of MNIST models since there is no corrupted MNIST dataset available within our best knowledge.

K.8.2 Empirical Accuracy Computation. We compute the **empirical accuracy** (on CIFAR-10-C/ImageNet-C) as the ratio of correctly predicted samples among the test samples, where the test samples are all corrupted at the highest severity level (level 5) to model the strongest unforeseen semantic attacker. For each corruption type, there is a full test set generated by 1-to-1 mapping from the original clean test set samples processed with the corruption. Being consistent with the main experiment's protocol, for each corruption type, we uniformly pick 500 samples from the corresponding test set. Then, we compute the average empirical accuracy among all 19 corruptions and report it in Table 3 in main text.

As a reference, we also include their certified accuracy against the composition of Gaussian blur, brightness, contrast, and translation. Since our **TSS** provides robustness certification only for smoothed models, we apply the same smoothing strategy as our **TSS** models, hoping for providing robustness certificates for baseline models. As shown in Table 3, only **TSS** models can be certified with nontrivial certified robust accuracy.

K.8.3 Breakdown. In Table 11, we show the breakdown of empirical accuracy for all models evaluated in Table 3. Note that our **TSS** models are trained using only four of these 19 corruptions (brightness, contrast, Gaussian blur, and additive Gaussian noise). Almost on all the corruptions, **TSS** has higher accuracy than vanilla models and sometimes higher than the state-of-the-art defense—AugMix. Interestingly, we find **TSS** models have different generalization abilities on these corruptions. The additive Gaussian noise has the best generalization ability, because **TSS** model also achieves much higher accuracy against impulse noise and shot noise than all the baselines. The Gaussian blur also generalizes well, because we can see significantly higher accuracy of **TSS** models against zoom blur, glass blur, motion blur, and defocus blur especially on CIFAR-10-C. Finally, brightness and contrast, even though they seem to be among the simplest transformations, have the poorest generalization ability. For example, under severe corruptions, the empirical accuracy of brightness is even below than that of vanilla models. Manual inspection of the corrupted images showed that corrupted brightness or contrast images are severely altered so that they are hard to be distinguished even by humans, giving a hint for possible reasons for the poor performance on these images. We thus conjecture that overly severe corruption could be the reason, and we think that it would be an interesting future direction to study these different generalization abilities in depth.

K.9 Certified Accuracy Beyond Certified Radii

Table 13: Empirical and certified robust accuracy on **CIFAR-10** when attack radii go beyond the predefined one. The predefined certified radii are consistent with Table 2.

* For Contrast we allow additional 40% brightness change since a single Contrast attack is not powerful enough.

				Beyond Predefined Radii		
		Radii:	16	20	24	28
Gaussian Blur	TSS	Empirical	65.8%	63.4%	61.2%	56.4%
		Certified	63.6%	60.8%	56.0%	52.6%
	Vanilla	Empirical	3.4%	3.2%	3.0%	2.8%
		Radii:	20	25	30	35
Tranlation (Reflection Pad.)	TSS	Empirical	86.0%	86.0%	85.8%	85.8%
		Certified	80.8%	77.4%	74.8%	70.6%
	Vanilla	Empirical	4.2%	3.6%	3.6%	2.8%
		Radii:	40%	45%	50%	55%
Brightness	TSS	Empirical	87.2%	87.0%	87.0%	87.0%
		Certified	87.0%	87.0%	87.0%	87.0%
	Vanilla	Empirical	42.6%	32.8%	21.2%	14.0%
		Radii	40%	45%	50%	55%
Contrast*	TSS	Empirical	85.8%	85.8%	85.4%	85.2%
		Certified	82.4%	80.8%	79.2%	71.8%
	Vanilla	Empirical	9.6%	7.8%	5.6%	4.8%
		Radii:	30°	40°	50°	60°
Rotation	TSS	Empirical	69.2%	64.0%	57.8%	46.8%
		Certified	63.6%	57.6%	48.2%	37.4%
	Vanilla	Empirical	21.4%	8.8%	5.0%	3.2%
		Radii:	30%	35%	40%	50%
Scaling	TSS	Empirical	67.0%	65.0%	60.6%	54.8%
		Certified	58.8%	53.6%	51.0%	43.4%
	Vanilla	Empirical	51.2%	43.0%	34.8%	21.2%

Table 14: Empirical and certified robust accuracy on **ImageNet** when attack radii go beyond the predefined one. The predefined certified radii are consistent with Table 2.

* For Contrast/Rotation/Scaling we allow additional ±40%/±20%/±30% brightness change since a single Contrast/Rotation/Scaling attack is not powerful enough.

				Beyond Predefined Radii	
		Radii:	36	40	44
Gaussian Blur	TSS	Empirical	52.6%	51.2%	49.8%
		Certified	51.6%	50.0%	48.8%
	Vanilla	Empirical	8.2%	7.2%	6.2%
		Radii:	100	105	110
Translation (Reflection Pad.)	TSS	Empirical	69.2%	69.2%	69.0%
		Certified	50.0%	49.4%	46.8%
	Vanilla	Empirical	46.2%	37.6%	36.6%
		Radii:	40%	45%	50%
Brightness	TSS	Empirical	70.4%	70.2%	70.0%
		Certified	70.0%	69.8%	69.6%
	Vanilla	Empirical	18.4%	10.0%	5.2%
		Radii:	40%	45%	50%
Contrast*	TSS	Empirical	68.4%	68.2%	67.6%
		Certified	61.4%	55.8%	45.0%
	Vanilla	Empirical	0.0%	0.0%	0.0%
		Radii:	30°	35°	45°
Rotation*	TSS	Empirical	36.8%	36.4%	33.4%
		Certified	26.8%	26.2%	21.8%
	Vanilla	Empirical	21.2%	19.4%	16.2%
		Radii:	30%	40%	50%
Scaling*	TSS	Empirical	36.0%	32.4%	26.6%
		Certified	23.4%	18.4%	11.6%
	Vanilla	Empirical	8.8%	8.8%	7.0%

In Figure 10 and Figure 11, on MNIST and CIFAR-10, the purple vertical dotted lines stand for the predefined certified radii that the models aim to defend, and the blue curves show the certified robust accuracy (y axis) with respect to attack radii (x axis). The figures imply that the **TSS** models that aim to defend against transformations within certain thresholds still maintain **high certified accuracy** when the transformation parameters go even far beyond the thresholds.

In Table 12, Table 13, and Table 14, we further list the empirical robust accuracy of **TSS** and vanilla models when the attacker goes beyond the predefined certified radius. The empirical robust accuracy is computed as the minimum among all three attacks: Random, Random+, and PGD. We observe that the empirical robust accuracy follows the same tendency. For example, on CIFAR-10 dataset, the **TSS** model is trained to defend against the rotation transformation within $30°$ where it achieves 69.2%/63.6% empirical/certified accuracy. When the rotation angle goes up to $60°$ the model still preserves 46.8%/37.4% empirical/certified accuracy. On the contrary, the vanilla model's empirical accuracy is reduced from 21.4% ($30°$ rotation) to 3.2% ($60°$ rotation).

Table 15: Study of the impact of different smoothing variance levels on certified robust accuracy and benign accuracy on **MNIST** for **TSS**. The attack radii are consistent with Table 2. The "Dist." refers to both training and smoothing distribution. The variance used in Table 2 is labeled in gray.

Transformation	Attack Radius	Certified Accuracy and Benign Accuracy under Different Variance Levels		
Gaussian Blur	$\alpha \leq 36$	Dist. of α	$\mathrm{Exp}(1/5)$	$\mathrm{Exp}(1/10)$	$\mathrm{Exp}(1/20)$
		Cert. Rob. Acc.	90.4%	**90.6%**	89.2%
		Benign Acc.	**97.0%**	96.8%	93.4%
Translation (Reflection Pad.)	$\sqrt{\Delta x^2 + \Delta y^2} \leq 8$	Dist. of $(\Delta x, \Delta y)$	$\mathcal{N}(0, 5^2 I)$	$\mathcal{N}(0, 10^2 I)$	$\mathcal{N}(0, 15^2 I)$
		Cert. Rob. Acc.	99.0%	**99.6%**	99.4%
		Benign Acc.	99.6%	**99.6%**	99.6%
Brightness	$b \pm 50\%$	Dist. of (c, b)	$\mathcal{N}(0, 0.5^2 I)$	$\mathcal{N}(0, 0.6^2 I)$	$\mathcal{N}(0, 0.7^2 I)$
		Cert. Rob. Acc.	98.4%	98.2%	98.4%
		Benign Acc.	98.4%	**98.4%**	98.4%
Contrast	$c \pm 50\%$	Dist. of (c, b)	$\mathcal{N}(0, 0.5^2 I)$	$\mathcal{N}(0, 0.6^2 I)$	$\mathcal{N}(0, 0.7^2 I)$
		Cert. Rob. Acc.	0.0%	98.0%	98.4%
		Benign Acc.	98.4%	**98.4%**	98.4%
Rotation	$r \pm 50°$	Dist. of ε	$\mathcal{N}(0, 0.05^2 I)$	$\mathcal{N}(0, 0.12^2 I)$	$\mathcal{N}(0, 0.20^2 I)$
		Cert. Rob. Acc.	97.6%	97.4%	97.6%
		Benign Acc.	99.2%	**99.4%**	99.2%
Scaling	$s \pm 30\%$	Dist. of ε	$\mathcal{N}(0, 0.05^2 I)$	$\mathcal{N}(0, 0.12^2 I)$	$\mathcal{N}(0, 0.20^2 I)$
		Cert. Rob. Acc.	96.6%	97.2%	96.0%
		Benign Acc.	**99.4%**	99.4%	99.0%

K.10 Different Smoothing Variance Levels: More Results

In Section 7.3.2 we have shown the study on smoothing variance levels on ImageNet (Table 4). Here, we further present our study of smoothing variance levels on MNIST and CIFAR-10. They are shown in Table 15 and Table 16 respectively. The smoothing variances shown in the two tables are for both training and inference-time smoothing. Except for smoothing variance, all other hyper-parameters for training and certification are kept the same and consistent with the main experiments (Table 2). As we can observe, the same conclusion still holds: usually, when the smoothing variance increases, the benign accuracy drops and the certified robust accuracy first rises and then drops. The reason is that larger smoothing variance makes the input more severely transformed so that the benign accuracy becomes smaller. On the other hand, larger

Table 16: Study of the impact of different smoothing variance levels on certified robust accuracy and benign accuracy on **CIFAR-10** for **TSS**. The attack radii are consistent with Table 2. The "Dist." refers to both training and smoothing distribution. The variance used in Table 2 is labeled in gray.

Transformation	Attack Radius	Certified Accuracy and Benign Accuracy under Different Variance Levels			
Gaussian Blur	$\alpha \leq 16$	Dist. of α	Exp(1/5)	Exp(1/10)	Exp(1/20)
		Cert. Rob. Acc.	**63.6%**	60.6%	53.0%
		Benign Acc.	**76.2%**	68.0%	57.4%
Translation (Reflection Pad.)	$\sqrt{\Delta x^2 + \Delta y^2} \leq 20$	Dist. of $(\Delta x, \Delta y)$	$\mathcal{N}(0, 10^2 I)$	$\mathcal{N}(0, 15^2 I)$	$\mathcal{N}(0, 20^2 I)$
		Cert. Rob. Acc.	76.2%	**80.8%**	74.4%
		Benign Acc.	**89.0%**	87.0%	84.6%
Brightness	$b \pm 40\%$	Dist. of (c, b)	$\mathcal{N}(0, 0.2^2 I)$	$\mathcal{N}(0, 0.3^2 I)$	$\mathcal{N}(0, 0.4^2 I)$
		Cert. Rob. Acc.	**87.4%**	87.0%	86.2%
		Benign Acc.	**87.8%**	87.8%	86.4%
Contrast	$c \pm 40\%$	Dist. of (c, b)	$\mathcal{N}(0, 0.2^2 I)$	$\mathcal{N}(0, 0.3^2 I)$	$\mathcal{N}(0, 0.4^2 I)$
		Cert. Rob. Acc.	0.0%	82.4%	82.4%
		Benign Acc.	**87.8%**	87.8%	86.4%
Rotation	$r \pm 30°$	Dist. of ε	$\mathcal{N}(0, 0.05^2 I)$	$\mathcal{N}(0, 0.09^2 I)$	$\mathcal{N}(0, 0.12^2 I)$
		Cert. Rob. Acc.	**63.6%**	62.0%	59.0%
		Benign Acc.	**82.0%**	78.6%	72.2%
Scaling	$s \pm 30\%$	Dist. of ε	$\mathcal{N}(0, 0.05^2 I)$	$\mathcal{N}(0, 0.09^2 I)$	$\mathcal{N}(0, 0.12^2 I)$
		Cert. Rob. Acc.	59.0%	**59.4%**	58.8%
		Benign Acc.	**85.4%**	81.6%	79.2%

smoothing variance makes the robustness easier to be certified as we can observe in various robustness conditions in Appendix D, where the required lower bound of p_A becomes smaller. This is the reason for the "first rise" on certified accuracy. However, when the smoothing variance becomes too large, the benign accuracy becomes too low, and according to our definition, the certified accuracy is upper bounded by the benign accuracy (precondition of robustness is correctness). This is the reason for the "then drop" on certified accuracy.

We again observe that the range of acceptable variance is usually wide. For example, on CIFAR-10, for rotation transformation, the certified robust accuracy is 63.6%/62.0%/59.0% across a wide range of smoothing variance: 0.05, 0.09, 0.12. Thus, even in the presence of such trade-off, without fine-tuning the smoothing variances, we can still obtain high certified robust accuracy and high benign accuracy as reported in Table 2 and Table 6 respectively.

We remark that the gray cells in Table 15 and Table 16 indicate the smoothing variances used in our main experiments. We did not tune the smoothing variances so these cells might be sub-optimal (though usually close to optimal) and they are just placed here for indication purpose.

K.11 Tightness-Efficiency Trade-Off

We notice that as we increase the number of samples when estimating the interpolation error in (24) and (27), the interpolation

error M_S and the upper bound $\sqrt{M} \geq M_S$ become smaller and the certification becomes tighter, leading to higher certified robust accuracy. However, the computation time is also increased, resulting in a trade-off between speed and accuracy. In Table 17 and Table 18, we illustrate this trade-off on two differentially resolvable transformations: composition of rotation and brightness on CIFAR-10, and composition of scaling and brightnes on MNIST. From the tables, we find that, for these compositions, as the sample numbers N and n increase, the interpolation error decreases and computing time increases (linearly with N and n). As a consequence, if using a large number of samples, we can decrease the smoothing noise level σ and achieve both higher certified accuracy and higher benign accuracy at the cost of larger computation time.

Table 17: Average interpolation upper bound $\sqrt{M}$ (24), average computation time, and "Certified accuracy (average certification time)" for varying number of samples and smoothing noise levels. Results on **CIFAR-10** against the composition of **rotation** $\pm 10°$ and **brightness** change $\pm 10\%$.

Number of Samples		Interpolation		Smoothing Noise Level σ		
First-Level	Second-Level	Avg. $\sqrt{M}$	Avg. Comp. Time	0.05	0.09	0.12
$N = 556$	$n = 2,000$	0.050	22.50 s	70.2% (62.32 s)	65.2% (86.60 s)	61.2% (53.73 s)
$N = 556$	$n = 200$	0.131	1.97 s	42.0% (490.21 s)	59.2% (93.19 s)	60.4% (86.60 s)
$N = 56$	$n = 2,000$	0.322	1.90 s	1.2% (6.18 s)	12.6% (16.64 s)	29.2% (25.77 s)
$N = 56$	$n = 200$	0.499	0.27 s	0.0% (5.22 s)	1.2% (5.68 s)	3.4% (8.49 s)
			Benign Accuracy:	83.0%	79.2%	79.6%

Table 18: Average interpolation upper bound $\sqrt{M}$ (24), average bound computation time, and "Certified robust accuracy (average certification time)" when using different number of samples and various smoothing noise levels. Data is collected on **MNIST** dataset against the composition of **scaling** $\pm 50\%$ and **brightness** change $\pm 50\%$.

Number of Samples		Interpolation		Smoothing Noise Level σ		
First-Level	Second-Level	Avg. $\sqrt{M}$	Avg. Comp. Time	0.05	0.09	0.12
$N = 2,500$	$n = 500$	0.064	10.52 s	97.2% (92.36 s)	97.4% (76.25 s)	96.6% (67.44 s)
$N = 2,500$	$n = 50$	0.163	0.90 s	18.8% (157.48 s)	97.0% (217.97 s)	95.0% (97.91 s)
$N = 250$	$n = 500$	0.441	0.74 s	0.0% (0.80 s)	6.0% (4.91 s)	16.2% (12.48 s)
$N = 250$	$n = 50$	0.641	0.13 s	0.0% (0.79 s)	0.0% (0.71 s)	0.6% (1.60 s)
			Benign Accuracy:	99.4%	99.6%	99.4%