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

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

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

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 ℓp 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 robustness 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 [19] and game playing [36, 46]. Despite all of these advances, ML systems are also found exceedingly vulnerable to adversarial attacks: image recognition systems can be adversarially misled [16, 48, 59], and malware detection systems can be evaded easily [52, 62].

The existing practice of security in ML has fallen into the cycle where new empirical defense techniques are proposed [31, 54], followed by new adaptive attacks breaking these defenses [1, 12, 16, 60]. 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...
These transformations are common and practical. For reasons about (1) function smoothing under different smoothings, based on the building block and scaling, based on the techniques for 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 smoothings 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 [7, 27, 63] that studies different smoothing distributions considers only $\ell_p$ perturbations.

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 [2, 13] 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 contributions:

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 [7] that has been proposed to certify the robustness against $\ell_p$ perturbations.
Semantic Transformation Based Attacks. Beyond adversarial $\ell_p$ 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 $\ell_p$ norm compared to $\ell_p$-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 [20, 22] and pose realistic threats for practical ML applications such as autonomous driving [38]. 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 [23, 55], 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 $\ell_p$ differences for the generated instances.

Randomized Smoothing. On a high level, randomized smoothing [7, 26, 27] 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 $\epsilon$ 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 $\ell_2$ 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 $\ell_p$ norm with a simultaneously high success rate of fooling the classifier [10, 63]. However, semantic transformations incur large $\ell_p$ perturbations, which renders classical randomized smoothing infeasible [3, 18, 25], 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 $X \subseteq \mathbb{R}^d$ and the set of labels as $Y = \{1, \ldots, C\}$ (where $C \geq 2$ is the number of classes). The set of transformation parameters is given by $Z \subseteq \mathbb{R}^m$ (e.g., rotation angles). We use the notation $p_X$ to denote the probability measure induced by the random variable $X$ and write $f_X$ for its probability density function. For a set $S$, we denote its probability by $p_X(S)$. A classifier is defined to be a deterministic function $h$ mapping inputs $x \in X$ to classes $y \in Y$. Formally, a classifier learns a conditional probability distribution $p(y|x)$ over labels and outputs the class that maximizes $p$, i.e., $h(x) = \arg\max_{y \in Y} p(y|x)$.

3.1 Threat Model and Certification Goal

Semantic Transformations. We model semantic transformations as deterministic functions $\phi : X \times Z \to X$, transforming an image $x \in X$ with a $Z$-valued parameter $\alpha$. For example, we use $\phi_R(x, \alpha)$ to model a rotation of the image $x$ by $\alpha$ 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 $\phi$ with itself as applying the same transformation just once, but with a different parameter, i.e., whether for any $\alpha, \beta \in Z$ there exists $y$ such that $\phi(x, \alpha) = \phi(x, \beta)$. 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 $\phi$ with parameter $\alpha$ to an input image $x \to \phi(x, \alpha)$. We allow the attacker to choose an arbitrary parameter $\alpha$ within a predefined (attack) parameter space $S$. For instance, a naive 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 $S_{adv} \subseteq Z$ such that, for a classifier $h$ and adversarial transformation $\phi$, we have

$$h(x) = h(\phi(x, \alpha)) \quad \forall \alpha \in S_{adv}. \quad (1)$$

Challenges for Certifying Semantic Transformations. Certifying ML robustness against semantic transformations is nontrivial and
We then consider two different types of transformation attacks. For work and show how it can be leveraged to certify semantic transformations such as rotations (Section 6) and propose a series of randomized smoothing, to provide robustness certifications under specific transformation dimensions. We then categorize transformations as either resolvable transformations (Section 5) or differentially resolvable transformations (Section 6). 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 (Transformation-Specific Smoothing-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 (\( \epsilon \)-Smoothed Classifier). Let \( \phi: X \times Z \rightarrow X \) be a transformation, \( \epsilon \sim \mathbb{P}_\varepsilon \) a random variable taking values in \( Z \) and let \( h: X \rightarrow Y \) be a base classifier. We define the \( \epsilon \)-smoothed classifier \( g: X \rightarrow Y \) as \( g(x; \epsilon) = \arg \max_{y \in Y} q(y | x; \epsilon) \) where \( q \) is given by the expectation with respect to the smoothing distribution \( \epsilon \), i.e.,

\[
q(y | x; \epsilon) = \mathbb{E}(p(y | \phi(x, \epsilon))).
\]

A key to certifying robustness against a specific transformation is the choice of transformation \( \phi \) 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 certification, 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 \( x \in X \) and a random variable \( \epsilon \) taking values in \( Z \), suppose that the base classifier \( h \) predicts \( \phi(x, \epsilon) \) to be of class \( y \) 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 \( \epsilon \)-smoothed classifier \( g \), i.e., we aim to find a set of perturbation parameters \( S_{\text{adv}} \) depending on \( p_A, p_B \), and smoothing parameter \( \epsilon \) such that, for all possible perturbation \( \alpha \in S_{\text{adv}} \), it is guaranteed that

\[
g(\phi(x, \alpha); \epsilon) = g(x; \epsilon)
\]

In other words, the prediction of the smoothed classifier can never be changed by applying the transformation \( \phi \) with parameters \( \alpha \) that are in the robust set \( S_{\text{adv}} \). 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 \( \epsilon_0 \sim \mathbb{P}_\varepsilon \) and \( \epsilon_1 \sim \mathbb{P}_\varepsilon \) be \( Z \)-valued random variables with probability density functions \( f_0 \) and \( f_1 \) with respect to a

![Figure 3: An overview of TSS.](image-url)
measure $\mu$ on $\mathcal{Z}$ and let $\phi: X \times \mathcal{Z} \to X$ be a semantic transformation. Suppose that $y_A = g(x; \epsilon_0)$ and let $p_A, p_B \in [0, 1]$ be bounds to the class probabilities, i.e.,

$$q(y_A | x, \epsilon_0) \geq p_A > p_B \geq \max_{y \neq y_A} q(y | x, \epsilon_0). \quad (4)$$

For $t \geq 0$, let $S_{t_0} := \{f_i | f_0 < t\}$ and $S_t := \{f_i | f_0 \leq t\}$ and define the function $\xi: [0, 1) \to [0, 1]$ by

$$\xi(p) := \sup\{P_1(S): S_{t_0} \subseteq S \subseteq S_t\}$$

where $t_0 := \inf\{t \geq 0: P_0(S_t) \geq p\}$. \quad (5)

Then, if the condition

$$\xi(p_A) + \xi(1 - p_B) > 1 \quad (6)$$

is satisfied, then it is guaranteed that $g(x; \epsilon_1) = g(x; \epsilon_0)$.

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 $\epsilon_0$. Based on the Neyman Pearson Lemma from statistical hypothesis testing, shifting $\epsilon_0 \to \epsilon_1$ results in bounds to the class probabilities associated with smoothing $x$ with $\epsilon_1$. 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 robustness condition $\xi(p_A) > 1 - \xi(1 - p_B)$. It is a more general version of what is proved by Cohen et al. [7], 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: Resolvable 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 [13, 63].

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 $\phi: X \times Z \to X$ is called resolvable if for any $\alpha \in Z$ there exists a resolving function $y_A: Z \to X$ that is injective, continuously differentiable, has non-vanishing Jacobian and for which

$$\phi(\phi(x, \alpha), \beta) = \phi(x, y_A(\beta)) \quad \forall x \in X, \beta \in Z. \quad (7)$$

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

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 $\phi$ in Theorem 1 is resolvable with resolving function $y_A$. Let $\alpha \in Z$ and set $\epsilon_1 := y_A(\alpha_0)$ in the definition of the function $\xi$. Then, if $\alpha$ satisfies condition (6), it is guaranteed that $g(\phi(x, \alpha); \epsilon_1) = g(x; \epsilon_0)$.

This corollary implies that for resolvable transformations, after we choose the smoothing distribution for the random variable $\epsilon_0$, we can infer the distribution of $\epsilon_1 = y_A(\alpha_0)$. Then, by plugging in $\epsilon_0$ and $\epsilon_1$ into Theorem 1, we can derive an explicit robustness condition from (6) such that for any $\alpha$ satisfying this condition, we can certify the robustness. In particular, for additive transformations we have $\epsilon_1 = y_A(\epsilon_0) = \alpha + \epsilon_0$. For common smoothing distributions $\epsilon_0$ 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

$$G_a(k) = \frac{1}{\sqrt{2\pi a}} \exp\left(-k^2/(2a)\right)$$

(8)

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

$$\phi_B(x, a) = x \ast G_a$$

(9)

and $\ast$ 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 $a, b \geq 0$, we have $\phi_B(\phi_B(x, a), b) = \phi_B(x, a + b)$.

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 $\phi_{BC}: X \times \mathbb{R} \to X$ as

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

(10)

where $k, b \in \mathbb{R}$ are contrast and brightness parameters, respectively. We remark that $\phi_{BC}$ is resolvable; however, it is not additive and applying Corollary 1 directly using the resolving function $y_B$ 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 $\epsilon_0 \sim \mathcal{N}(0, \text{diag}(\sigma^2, \tau^2))$,
Lemma 2. Let \( x \in \mathcal{X}, k \in \mathbb{R}, \epsilon_0 \sim \mathcal{N}(0, \text{diag}(\sigma^2, \tau^2)) \) and \( \epsilon_1 \sim \mathcal{N}(0, \text{diag}(\sigma^2, e^{-2\epsilon} \tau^2)) \). Suppose that \( q(y|x; \epsilon_0) \geq p \) for some \( p \in [0, 1] \) and \( y \in \mathcal{Y} \). Let \( \Phi \) be the cumulative density function of the standard Gaussian. Then
\[
q(y|x; \epsilon_1) \geq \begin{cases} 
2\Phi \left( e^{k-\frac{1}{2}} \left(1+\epsilon\right) \right) - 1 & \text{if } k \leq 0 \\
2 \left( \Phi \left( e^{k-\frac{1}{2}} \left(1-\epsilon\right) \right) - \Phi \right) & \text{if } k > 0.
\end{cases}
\]

Now suppose that \( g(\cdot; \epsilon_0) \) makes the prediction \( y_A \) at \( x \) with probability at least \( p_A \). Then, the preceding lemma tells us that the prediction confidence of \( g(\cdot; \epsilon_1) \) satisfies the lower bound (11) for the same class. Based on these confidence levels, we instantiate Theorem 1 with the random variables \( \epsilon_1 \) and \( \alpha + \epsilon_1 \) to get a robustness condition.

Lemma 3. Let \( \epsilon_0 \) and \( \epsilon_1 \) be as in Lemma 2 and suppose that
\[
q(y_A|x; \epsilon_1) \geq p_A > p_B \geq \max_{g \neq y_A} q(y|x; \epsilon_1).
\]
Then it is guaranteed that \( y_A = g(\theta_{BC}(x, \alpha); \epsilon_0) \) as long as \( \alpha = \langle k, b \rangle^T \) satisfies
\[
\sqrt{(k/\sigma)^2 + (b/(e^{-k} \tau))^2} < \frac{1}{2} \left( \Phi^{-1}(p_A) - \Phi^{-1}(p_B) \right).
\]

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 \( g(x; \epsilon_0) \).

5.1.3 Translation. Let \( \hat{\theta}_T \colon \mathcal{X} \times \mathbb{R}^2 \to \mathcal{X} \) be the transformation moving an image \( k \times k \) pixels to the right and \( k \times k \) pixels to the bottom with reflection padding. In order to handle continuous noise distributions, we define the translation transformation \( \hat{\theta}_T \colon \mathcal{X} \times \mathbb{R}^2 \to \mathcal{X} \) as \( \hat{\theta}_T(x, \alpha) = \hat{\theta}(x, \lfloor \alpha \rfloor) \) where \( \lfloor \cdot \rfloor \) denotes rounding to the nearest integer, applied element-wise. We note that \( \hat{\theta}_T \) 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 [35, 39].

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.

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 [7, 64], 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 \( \phi \) and with \( \epsilon_1 := \alpha + \epsilon_0 \) where \( \epsilon_0 \) is the smoothing distribution and \( \alpha \) is the transformation parameter. The robust radius is then derived by solving condition (6) for \( \alpha \).

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 \( g(\cdot; \epsilon) \) 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 $\phi_R$. As shown in Figure 5b, despite very similar, the image rotated by $30^\circ$ is different from the image rotated separately by $15^\circ$ and then again by $15^\circ$. 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

$$g(\phi_R(x, 15); r) = \arg\max_{y \in Y} \mathbb{E}(p(y) | \phi_R(x, 15), r)),$$  

which is a weighted average over the predictions of the base classifier on the randomly perturbed set $\{\phi_R(x, 15), \alpha \in \mathbb{Z}\}$. 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 $\{\phi_R(x, \alpha): \alpha \in \mathbb{Z}\}$), we need $\phi_R$ 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 $\phi$ to be differentially resolvable, if it can be written in terms of a resolvable transformation $\psi$ and a parameter mapping $\delta$.

**Definition 3 (Differentially resolvable transform).** Let $\phi: X \times Z_\phi \rightarrow X$ be a transformation with noise space $Z_\phi$ and let $\psi: X \times Z_\psi \rightarrow X$ be a resolvable transformation with noise space $Z_\psi$. We say that $\phi$ can be resolved by $\psi$ if for any $x \in X$ there exists function $\delta_x: Z_\phi \times Z_\psi \rightarrow Z_\psi$ such that for any $\alpha \in Z_\phi$ and any $\beta \in Z_\psi$,

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

This definition leaves open a certain degree of freedom with regard to the choice of resolvable transformation $\psi$. For example, we can choose the resolvable transformation corresponding to additive noise $(x, \delta) \rightarrow \psi(x, \delta) = x + \delta$, which lets us write any transformation $\phi$ 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 $\phi$ that can be resolved by $\psi$ and for transformation parameters from the set $S \subseteq Z_\phi$. To that end, we first sample a set of parameters $\{q_i\}_{i=1}^N \subseteq S$, and transform the input (with those sampled parameters) that yields $\phi(x, q_i)$. In the second step, we compute the class probabilities for each transformed input $\phi(x, q_i)$ with the classifier smoothed with the resolvable transformation $\psi$. Finally, the intuition is that, if every $q \in S$ is close enough to one of the sampled parameters, then the classifier is guaranteed to be robust against parameters from the set $S$. In the next theorem, we show the existence of such a "proximity set" for general $\delta_x$.

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

$$q(y_A | \phi(x, q_i); \varepsilon) \geq p_A^{(i)} \geq P_B^{(i)} \geq \max_{y \neq y_A} q(y | \phi(x, q_i); \varepsilon).$$  

Then there exists a set $\Delta^* \subseteq Z_\phi$ with the property that, for any $\alpha \in S$, $\exists \delta_x(\alpha, \alpha) \in \Delta^*$, then it is guaranteed that

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

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

The key of using this theorem for a specific transformation is to choose the reversible transformation \( \psi \) that can enable a tight calculation of \( \Lambda \) under a specific way of sampling \( \{a_t\}_{t=1}^N \). First, we observe that a large family of transformations including rotation and scaling can be resolved by the additive transformation \( \psi: X \times X \rightarrow X \) 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 \( \epsilon \sim \mathcal{N}(0, \sigma^2 I_d) \) as smoothing noise then leads to the condition that the maximum \( \ell_2 \)-interpolation error between the interval \( S = [a, b] \) (which is to be certified) and the sampled parameters \( a_t \) 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 \( \epsilon \sim \mathcal{N}(0, \sigma^2 I_d) \). Furthermore, let \( \phi \) be a transformation with parameters in \( \mathbb{Z}^N \subseteq \mathbb{R}^m \) and let \( S \subseteq \mathbb{Z}^2 \) and \( \{a_t\}_{t=1}^N \subseteq S \). Let \( y_A \in Y \) and suppose that for any \( i \), the \( \epsilon \)-smoothed classifier defined by \( q(y | x; \epsilon) : \equiv B(p(y | x + \epsilon)) \) has class probabilities that satisfy

\[
q(y_A | \phi(x, \alpha); \epsilon) \geq p_A^{(i)} \geq p_B^{(i)} \geq \max_{y \neq y_A} q(y | \phi(x, \alpha); \epsilon).
\]

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

\[
M_S := \max_{\alpha \in S} \min_{1 \leq i \leq N} \left\| \phi(x, \alpha) - \phi(x, a_i) \right\|_2
\]

satisfies

\[
M_S < R := \frac{\sigma}{2} \min_{1 \leq i \leq N} \left( p_A^{(i)} - p_B^{(i)} \right). \tag{20}
\]

In a nutshell, this corollary shows that if the smoothed classifier classifies all samples of transformed inputs \( \{\phi(x, \alpha_t)\}_{t=1}^N \) 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 \( \alpha \in S \).

The main challenge now lies in computing a tight and scalable upper bound \( M \geq M_S \). Given this bound, a set of transformation parameters \( S \) can then be certified by computing \( R \) in (20) and checking that \( R > M_S \). 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 \( M_S \) 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 + \( \ell_2 \) perturbations.

We remark that dealing with the interpolation error has already been tried before [2, 13]. 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).

### 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 \( S = [a, b] \) be an interval of transformation parameters that we wish to certify and let \( \{a_t\}_{t=1}^N \) be parameters dividing \( S \) uniformly, i.e.,

\[
a_i = a + (b - a) \cdot \frac{i - 1}{N - 1}, \quad i = 1, \ldots, N. \tag{21}
\]

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 \( g_i : [a, b] \rightarrow \mathbb{R} \geq 0 \) as

\[
a \mapsto g_i(a) := \left\| \phi(x, \alpha) - \phi(x, a_i) \right\|^2 \tag{22}
\]

corresponding to the squared \( \ell_2 \) interpolation error between the image \( x \) transformed with \( \alpha \) and \( a_i \), respectively. For each first-level interval \( [a_i, a_{i+1}] \) we look for an upper bound \( M_i \) such that

\[
M_i \geq \max_{\alpha \in [a_i, a_{i+1}]} \min_{1 \leq i \leq N} (g_i(a), g_{i+1}(a)). \tag{23}
\]

It is easy to see that \( \max_{1 \leq i \leq N-1} M_i \geq M_S^2 \) and hence setting

\[
\sqrt{M} = \max_{1 \leq i \leq N-1} \sqrt{M_i} \tag{24}
\]

is a valid upper bound to \( M_S \). The problem has thus reduced to computing the upper bounds \( M_i \) associated with each first-level interval \( [a_i, a_{i+1}] \). To that end, we now continue with a second-level sampling within the interval \( [a_i, a_{i+1}] \) for each \( i \). Namely, let \( \{\gamma_i\}_{j=1}^N \) be parameters dividing \( [a_i, a_{i+1}] \) uniformly, i.e.,

\[
\gamma_i = a_i + (a_{i+1} - a_i) \cdot \frac{j}{n - 1}, \quad j = 1, \ldots, n. \tag{25}
\]

Now, suppose that \( L \) is a global Lipschitz constant for all functions \( g_i \). By definition, for any \( 1 \leq i \leq N-1, L \) satisfies

\[
L \geq \max_{c, d \in [a_i, a_{i+1}]} \left\{ \max_{c, d \in [a_i, a_{i+1}]} \left\| \frac{g_i(c) - g_i(d)}{c - d} \right\| \right\}. \tag{26}
\]
An illustration of this bounding technique using stratified sampling is shown in Figure 6. We notice that, as the number of first-level samples is increased, the interpolation error $M_I$ becomes smaller by shrinking the sampling interval $[a_i, a_{i+1}]$; 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 $M \to M_\infty$, 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 $N \times n a_i$’s in order to achieve the same level of approximation accuracy. As a consequence, these $N \times n$ $a_i$’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 $d_{rs}$ to be the $L_2$-distance to the center of an image, i.e.

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

(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

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

(29)

**Definition 5 (Max-Color Extractor).** We define the operator that extracts the channel-wise maximum pixel value on a grid $S \subseteq \Omega$

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

$$\overline{m}(x, k, S) := \max_{(i, j) \in S} \max_{(r, s) \in G_{ij}} x_{k, r, s}.  

(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 $m_\Delta : \mathbb{R}^{K \times W \times H} \times [0, \ldots, K - 1] \times 2^\Omega \to \mathbb{R}$ with

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

(31)$$

**Rotation.** The rotation transformation is defined as rotating an image by an angle $\alpha$ 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 $\phi_R$ as the (raw) rotation $\hat{\phi}_R$ followed by interpolation and the aforementioned preprocessing step $P$ so that $\hat{\phi}_R = P \circ I \circ \phi_R$ 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 $\hat{\phi}_R = P \circ I \circ \phi_R$ 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 \leq i \leq N} \sum_{k=0}^{K-1} \sum_{(r, s) \in V} 2d_{rs} \cdot m_\Delta(x, k, \mathcal{P}_{r,s}^{(i)}) \cdot \overline{m}(x, k, \mathcal{P}_{r,s}^{(i)})  

(32)$$

where $V = \{(r, s) \in \mathbb{N}^2 | d_{rs} < \frac{1}{2} (\min \{W, H\} - 1)\}$. 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 $a_i$ to $a_{i+1}$.

**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 $O(3NR \cdot KWH)$, for both scaling and rotation. We note that $\mathcal{P}_{r,s}$ contains only a constant number of pixels since each interval $[a_i, a_{i+1}]$ 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.
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 Discussion

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 under any attacks is lower bounded by the certified accuracy. We compare with vanilla 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 under any attacks is lower bounded by the certified accuracy.

#### 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 $224 \times 224$ size [7, 24, 40, 63]. 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 [7] combined with consistency regularization [24] 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 $\ell_2$ perturbations [42]. 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 [7]. In line with related work [7, 24, 42, 63], 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 $p_A$ for resolvable transformations, and $n_s = 10^5$ samples to obtain each $p_A(i)$ 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 [2, 13]) 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 the notations: $\alpha$ for squared kernel radius for Gaussian blur; $\sqrt{\alpha x^2 + \frac{\alpha}{b}}$ 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 $\ell_2$ 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 [2], Interval [47], VeriVis [39], Semaify-NN [35], and DistSPT [13]. 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.

### 6.4 Discussion

We here 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 $\{\phi(x, a_i)\}_{i=1}^N$, 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 $\phi(x, a_i)$ 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 [7] 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 $\phi(x, a_i)$ if the check fails, the model is not certifiably robust and there is no need to proceed. Based on (1), for the current $\phi(x, a_i)$, 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 $\phi(x, a_{i+1})$. Based on (2), we terminate early if the check $R > M_S$ for the current $\phi(x, a_i)$ 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 $\ell_p$-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.
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**.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Gaussian Blur</td>
<td>Resolvable</td>
<td>MNIST</td>
<td>$\sigma \leq 36$</td>
<td>99.6%</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CIFAR-10</td>
<td>$\sigma \leq 36$</td>
<td>93.6%</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ImageNet</td>
<td>$\sigma \leq 36$</td>
<td>93.6%</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Translation (Reflection Pad)</td>
<td>Resolvable, Discrete</td>
<td>MNIST</td>
<td>$\sqrt{\Delta x^2 + \Delta y^2} \leq 8$</td>
<td>99.6%</td>
<td>-</td>
<td>-</td>
<td>98.8%</td>
<td>-</td>
<td>98.8%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CIFAR-10</td>
<td>$\sqrt{\Delta x^2 + \Delta y^2} \leq 20$</td>
<td>90.8%</td>
<td>-</td>
<td>-</td>
<td>65.0%</td>
<td>-</td>
<td>65.0%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ImageNet</td>
<td>$\sqrt{\Delta x^2 + \Delta y^2} \leq 100$</td>
<td>50.0%</td>
<td>-</td>
<td>-</td>
<td>43.2%</td>
<td>-</td>
<td>43.2%</td>
</tr>
<tr>
<td>Brightness</td>
<td>Resolvable</td>
<td>MNIST</td>
<td>$h \leq 50%$</td>
<td>98.2%</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CIFAR-10</td>
<td>$b \leq 40%$</td>
<td>87.0%</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ImageNet</td>
<td>$b \leq 40%$</td>
<td>70.0%</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Contrast and Brightness</td>
<td>Resolvable, Composition</td>
<td>MNIST</td>
<td>$c \leq 50%$, $b \leq 50%$</td>
<td>97.6%</td>
<td>$ \leq 4%$</td>
<td>$0%$</td>
<td>$0%$</td>
<td>$\leq 74%$</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CIFAR-10</td>
<td>$c \leq 40%$, $b \leq 40%$</td>
<td>82.4%</td>
<td>$0%$</td>
<td>$0%$</td>
<td>-</td>
<td>-</td>
<td>86.0%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ImageNet</td>
<td>$c \leq 40%$, $b \leq 40%$</td>
<td>61.4%</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>68.4%</td>
</tr>
<tr>
<td>Gaussian Blur, Translation, Brightness, and Contrast</td>
<td>Resolvable, Composition</td>
<td>MNIST</td>
<td>$c \leq 50%$, $b \leq 50%$, $\sigma \leq 10%$</td>
<td>98.0%</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>97.2%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CIFAR-10</td>
<td>$c \leq 40%$, $b \leq 40%$, $\sigma \leq 10%$</td>
<td>88.2%</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>67.6%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ImageNet</td>
<td>$c \leq 40%$, $b \leq 40%$, $\sigma \leq 10%$</td>
<td>52.8%</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>48.8%</td>
</tr>
<tr>
<td>Rotation</td>
<td>Differentially Resolvable</td>
<td>MNIST</td>
<td>$r \leq 50%$</td>
<td>97.4%</td>
<td>$ \leq 8%$</td>
<td>$0%$</td>
<td>$0%$</td>
<td>$\leq 92.48%$</td>
<td>98.4%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CIFAR-10</td>
<td>$r \leq 40%$</td>
<td>94.6%</td>
<td>$62.5%$</td>
<td>$20.2%$</td>
<td>-</td>
<td>-</td>
<td>87.6%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ImageNet</td>
<td>$r \leq 40%$</td>
<td>63.6%</td>
<td>$10.6%$</td>
<td>$0%$</td>
<td>-</td>
<td>$\leq 49.37%$</td>
<td>69.2%</td>
</tr>
<tr>
<td>Scaling</td>
<td>Differentially Resolvable</td>
<td>MNIST</td>
<td>$s \leq 30%$</td>
<td>97.2%</td>
<td>$83.0%$</td>
<td>$16.4%$</td>
<td>-</td>
<td>-</td>
<td>99.2%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CIFAR-10</td>
<td>$s \leq 30%$</td>
<td>86.8%</td>
<td>$0.4%$</td>
<td>$0.0%$</td>
<td>-</td>
<td>-</td>
<td>96.2%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ImageNet</td>
<td>$s \leq 30%$</td>
<td>26.4%</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>37.4%</td>
</tr>
<tr>
<td>Rotation and Brightness</td>
<td>Differentially Resolvable, Composition</td>
<td>MNIST</td>
<td>$r \leq 50%$, $b \leq 20%$</td>
<td>96.8%</td>
<td>-</td>
<td>-</td>
<td>98.8%</td>
<td>-</td>
<td>66.8%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CIFAR-10</td>
<td>$r \leq 40%$, $b \leq 10%$</td>
<td>76.6%</td>
<td>$59.4%$</td>
<td>$20.0%$</td>
<td>-</td>
<td>-</td>
<td>76.6%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ImageNet</td>
<td>$r \leq 40%$, $b \leq 10%$</td>
<td>61.4%</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>68.8%</td>
</tr>
<tr>
<td>Scaling, and Brightness</td>
<td>Differentially Resolvable, Composition</td>
<td>MNIST</td>
<td>$s \leq 50%$, $b \leq 50%$</td>
<td>96.6%</td>
<td>-</td>
<td>-</td>
<td>97.8%</td>
<td>-</td>
<td>70.7%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CIFAR-10</td>
<td>$s \leq 30%$, $b \leq 30%$</td>
<td>54.2%</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>36.8%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ImageNet</td>
<td>$s \leq 30%$, $b \leq 30%$</td>
<td>43.4%</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>37.4%</td>
</tr>
<tr>
<td>Rotation, Brightness, and $\ell_2$</td>
<td>Differentially Resolvable, Composition</td>
<td>MNIST</td>
<td>$r \leq 50%$, $b \leq 20%$, $|\ell_2| \leq 5%$</td>
<td>96.6%</td>
<td>-</td>
<td>-</td>
<td>97.6%</td>
<td>-</td>
<td>66.8%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CIFAR-10</td>
<td>$r \leq 40%$, $b \leq 10%$, $|\ell_2| \leq 5%$</td>
<td>64.2%</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>71.6%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ImageNet</td>
<td>$r \leq 40%$, $b \leq 10%$, $|\ell_2| \leq 5%$</td>
<td>55.2%</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>65.2%</td>
</tr>
<tr>
<td>Scaling, Brightness, and $\ell_2$</td>
<td>Differentially Resolvable, Composition</td>
<td>MNIST</td>
<td>$s \leq 50%$, $b \leq 50%$, $|\ell_2| \leq 5%$</td>
<td>96.6%</td>
<td>-</td>
<td>-</td>
<td>97.6%</td>
<td>-</td>
<td>66.0%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CIFAR-10</td>
<td>$s \leq 30%$, $b \leq 30%$, $|\ell_2| \leq 5%$</td>
<td>51.2%</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>65.0%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ImageNet</td>
<td>$s \leq 30%$, $b \leq 30%$, $|\ell_2| \leq 5%$</td>
<td>22.6%</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>36.0%</td>
</tr>
</tbody>
</table>

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 non-trivial certified robustness against certain semantic attacks. Note that DistSPT [13] 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 [13, 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 $\sim 2\%$ on MNIST to $\sim 10\%$ - $20\%$ 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 [20]. 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 + contrast + translation + rotation, to defend against these corruptions. We select two baselines: vanilla models and AugMix [21]. AugMix is the state of the art model on CIFAR-10-C and ImageNet-C [8].

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.

### 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 $\pm 40\%$ 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

---

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).

<table>
<thead>
<tr>
<th>Transformation</th>
<th>Attack Radii</th>
<th>Certified Accuracy %</th>
<th>Model</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>Vanilla</td>
</tr>
<tr>
<td>Gaussian Blur</td>
<td>$\pm 0%$</td>
<td>6%</td>
<td>58.2%</td>
</tr>
<tr>
<td>Translation</td>
<td>$\pm \alpha^2$</td>
<td>$\pm \beta^2$</td>
<td></td>
</tr>
<tr>
<td>Rotation</td>
<td>$\pm \gamma^2$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Scaling</td>
<td>$\pm \delta^2$</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

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.

<table>
<thead>
<tr>
<th>Transformation</th>
<th>Attack Radii</th>
<th>Certified Accuracy %</th>
<th>Model</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>Vanilla</td>
</tr>
<tr>
<td>Gaussian Blur</td>
<td>$\pm 0%$</td>
<td>51.6%</td>
<td>48.8%</td>
</tr>
<tr>
<td>Translation</td>
<td>$\pm 0.02$</td>
<td>59.2%</td>
<td>55.4%</td>
</tr>
<tr>
<td>Brightness</td>
<td>$\pm 0.2$</td>
<td>70.0%</td>
<td>68.0%</td>
</tr>
<tr>
<td>Contrast</td>
<td>$\pm 2$</td>
<td>72.2%</td>
<td>69.4%</td>
</tr>
<tr>
<td>Rotation</td>
<td>$\pm 1$</td>
<td>75.6%</td>
<td>75.6%</td>
</tr>
<tr>
<td>Scaling</td>
<td>$\pm 2$</td>
<td>78.8%</td>
<td>80.8%</td>
</tr>
</tbody>
</table>
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 in 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 [7, 63]. 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 $\ell_p$ perturbations. Since the studies of adversarial vulnerability of neural networks [16, 48], there has emerged a rich body of research on evasion attacks (e.g., [1, 5, 53, 59]) and empirical defenses (e.g., [32, 44, 45]). To provide robustness certification, different robustness training and verification approaches have been proposed. In particular, interval bound propagation [17, 66], linear relaxations [33, 56–58, 61], and semidefinite programming [9, 41] have been applied to certify NN robustness. Recently, robustness certification based on randomized smoothing has shown to be scalable and with tight guarantees [7, 26, 27]. With improvements on optimizing the smoothing distribution [10, 49, 63] and better training mechanisms [6, 24, 42, 64], the verified robustness of randomized smoothing is further improved. A recent survey summarizes certified robustness approaches [28].

Semantic Attacks for Neural Networks. Recent work has shown that semantic transformations are able to mislead ML models [14, 22, 60]. 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 $\ell_p$-bounded perturbations [11]. Brightness/contrast attacks can achieve 91.6% attack success on CIFAR-10, and 71%-100% attack success rate on ImageNet [20]. 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 [38]. Empirical defenses against semantic transformations have been investigated [11, 20].

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 [39] can handle only discrete transformations. Interval bound propagation has been used to certify common semantic transformations [2, 13, 47]. To tighten the interval bounds, linear relaxations are introduced. DeepG [2] optimizes linear relaxations for given semantic transformations, and Semantify-NN [35] encodes semantic transformations by neural networks and applies linear relaxations for NNs [56, 66]. However, linear relaxations are loose and computationally intensive compared to our TSS. Recently, Fischer et al [13] 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, sets 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.

ACKNOWLEDGMENTS

This work was performed under the auspices of the U.S. Department of Energy by the Lawrence Livermore National Laboratory under Contract No. DE-AC52-07NA27344, Lawrence Livermore National Security, LLC. The views and opinions of the authors expressed herein do not necessarily state or reflect those of the United States Government or Lawrence Livermore National Security, LLC, and shall not be used for advertising or product endorsement purposes. This work was supported by LLNL Laboratory Directed Research and Development project 20-ER-014 and released with LLNL tracking number LLNL-CONF-822465. This work is supported in part by NSF under grant no. CNS-2046726, CCF-1910100, CCF-1816615, and 2020 Amazon Research Award.

CZ and the DS3Lab gratefully acknowledge the support from the Swiss National Science Foundation (Project Number 200021_184628, and 197485), Insosuisse/SNF BRIDGE Discovery (Project Number 40B2_0_187132), European Union Horizon 2020 Research and Innovation Programme (DAPHNE, 957407), Botnar Research Centre for Child Health, Swiss Data Science Center, Alibaba, Cisco, eBay, Google Focused Research Awards, Kuaishou Inc., Oracle Labs, Zurich Insurance, and the Department of Computer Science at ETH Zurich.

The authors thank the anonymous reviewers for valuable feedback, Adel Bibi (University of Oxford) for pointing out a bug in the initial implementation, and the authors of [13], especially Marc Fischer and Martin Vechev, for insightful discussions and support on experimental evaluation.


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. Appendix I proves our supporting theorems for computing the certification generally follows the same procedure as in certification procedure of $S_{\text{adv}} = \{(k, b) | k \in [-k_0, k_0], b \in [-b_0, b_0]\}$. (35) which stands for any contrast change within $e^b_0$ and brightness change within $b_0$. It is infeasible to check every $(k, b) \in S_{\text{adv}}$. To mitigate this, we relax the robustness condition in Lemma 3 from (36) to (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 $\Phi^{-1}(\hat{p}_B)$ 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 $\phi_{\text{BTC}}$ is defined as:

$$\phi_{\text{BTC}}(x, \alpha) := \phi_B(\phi_T(\phi_{\text{BC}}(x, a, k, b), \alpha_{tx}, \alpha_{ty}, \alpha_{b}), a, k, b).$$ (38)

where $\phi_B$, $\phi_T$ and $\phi_{\text{BC}}$ are Gaussian blur, translation, and brightness and contrast transformations respectively as defined before; $\alpha := (a, k, b, \alpha_{tx}, \alpha_{ty}, \alpha_{b})^T \in \mathbb{R}^6 \times \mathbb{R}^2 \times \mathbb{R}$ 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 $x \in X$, $k \in \mathbb{R}$ and let $\epsilon_0 := (\epsilon_{a0}^b, 1)^T$ be a random variable defined as

$$\epsilon_{a0}^b \sim N(0, \text{diag}(\sigma_a^2, \sigma_b^2, \sigma_{tx}^2, \sigma_{ty}^2)) \text{ and } \epsilon_0^b \sim \text{Exp}(\lambda_B).$$ (39)
Similarly, let \( \epsilon_1 := (\epsilon_1^1, \epsilon_1^2) \) be a random variable with
\[
\epsilon_1^1 \sim \mathcal{N}(0, \text{diag} \sigma_k^2, e^{-2\kappa^2}, \sigma_b^2, 2)) \text{ and } \epsilon_1^2 \sim \text{Exp}(\lambda_B).
\]
(40)
For either random variable (denoted as \( \epsilon \)), recall that \( q(y; x; \epsilon) := \mathbb{E}(p(y|\phi_{\text{HBC}}(x, \epsilon))) \).
Suppose that \( q(y; x; \epsilon) \geq p \) for some \( p \in [0, 1] \) and \( y \in \mathcal{Y} \).
Then \( q(y; x; \epsilon_1) \) satisfies Eq. (11).

**Lemma B.1.** Let \( \epsilon_0 \) and \( \epsilon_1 \) be as in Corollary 3 and suppose that
\[
q(y_A; x; \epsilon_1) \geq \hat{p}_A > \hat{p}_B \geq \max_{y\neq y_A} q(y; x; \epsilon_1).
\]
Then it is guaranteed that \( y_A = g(\phi_{\text{HBC}}(x, \alpha); \epsilon_0) \) as long as \( p_A' > p_B' \), where
\[
p_A' = \begin{cases} 
0, & \text{if } \hat{p}_A \leq 1 - \exp(-\lambda_B \sigma_B), \\
\Phi^{-1}(1 - (1 - \hat{p}_A) \exp(\lambda_B \sigma_B)) - \frac{\alpha_b^2 + \alpha_y^2 + (\alpha^2_T + \alpha^2_y)/\sigma^2_T}{\alpha_k^2/\sigma^2_k + e^{-2\kappa^2} \sigma^2_b + (\alpha^2_T + \alpha^2_y)/\sigma^2_T} \text{ otherwise},
\end{cases}
\]
and
\[
p_B' = \begin{cases} 
1, & \text{if } \hat{p}_B \geq \exp(-\lambda_B \sigma_B), \\
1 - \Phi^{-1}(1 - \hat{p}_B \exp(\lambda_B \sigma_B)) - \frac{\alpha_b^2 + \alpha_y^2 + (\alpha^2_T + \alpha^2_y)/\sigma^2_T}{\alpha_k^2/\sigma^2_k + e^{-2\kappa^2} \sigma^2_b + (\alpha^2_T + \alpha^2_y)/\sigma^2_T} \text{ otherwise},
\end{cases}
\]
(42)
(43)

The \( \epsilon_0 \) specified by (39) is the smoothing distribution. Similar as in brightness and contrast certification, we first obtain \( \hat{p}_A \), a lower bound of \( q(y_A; x, \epsilon_0) \) by Monte-Carlo sampling. For a given transformation parameter \( \alpha := (\alpha_k, \alpha_b, \alpha_T, \alpha_y, \alpha_B)^T \), we then trigger Corollary 3 to get \( \hat{p}_A \), a lower bound of \( q(y_A; x, \epsilon_1) \) and set \( \hat{p}_B = 1 - \hat{p}_A \).
Finally, we use the explicit condition in Lemma B.1 to obtain the certification. Indeed, with \( \hat{p}_B = 1 - \hat{p}_A \), Lemma B.1 can be simplified to the following corollary.

**Corollary 4.** Let \( \epsilon_0 \) and \( \epsilon_1 \) be as in Corollary 3 and suppose that
\[
q(y_A; x; \epsilon_1) \geq \hat{p}_A.
\]
Then it is guaranteed that \( y_A = g(\phi_{\text{HBC}}(x, \alpha); \epsilon_0) \) as long as
\[
\hat{p}_A > 1 - \exp(-\lambda_B \sigma_B)
\]
(44)
(45)
(46)

To certify against a set of transformation parameters
\[
S_{\text{adv}} := \{(\alpha_k, \alpha_b, \alpha_T, \alpha_y, \alpha_B)^T | \alpha_k \in [-k_0, k_0], \alpha_b \in [-b_0, b_0], \| (\alpha_T, \alpha_y) \|_2 \leq T, \alpha_B \leq B_0\},
\]
we relax the robust condition in (45) to
\[
\hat{p}_A > 1 - \exp(-\lambda_B \sigma_B)
\]
(47)
The LHS of Equation (47) is monotonically decreasing w.r.t. \( |\alpha_k| \) and the RHS is monotonically increasing w.r.t. \( |\alpha_b|, |\alpha_y|, \| (\alpha_T, \alpha_y) \|_2 \).
and \( |\alpha_B|, \) and the RIS is symmetric w.r.t. \( \alpha_B \) and \( \| (\alpha_T, \alpha_y) \|_2 \).
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 \( S_{\text{adv}} \).
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 \( M_S \).
We only make the following two changes: (1) alter the smoothing distribution from additive Gaussian noise \( \psi(x, \delta) = x + \delta \) where \( \delta \sim N(0, \sigma^2 \mathbb{I}_d) \) to additive Gaussian noise and Gaussian brightness change \( \psi(x, \delta, b) = x + \delta + b \cdot \mathbb{I}_d \) where \( \delta \sim N(0, \sigma^2 \mathbb{I}_d), b \sim N(0, \sigma^2_b) \); (2) change the robust condition from \( R > M_S \) in Corollary 2 to \( R > \sqrt{M^2_S + (\sigma_b^2/\sigma^2_B)} \).
We formalize this robust condition in the following corollary, and the proof is entailed in Appendix G.

**Corollary 5.** Let \( \psi_B(x, \delta, b) = x + \delta + b \cdot \mathbb{I}_d \) and let \( \epsilon \sim N(0, \sigma^2 \mathbb{I}_d), \epsilon_B \sim N(0, \sigma^2_B) \).
Furthermore, let \( \phi \) be a transformation in \( \mathbb{Z}_\phi \subseteq \mathbb{R}^m \) and let \( S \subseteq \mathbb{Z}_\phi \) and \( (a_i)_{i=1}^N \subseteq S \).
Let \( y_A \in \mathcal{Y} \) and suppose that for any \( i \), the \( (\epsilon_i, \epsilon_B) \)-smoothed classifier \( \tilde{q}(y; \epsilon, \epsilon_B) := \mathbb{E}(p(y|\psi(x, \epsilon, \epsilon_B)) \) satisfies
\[
q(y_A; x; \epsilon, \epsilon_B) \geq \tilde{p}_A^{(i)} > \tilde{p}_B^{(i)} \geq \max_{y\neq y_A} q(y; x; \epsilon, \epsilon_B).
\]
(48)
for each \( i \). Let
\[
R := \sigma \min_{1 \leq i \leq N} \left( \Phi^{-1} \left( \tilde{p}_A^{(i)} \right) - \Phi^{-1} \left( \tilde{p}_B^{(i)} \right) \right)
\]
(49)
Then, \( \forall \alpha \in S \) and \( \forall b \in [-b_0, b_0] \) it is guaranteed that \( y_A = \arg \max_y q(y; \phi(x, \alpha) + b \cdot \mathbb{I}_d; \epsilon, \epsilon_B) \) as long as
\[
R > \sqrt{M^2_S + (\sigma_b^2/\sigma^2_B)}.
\]
(50)

**B.4 Scaling/Rotation, Brightness, and \( \ell_2 \) 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 \( \ell_2 \)-bounded perturbations—we only need to change the robust condition from \( R > \sqrt{M^2_S + (\sigma_b^2/\sigma^2_B)} \) to \( R > \sqrt{(M_S + r)^2 + (\sigma_b^2/\sigma^2_B)} \).
The proof is given in Appendix G.

**Corollary 6.** Under the same setting as in Corollary 5, for \( \forall \alpha \in S \), \( \forall b \in [-b_0, b_0] \), and \( \forall \delta \in \mathbb{R}^d \), such that \( \| \delta \|_2 \leq r \), it is guaranteed that \( y_A = \arg \max_y q(y; \phi(x, \alpha) + b \cdot \mathbb{I}_d + \delta; \epsilon, \epsilon_B) \) as long as
\[
R > \sqrt{(M_S + r)^2 + (\sigma_b^2/\sigma^2_B)}.
\]
(51)
where \( M_S \) 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 $Z$, 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 $\gamma_a$. Next, we choose a smoothing distribution, i.e., the distribution of the random variable $r_0$, and identify the distribution of $e_1 = \gamma_a(r_0)$. 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 $Z$ 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: $Z := Z_{no-resolve} \times Z_{resolve}$. 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 $t_0$ difference as we did for rotation and scaling or other regimes. Specifically, our Lemma 8 shown in Appendix I is a useful tool to bound $t_0$ 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 [21, 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 $\phi : X \times Z \to X$ be a transformation, $\epsilon \sim P_\epsilon$ a random variable taking values in $Z$ and let $h : X \to Y$ be a base classifier. We define the $\epsilon$-smoothed classifier $g : X \to Y$ as $g(x; \epsilon) = \arg \max_{y \in Y} q(y|x; \epsilon)$ where $q$ is given by the expectation with respect to the smoothing distribution $\epsilon$, i.e.,

$$q(y|x; \epsilon) := \mathbb{E}[p(y|\phi(x, \epsilon))].$$

(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 $r_0 \sim P_0, e_1 \sim P_1$ be $Z$-valued random variables with probability density functions $f_0$ and $f_1$ with respect to a measure $\mu$. For $t \geq 0$ we define lower and strict lower level sets as

$$S_t := \{ z \in Z : \Lambda(z) < t \}, \quad S_t^\sim := \{ z \in Z : \Lambda(z) \leq t \},$$

where $\Lambda(z) := \frac{f_1(z)}{f_0(z)}$. (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 $\epsilon \in X$, $y_A \in Y$ and $p_A, p_B \in [0, 1]$ with $p_A > p_B$. We say that the $\epsilon$-smoothed classifier $q$ is $(p_A, p_B)$-confident at $x$ if

$$q(y_A|x; \epsilon) \geq p_A \geq p_B \geq \max_{y \neq y_A} q(y|x; \epsilon).$$

(54)

C.1 Auxiliary Lemmas

**Lemma 6.** Let $e_0$ and $e_2$ be random variables taking values in $Z$ and with probability density functions $f_0$ and $f_1$ with respect to a measure $\mu$. Denote by $\Lambda$ the likelihood ratio $\Lambda(z) = f_1(z)/f_0(z)$. For $p \in [0, 1]$ let $t_p := \inf \{ t \geq 0 : P_0(S_t) \geq p \}$. Then, it holds that

$$P_0(S_{t_p}) \leq p \leq P_0(S_{t_p}).$$

(55)

**Proof.** We first show the RHS of inequality (55). This follows directly from the definition of $t_p$ if we show that the function $t \mapsto P_0(S_t)$ 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 $z \in S_t$, then $\forall n: \Lambda(z) \leq t \leq t_n$, thus $z \in \cap_n A_n$ and hence $S_t \subseteq \cap_n A_n$. If on the other hand $z \in \cap_n A_n$, then $\forall n: \Lambda(z) \leq t_n \rightarrow t$ as $n \rightarrow \infty$ and thus $z \in S_t$, yielding $S_t = \cap_n A_n$. Hence for any $t \geq 0$ we have that

$$\lim_{n \rightarrow \infty} P_0(A_n) = P_0(\cap_n A_n) = P_0(S_t).$$

(56)

Thus, the function $t \mapsto P_0(S_t)$ is right continuous and in particular it follows that $P_0(S_{t_p}) \geq p$. We now show the LHS of inequality (55). Consider the sets $B_n := \{ z : \Lambda(z) < t_p - 1/n \}$ and note that $B_n \subseteq B_{n+1}$. Clearly, if $z \in \cup_n B_n$, then $\exists n$ such that $\Lambda(z) < t_p - 1/n < t_p$ and thus $z \in S_{t_p}$. If on the other hand $z \in S_{t_p}$, then we can choose $n$ large enough such that $\Lambda(z) < t_p - 1/n$ and thus $z \in \cup_n B_n$ yielding $S_{t_p} = \cup_n B_n$. Furthermore, by the definition of $t_p$ and since for any $n \in \mathbb{N}$ we have that $P_0(B_n) = P_0(S_{t_p} - 1/n)) < p$ it follows that

$$P_0(S_{t_p}) = P_0 \left( \bigcup_n B_n \right) \leq \lim_{n \rightarrow \infty} P_0(B_n) \leq p.$$

(57)

concluding the proof.

**Lemma 7.** Let $e_0$ and $e_2$ be random variables taking values in $Z$ and with probability density functions $f_0$ and $f_1$ with respect to a measure $\mu$. Let $h : Z \to [0, 1]$ be a deterministic function. Then, for any $t \geq 0$ the following implications hold:

$$
\square$$
The inequality in (63) follows from the fact that whenever $\delta \in |S|$ and $x \in A$ we have that $\mathbb{E}[h(x)] \leq P_0(S) \Rightarrow \mathbb{E}[h(x)] \leq P_1(S)$. (58)

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

$$\mathbb{E}[h(x)] \leq P_0(S) \Rightarrow \mathbb{E}[h(x)] \leq P_1(S).$$

(ii) For any $S \subseteq Z$ with $S \subseteq S \subseteq \overline{S}$, the following implication holds:

$$\mathbb{E}[h(x)] \leq P_0(S) \Rightarrow \mathbb{E}[h(x)] \leq P_1(S).$$

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

$$\mathbb{E}[f(e)] - P_1(S) = \int f_1 d\mu - \int S f_1 d\mu$$

$$= \int S f_1 d\mu - \int S (1-h) f_1 d\mu$$

$$= \int S h \Lambda f_0 d\mu - \int S (1-h) \Lambda f_0 d\mu$$

$$\geq t \cdot \int S f_1 d\mu - t \cdot \int S (1-h) f_1 d\mu$$

$$= t \cdot \int S f_1 d\mu - t \cdot \int S f_0 d\mu$$

$$= t \cdot \mathbb{E}[f(e)] - P_0(S) \geq 0.$$ (65)

The inequality in (63) follows from the fact that whenever $z \in S$, 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 $t_0 \sim P_0$ and $t_1 \sim P_1$ be $Z$-valued random variables with probability density functions $f_0$ and $f_1$ with respect to a measure $\mu$ on $Z$ and let $\phi : X \times Z \to X$ be a semantic transformation. Suppose that $y_A = g(x;x_0)$ and let $p_A, p_B \in [0, 1]$ be bounds to the class probabilities, i.e.,

$$q(y_A|x; x_0) \geq p_A \geq p_B \geq \max_{y \neq y_A} q(y|x; x_0).$$ (66)

For $t \geq 0$, let $S_t, S_{t_1} \subseteq Z$ be the sets defined as $S_t = \{ f_1/f_0 < t \}$ and $S_{t_1} = \{ f_1/f_0 \leq t \}$ and define the function $\xi : [0, 1] \to [0, 1]$ by

$$\xi(p) = \sup_{S_{t_0}} \{ P_1(S) : S_{t_0} \subseteq S \subseteq S_{t_1} \}.$$ (67)

where $t_p = \inf \{ t \geq 0 : P_0(S_{t_p}) > 0 \}$. (68)

If the condition

$$\xi(p_A) + \xi(1 - p_B) > 1$$

is satisfied, then it is guaranteed that $g(x; e_1) = g(x; x_0)$. (69)

PROOF. For ease of notation, let $\xi$ be the function defined by

$$t \mapsto \xi(t) = P_0(S_t)$$

and notice that $t_p = \xi^{-1}(p)$ where $\xi^{-1}$ denotes the generalized inverse of $\xi$. Furthermore, let $S_t, t_B = t_1 - t_B, \overline{S} = \overline{S}(t_A), S_{\overline{S}} = \overline{S}(t_B), S_A = \overline{S}(t_A), S_B = \overline{S}(t_B), S_{\overline{S}} = \overline{S}(t_B)$. We first show that $q(y_A|x; e_1)$ is lower bounded by $\xi(t_A)$. For that purpose, note that by Lemma 6 we have that $\xi(t_A) = P_0(S_{\overline{S}}) \geq p_A \geq P_0(S_{\overline{S}})$. Thus, the collection of sets

$$\mathcal{S}_A = \{ S \subseteq Z : S \subseteq S \subseteq S_{\overline{S}} \}.$$ (70)

is not empty. Pick some $A \in \mathcal{S}_A$ arbitrary and note that, since by assumption $g(x; e_0)$ is $(p_A, p_B)$-confident at $x$ it holds that

$$\mathbb{E}(p(y_A|\phi(x; e_0))) = q(y_A|x; x_0) \geq p_A \geq P_0(A).$$ (71)

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

$$q(y_A|x; e_1) = \mathbb{E}(p(y_A|\phi(x; e_1))) \geq P_1(A).$$ (72)

Since $A \in S_A$ was arbitrary, we take the sup over all $A \in S_A$ and obtain

$$q(y_A|x; e_1) \geq \sup_{A \in S_A} P_1(A) = \xi(p_A)$$ (73)

We now show that for any $y \neq y_A$ the prediction $q(y|x; e_1)$ is upper bounded by $1 - \xi(1 - p_B)$. For that purpose, note that by Lemma 6 we have that $\xi(t_B) = P_0(S_{\overline{S}}) \geq 1 - p_B \geq \mathbb{P}_0(S_{\overline{S}})$. Thus, the collection of sets

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

is not empty. Pick some $B \in \mathcal{S}_B$ arbitrary and note that, since by assumption $g(x; e_0)$ is $(p_A, p_B)$-confident at $x$ it holds that

$$\mathbb{E}(p(y|\phi(x; e_0))) \leq q(y|x; e_0) \leq P_B$$

$$= 1 - (1 - p_B) \leq 1 - \mathbb{P}_0(S_{\overline{S}}).$$ (75)

Since $S_{\overline{S}} \subseteq S_{\overline{S}} \subseteq S_{\overline{S}}$ we can apply part (ii) of Lemma 7 and obtain the upper bound

$$q(y|x; e_1) = \mathbb{E}(p(y|\phi(x; e_1))) \leq 1 - P_1(B).$$ (76)

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

$$q(y|x; e_1) \leq \inf_{B \in \mathcal{S}_B} (1 - P_1(B)) = 1 - (1 - p_B).$$ (77)

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

$$\xi(p_A) \geq 1 - (1 - p_B) > 1$$

it is guaranteed that

$$q(y_A|x; e_1) \geq \max_{y \neq y_A} q(y|x; e_1)$$ (79)

what concludes the proof. □

D PROOFS FOR CERTIFICATION WITH DIFFERENT SMOOTHING DISTRIBUTIONS

Hereby we only list selected main results and leave the proofs to the arXiv version [29] due to space limit.

COROLLARY 7 (Non-Isotropic Gaussian Smoothing). Suppose $Z = \mathbb{R}^m, \Sigma := \sigma^2 \Sigma, e_0, e_1 \sim (0, \Sigma)$ and $e_1 := e_0 + \sigma \epsilon$, $\epsilon \sim \mathcal{N}(0, \Sigma)$ for some $\alpha \in \mathbb{R}^m$. Since the $e_0$-smoothed classifier $g$ is $(p_A, p_B)$-confident at $x \in X$ for some $y_A \in Y$, then, it holds that

$$q(y_A|x; e_1) \geq \max_{y \neq y_A} q(y|x; e_1)$$

where

$$\sqrt{\sum_{i=1}^{m} \sigma_i^2} < \frac{1}{2} \left( \Phi^{-1}(p_A) - \Phi^{-1}(p_B) \right).$$ (80)

COROLLARY 8 (Exponential Smoothing). Suppose $Z = \mathbb{R}^m$, fix some $\lambda > 0$ and let $t_0 \sim \mathcal{E}(1/\lambda), e_0 \sim \epsilon(\epsilon_0, \ldots, \epsilon_m)^T$, $\epsilon_i \sim \mathcal{N}(0, \Sigma)$ for some $\alpha \in \mathbb{R}^m$. Since the $e_0$-smoothed classifier $g$ is $(p_A, p_B)$-confident at $x \in X$ for some $y_A \in Y$, then, it holds that

$$q(y_A|x; e_1) \geq \max_{y \neq y_A} q(y|x; e_1)$$

where

$$\|\epsilon\|_1 < \frac{-\log(1 - p_A + p_B)}{\lambda}.$$ (81)
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 $\{a_i\}_{i=1}^n$; upper bound $M$ of squared $\ell_2$-interpolation error $M^2_k = \arg\max_{\alpha \in [a, b]} \min_{1 \leq i \leq N} \|\phi_R(x, a) - \phi_R(x, a_i)\|^2$.

/* Compute Lipschitz constant $L_0$ */
$a_i \leftarrow a$
for $i = 1, \ldots, N - 1$ do
$
\alpha_{i+1} \leftarrow a + (b - a) \cdot \frac{i}{N - 1} \quad (21)
$
end for
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 $2d_{r,s} \cdot m_A(x, k, \mathcal{P}_{r,s}^{(i)}) \cdot \hat{m}(x, k, \mathcal{P}_{r,s}^{(i)})$ (32)
end for
end for
$L_0 \leftarrow \max_{1 \leq i \leq N - 1} L_0_i \quad (33)$
/* Compute interpolation error bound $M$ (24) from stratified sampling */
for $i = 1, \ldots, N - 1$ do
for $j = 1, \ldots, n - 1$ do
/* Second-level sampling */
$y_{i,j} \leftarrow a + (a_{i+1} - a_i) \cdot \frac{j}{n - 1} \quad (25)$
end for
for $j = 1, \ldots, n - 1$ do
Compute $g_i(y_{i,j}), g_i(y_{i,j+1}), g_{i+1}(y_{i,j}),$ and $g_{i+1}(y_{i,j+1})$ (22)
$M_i \leftarrow \min\{M_i, \min\{g_i(y_{i,j}) + g_i(y_{i,j+1}),
\quad g_{i+1}(y_{i,j}) + g_{i+1}(y_{i,j+1})\}\}$
end for
$M_j \leftarrow \frac{1}{2} M_j + L \cdot \frac{b-a}{(N-1)(n-1)} \quad (27)$
end for
Return: $M \leftarrow \max_{1 \leq i \leq N - 1} M_i \quad (24)$

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

<table>
<thead>
<tr>
<th>Distribution</th>
<th>Value Space</th>
<th>Robust Radius</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gaussian(0, 1)</td>
<td>$(-\infty, \infty)$</td>
<td>$\Phi^{-1}(\rho_A)$</td>
</tr>
<tr>
<td>Laplace(0, $\sqrt{2}$)</td>
<td>$(-\infty, \infty)$</td>
<td>$-\log(2 - 2\rho_A)\sqrt{2}$</td>
</tr>
<tr>
<td>Uniform[$-\sqrt{2}$, $+\sqrt{2}$]</td>
<td>$(-\infty, \infty)$</td>
<td>$2\sqrt{2} \cdot (\rho_A - 1/2)$</td>
</tr>
<tr>
<td>Exponential(1)</td>
<td>$[0, \infty)$</td>
<td>$-\log(2 - 2\rho_A)$</td>
</tr>
<tr>
<td>FoldedGaussian(0, $\sqrt{2/\pi}$)</td>
<td>$[0, \infty)$</td>
<td>$\sqrt{2/\pi} \cdot (\Phi^{-1}(\frac{1}{2\rho_A}) - \Phi^{-1}(\frac{1}{2}))$</td>
</tr>
</tbody>
</table>

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

We leave the proofs to the arXiv version [29] due to space limit.

G  PROOFS FOR DIFFERENTIALLY RESOLVABLE TRANSFORMATIONS

In the corresponding appendix of the arXiv version [29], we provide proofs and technical details for theoretical results about certifying differentially resolvable transformations.

H  TRANSFORMATION DETAILS

Due to space limit, we provide detailed definitions of rotation and scaling transformation in the corresponding appendix of the arXiv version [29].

I  PROOFS FOR INTERPOLATION BOUND COMPUTATION

We state the proofs for the theoretical results governing our approach to certifying rotations and scaling transformations using randomized smoothing in the corresponding appendix of the arXiv version [29]. As a quick glance, the following auxiliary lemma is used for both rotation and scaling:

**Lemma 8.** Let $x \in \mathbb{R}^{K \times W \times H}$, $-\infty < t_1 < t_2 < \infty$ and suppose $\rho: [t_1, t_2] \to [0, W - 1] \times [0, H - 1]$ is a curve of class $C^1$. Let $\psi_k: [t_1, t_2] \to \mathbb{R}, \psi_k(t) := Q_k(k, \rho_1(t), \rho_2(t))$ (82) where $k \in \Omega_k$ and $Q_k$ denotes bilinear interpolation. Then $\psi_k$ is $L_k$-Lipschitz continuous with constant

$L_k = \max_{t \in [t_1, t_2]} \left( \sqrt{2} \|\rho(t)\|_2 \cdot m_A(x, k, [\rho(t)]) \right) (83)$

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 $\sqrt{M^2_S + \sigma^2/\rho^2 \cdot b^2}$ (50) if certifying the composition of rotation or scaling with brightness change within $[-b_0, b_1]$; and is set to $\sqrt{(M_S + r)^2 + \sigma^2/\rho^2 \cdot b^2}$ (51) if certifying the composition of rotation or scaling, brightness change $[-b_0, b_0]$, and $\epsilon_2$ 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. Due to the space limit, most of the results are omitted to the corresponding appendix of the arXiv version [29].
We implement the whole approach along with the training scripts in Algorithm 2.

**Algorithm 2** Progressive Sampling for Certification.

**Input:** clean input image $x$ with true class $k_A$; first-level parameter samples $\{a_i\}_{i=1}^{N_1}$; perturbation random variable $\epsilon$ with variance $\sigma^2$; tolerance $T$; batch size $B$; sampling size limit $n_i$; confidence level $p$.

**Output:** with probability $1 - p$, whether $g(\cdot; \epsilon)$ is certifiably robust at $\phi(x, a_i)$.

for $i = 1, \ldots, N$ do

1. $x^{(i)} \leftarrow \phi(x, a_i)$
2. $j \leftarrow 0$
3. while $j \leq n_i$ do
   1. Sample $B$ instances of $\phi(x^{(i)}, \epsilon)$, and use them to update empirical mean $\hat{q}(y|A|x^{(i)}; \epsilon)$.
   2. $j \leftarrow j + B$.
   3. /* Lower confidence interval bound with these $j$ samples */
   4. $p_A^{(i)} = \text{LowerConfBound}(\hat{q}(y|A|x^{(i)}; \epsilon), j, 1 - p/N)$.
   5. if $R_i = \sigma \Phi^{-1}(p_A^{(i)}) > T$ then
      1. /* Already get the certification that $R_i > T$, break */
      2. Break
   end if
4. end while
5. if $R_i = \sigma \Phi^{-1}(p_A^{(i)}) \leq T$ then
   1. /* Cannot ensure that $R_i > T$. So cannot ensure that $R = \min_i R_i > T$. Early halt */
   2. Return: false
6. end if
7. end for
8. Return: true

K.1 Model Preparation

Due to the space limit, the full description of the model preparation including all hyperparameters are omitted to the corresponding section of the arXiv version [29].

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 [7] 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[37]-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/Al-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 [2, 13] 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. The detailed description of these three attacks can be found in the arXiv version [29].

K.4 Details on Baseline Approaches

DeepG [2] 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 [47] 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 [39] 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 [35] proposes to insert a preprocessing layer to reduce the verification against semantic transformations to the verification against classical $\ell_p$ 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 $\mathbb{P}[r \geq 50^\circ] \leq \mathbb{E}[r]/50 = 92.48\%$, i.e., the certified robust accuracy is no larger than 92.48% when fixing the rotation angle to be $50^\circ$.

DistSPT [13] 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 to TSS as reflected in Table 2. We use their reported numbers in [13, Table 4] for DistSPTV for comparison, since the certification goal and evaluation protocol are the same as ours. On ImageNet, as described in [13, 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 [13] 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...
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 [34, 63, 65] and empirically (e.g., [7, 24, 64]) in classical \( \ell_p \) 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.

### Table 6: Benign accuracy of our TSS models corresponding to those in Table 2. Certified robust accuracy shown as reference.

<table>
<thead>
<tr>
<th>Transformation (Reflection Pad.)</th>
<th>Dataset</th>
<th>Attack Radius</th>
<th>Certified Robust Acc.</th>
<th>Benign Acc.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gaussian Blur</td>
<td>MNIST</td>
<td>Squared Radius ( x \leq 36 )</td>
<td>90.6%</td>
<td>96.6%</td>
</tr>
<tr>
<td></td>
<td>CIFAR-10</td>
<td>Squared Radius ( x \leq 16 )</td>
<td>63.6%</td>
<td>78.2%</td>
</tr>
<tr>
<td></td>
<td>ImageNet</td>
<td>Squared Radius ( x \leq 16 )</td>
<td>51.6%</td>
<td>59.2%</td>
</tr>
<tr>
<td>Translation</td>
<td>MNIST</td>
<td>( \sqrt{x^2 + y^2} \leq 8 )</td>
<td>99.6%</td>
<td>99.6%</td>
</tr>
<tr>
<td></td>
<td>CIFAR-10</td>
<td>( \sqrt{x^2 + y^2} \leq 20 )</td>
<td>80.8%</td>
<td>87.0%</td>
</tr>
<tr>
<td></td>
<td>ImageNet</td>
<td>( \sqrt{x^2 + y^2} \leq 100 )</td>
<td>50.0%</td>
<td>73.0%</td>
</tr>
<tr>
<td>Brightness</td>
<td>MNIST</td>
<td>( b \leq 50)</td>
<td>98.2%</td>
<td>98.2%</td>
</tr>
<tr>
<td></td>
<td>CIFAR-10</td>
<td>( b \leq 40)</td>
<td>87.0%</td>
<td>87.8%</td>
</tr>
<tr>
<td></td>
<td>ImageNet</td>
<td>( b \leq 40)</td>
<td>70.0%</td>
<td>72.2%</td>
</tr>
<tr>
<td>Contrast and Brightness</td>
<td>MNIST</td>
<td>( c \leq 50), ( b \leq 50)</td>
<td>97.6%</td>
<td>98.0%</td>
</tr>
<tr>
<td></td>
<td>CIFAR-10</td>
<td>( c \leq 40), ( b \leq 40)</td>
<td>82.4%</td>
<td>86.8%</td>
</tr>
<tr>
<td></td>
<td>ImageNet</td>
<td>( c \leq 40), ( b \leq 40)</td>
<td>61.4%</td>
<td>72.2%</td>
</tr>
<tr>
<td>Gaussian Blur, Translation, Brightness, and Contrast</td>
<td>MNIST</td>
<td>( a \leq 1, b \leq 10, \sqrt{x^2 + y^2} \leq 5 )</td>
<td>90.2%</td>
<td>98.2%</td>
</tr>
<tr>
<td></td>
<td>CIFAR-10</td>
<td>( a \leq 1, b \leq 10, \sqrt{x^2 + y^2} \leq 5 )</td>
<td>58.2%</td>
<td>77.6%</td>
</tr>
<tr>
<td></td>
<td>ImageNet</td>
<td>( a \leq 0.1, b \leq 0.1, \sqrt{x^2 + y^2} \leq 0.1 )</td>
<td>32.8%</td>
<td>61.4%</td>
</tr>
<tr>
<td>Rotation</td>
<td>MNIST</td>
<td>( r \leq 5^\circ )</td>
<td>97.4%</td>
<td>99.4%</td>
</tr>
<tr>
<td></td>
<td>CIFAR-10</td>
<td>( r \leq 10^\circ )</td>
<td>79.6%</td>
<td>83.2%</td>
</tr>
<tr>
<td></td>
<td>ImageNet</td>
<td>( r \leq 30^\circ )</td>
<td>63.6%</td>
<td>82.6%</td>
</tr>
<tr>
<td>Scaling</td>
<td>MNIST</td>
<td>( s \leq 30)</td>
<td>99.0%</td>
<td>99.4%</td>
</tr>
<tr>
<td></td>
<td>CIFAR-10</td>
<td>( s \leq 30)</td>
<td>58.8%</td>
<td>79.8%</td>
</tr>
<tr>
<td></td>
<td>ImageNet</td>
<td>( s \leq 30)</td>
<td>26.4%</td>
<td>50.8%</td>
</tr>
<tr>
<td>Rotation and Brightness</td>
<td>MNIST</td>
<td>( r \leq 5^\circ ), ( b \leq 20)</td>
<td>97.0%</td>
<td>99.4%</td>
</tr>
<tr>
<td></td>
<td>CIFAR-10</td>
<td>( r \leq 10^\circ ), ( b \leq 10)</td>
<td>70.2%</td>
<td>83.0%</td>
</tr>
<tr>
<td></td>
<td>ImageNet</td>
<td>( r \leq 30^\circ ), ( b \leq 20)</td>
<td>61.4%</td>
<td>82.6%</td>
</tr>
<tr>
<td>Scaling</td>
<td>MNIST</td>
<td>( s \leq 50), ( b \leq 50)</td>
<td>96.6%</td>
<td>99.4%</td>
</tr>
<tr>
<td></td>
<td>CIFAR-10</td>
<td>( s \leq 50), ( b \leq 50)</td>
<td>54.2%</td>
<td>79.6%</td>
</tr>
<tr>
<td></td>
<td>ImageNet</td>
<td>( s \leq 50), ( b \leq 50)</td>
<td>23.4%</td>
<td>50.8%</td>
</tr>
<tr>
<td>Rotation, Brightness, and ( \ell_2 )</td>
<td>MNIST</td>
<td>( r \leq 5^\circ ), ( b \leq 20), ( |\delta|_2 \leq 0.05 )</td>
<td>96.6%</td>
<td>99.4%</td>
</tr>
<tr>
<td></td>
<td>CIFAR-10</td>
<td>( r \leq 10^\circ ), ( b \leq 10), ( |\delta|_2 \leq 0.05 )</td>
<td>64.2%</td>
<td>83.0%</td>
</tr>
<tr>
<td></td>
<td>ImageNet</td>
<td>( r \leq 30^\circ ), ( b \leq 20), ( |\delta|_2 \leq 0.05 )</td>
<td>55.2%</td>
<td>82.6%</td>
</tr>
<tr>
<td>Scaling, Brightness, and ( \ell_2 )</td>
<td>MNIST</td>
<td>( s \leq 50), ( b \leq 50), ( |\delta|_2 \leq 0.05 )</td>
<td>96.4%</td>
<td>99.4%</td>
</tr>
<tr>
<td></td>
<td>CIFAR-10</td>
<td>( s \leq 30), ( b \leq 30), ( |\delta|_2 \leq 0.05 )</td>
<td>51.2%</td>
<td>79.6%</td>
</tr>
<tr>
<td></td>
<td>ImageNet</td>
<td>( s \leq 30), ( b \leq 30), ( |\delta|_2 \leq 0.05 )</td>
<td>22.6%</td>
<td>50.8%</td>
</tr>
</tbody>
</table>

### 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, \( \alpha \) is for squared kernel radius for Gaussian blur; \( \Delta x \) and \( \Delta y \) 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; \( r \) is for additive noise vector; and \( \|\delta\|_2 \) for \( \ell_2 \) 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

Detailed results are omitted to the corresponding section of the arXiv version [29].

### K.8 Empirical Robustness against Unforeseen Attacks: Evaluation Protocol and Result Breakdown

Detailed results are omitted to the corresponding section of the arXiv version [29].

### K.10 Different Smoothing Variance Levels: More Results

Detailed results are omitted to the corresponding section of the arXiv version [29].

### K.11 Tightness-Efficiency Trade-Off

Detailed results are omitted to the corresponding section of the arXiv version [29].
<table>
<thead>
<tr>
<th>Transformation</th>
<th>Dataset</th>
<th>Attack</th>
<th>Training</th>
<th>Smoothing</th>
<th>Avg. Cert. Time per Sample</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>Distribution</td>
<td>Distribution</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Gaussian Blur</td>
<td>MNIST</td>
<td>Squared Radius $a \leq 36$</td>
<td>$a \sim \text{Exp}(1/10)$</td>
<td>$a \sim \text{Exp}(1/10)$</td>
<td>7.9 s</td>
</tr>
<tr>
<td></td>
<td>CIFAR-10</td>
<td>Squared Radius $a \leq 16$</td>
<td>$a \sim \text{Exp}(1/5)$</td>
<td>$a \sim \text{Exp}(1/5)$</td>
<td>30.9 s</td>
</tr>
<tr>
<td></td>
<td>ImageNet</td>
<td>Squared Radius $a \leq 36$</td>
<td>$a \sim \text{Exp}(1/10)$</td>
<td>$a \sim \text{Exp}(1/10)$</td>
<td>45.7 s</td>
</tr>
<tr>
<td>Translation (Reflection Pad.)</td>
<td>MNIST</td>
<td>$\sqrt{\Delta x^2 + \Delta y^2} \leq 8$</td>
<td>$(\Delta x, \Delta y) \sim \mathcal{N}(0, 10^2 I)$</td>
<td>$(\Delta x, \Delta y) \sim \mathcal{N}(0, 10^2 I)$</td>
<td>10.2 s</td>
</tr>
<tr>
<td></td>
<td>CIFAR-10</td>
<td>$\sqrt{\Delta x^2 + \Delta y^2} \leq 20$</td>
<td>$(\Delta x, \Delta y) \sim \mathcal{N}(0, 15^2 I)$</td>
<td>$(\Delta x, \Delta y) \sim \mathcal{N}(0, 15^2 I)$</td>
<td>39.4 s</td>
</tr>
<tr>
<td></td>
<td>ImageNet</td>
<td>$\sqrt{\Delta x^2 + \Delta y^2} \leq 100$</td>
<td>$(\Delta x, \Delta y) \sim \mathcal{N}(0, 30^2 I)$</td>
<td>$(\Delta x, \Delta y) \sim \mathcal{N}(0, 30^2 I)$</td>
<td>161.9 s</td>
</tr>
<tr>
<td>Brightness</td>
<td>MNIST</td>
<td>$b \leq 50%$</td>
<td>$b \sim \mathcal{N}(0, 0.6^2 I)$</td>
<td>$b \sim \mathcal{N}(0, 0.6^2 I)$</td>
<td>2.1 s</td>
</tr>
<tr>
<td></td>
<td>CIFAR-10</td>
<td>$b \leq 40%$</td>
<td>$b \sim \mathcal{N}(0, 0.5^2 I)$</td>
<td>$b \sim \mathcal{N}(0, 0.5^2 I)$</td>
<td>4.4 s</td>
</tr>
<tr>
<td></td>
<td>ImageNet</td>
<td>$b \leq 40%$</td>
<td>$b \sim \mathcal{N}(0, 0.4^2 I)$</td>
<td>$b \sim \mathcal{N}(0, 0.4^2 I)$</td>
<td>45.1 s</td>
</tr>
<tr>
<td>Contrast and Brightness</td>
<td>MNIST</td>
<td>$c \leq 50%; b \leq 50%$</td>
<td>$(c,b) \sim \mathcal{N}(0, 0.6^2 I)$</td>
<td>$(c,b) \sim \mathcal{N}(0, 0.6^2 I)$</td>
<td>9.8 s</td>
</tr>
<tr>
<td></td>
<td>CIFAR-10</td>
<td>$c \leq 40%; b \leq 40%$</td>
<td>$(c,b) \sim \mathcal{N}(0, 0.4^2 I)$</td>
<td>$(c,b) \sim \mathcal{N}(0, 0.4^2 I)$</td>
<td>45.0 s</td>
</tr>
<tr>
<td></td>
<td>ImageNet</td>
<td>$c \leq 40%; b \leq 40%$</td>
<td>$(c,b) \sim \mathcal{N}(0, 0.4^2 I)$</td>
<td>$c \sim \mathcal{N}(0.05^2 I)$</td>
<td>325.6 s</td>
</tr>
<tr>
<td>Gaussian Blur, Translation, Brightness, and Contrast</td>
<td>MNIST</td>
<td>$c \leq 5, c, b \leq 10%$, $\sqrt{\Delta x^2 + \Delta y^2} \leq 5$</td>
<td>$c \sim \text{Exp}(1/10)$</td>
<td>$c \sim \text{Exp}(1/10)$</td>
<td>12.9 s</td>
</tr>
<tr>
<td></td>
<td>CIFAR-10</td>
<td>$c \leq 1, c, b \leq 10%$, $\sqrt{\Delta x^2 + \Delta y^2} \leq 5$</td>
<td>$c \sim \text{Exp}(1/5)$</td>
<td>$c \sim \text{Exp}(1/5)$</td>
<td>43.1 s</td>
</tr>
<tr>
<td></td>
<td>ImageNet</td>
<td>$c \leq 10, c, b \leq 20%$, $\sqrt{\Delta x^2 + \Delta y^2} \leq 10$</td>
<td>$c \sim \text{Exp}(1/5)$</td>
<td>$c \sim \text{Exp}(1/5)$</td>
<td>238.1 s</td>
</tr>
<tr>
<td>Rotation</td>
<td>MNIST</td>
<td>$r \leq 50^\circ$</td>
<td>Same as Rotation and Brightness</td>
<td>$r \sim \mathcal{N}(0, 0.12^2 I)$</td>
<td>20.1 s</td>
</tr>
<tr>
<td></td>
<td>CIFAR-10</td>
<td>$r \leq 10^\circ$</td>
<td>Same as Rotation and Brightness</td>
<td>$r \sim \mathcal{N}(0, 0.05^2 I)$</td>
<td>52.8 s</td>
</tr>
<tr>
<td></td>
<td>ImageNet</td>
<td>$r \leq 10^\circ$</td>
<td>Same as Rotation and Brightness</td>
<td>$r \sim \mathcal{N}(0, 0.05^2 I)$</td>
<td>141.0 s</td>
</tr>
<tr>
<td>Scaling</td>
<td>MNIST</td>
<td>$s \leq 30%$</td>
<td>Same as Scaling and Brightness</td>
<td>$s \sim \mathcal{N}(0, 0.12^2 I)$</td>
<td>17.7 s</td>
</tr>
<tr>
<td></td>
<td>CIFAR-10</td>
<td>$s \leq 30%$</td>
<td>Same as Scaling and Brightness</td>
<td>$s \sim \mathcal{N}(0, 0.05^2 I)$</td>
<td>42.2 s</td>
</tr>
<tr>
<td></td>
<td>ImageNet</td>
<td>$s \leq 30%$</td>
<td>Same as Scaling and Brightness</td>
<td>$s \sim \mathcal{N}(0, 0.05^2 I)$</td>
<td>120.2 s</td>
</tr>
<tr>
<td>Rotation and Brightness</td>
<td>MNIST</td>
<td>$r \leq 50^\circ, b \leq 20%$</td>
<td>$r \sim \mathcal{U}([-55, 55])$</td>
<td>$r \sim \mathcal{N}(0, 0.12^2 I)$</td>
<td>31.4 s</td>
</tr>
<tr>
<td></td>
<td>CIFAR-10</td>
<td>$r \leq 10^\circ, b \leq 10%$</td>
<td>$r \sim \mathcal{U}([-12.5, 12.5])$</td>
<td>$r \sim \mathcal{N}(0, 0.05^2 I)$</td>
<td>62.3 s</td>
</tr>
<tr>
<td></td>
<td>ImageNet</td>
<td>$r \leq 30^\circ, b \leq 20%$</td>
<td>$r \sim \mathcal{U}([-35, 35])$</td>
<td>$r \sim \mathcal{N}(0, 0.05^2 I)$</td>
<td>157.0 s</td>
</tr>
<tr>
<td>Scaling and Brightness</td>
<td>MNIST</td>
<td>$s \leq 50%; b \leq 50%$</td>
<td>$s \sim \mathcal{U}([0.45, 1.55])$</td>
<td>$s \sim \mathcal{N}(0, 0.12^2 I)$</td>
<td>74.9 s</td>
</tr>
<tr>
<td></td>
<td>CIFAR-10</td>
<td>$s \leq 30%; b \leq 30%$</td>
<td>$s \sim \mathcal{U}([0.65, 1.35])$</td>
<td>$s \sim \mathcal{N}(0, 0.05^2 I)$</td>
<td>44.5 s</td>
</tr>
<tr>
<td></td>
<td>ImageNet</td>
<td>$s \leq 30%; b \leq 30%$</td>
<td>$s \sim \mathcal{U}([0.65, 1.35])$</td>
<td>$s \sim \mathcal{N}(0, 0.05^2 I)$</td>
<td>140.1 s</td>
</tr>
<tr>
<td>Rotation, Brightness, and $f_2$</td>
<td>MNIST</td>
<td>$r \leq 50^\circ, b \leq 20%, [0:b] \leq 0.05$</td>
<td>Same as Rotation and Brightness</td>
<td>$r \sim \mathcal{N}(0.12^2 I)$</td>
<td>35.1 s</td>
</tr>
<tr>
<td></td>
<td>CIFAR-10</td>
<td>$r \leq 10^\circ, b \leq 10%, [0:b] \leq 0.05$</td>
<td>Same as Rotation and Brightness</td>
<td>$r \sim \mathcal{N}(0.05^2 I)$</td>
<td>132.5 s</td>
</tr>
<tr>
<td></td>
<td>ImageNet</td>
<td>$r \leq 30^\circ, b \leq 20%, [0:b] \leq 0.05$</td>
<td>Same as Rotation and Brightness</td>
<td>$r \sim \mathcal{N}(0.05^2 I)$</td>
<td>520.2 s</td>
</tr>
<tr>
<td></td>
<td>ImageNet</td>
<td>$r \leq 50^\circ, b \leq 20%, [0:b] \leq 0.05$</td>
<td>Same as Rotation and Brightness</td>
<td>$r \sim \mathcal{N}(0.05^2 I)$</td>
<td>3463.8 s</td>
</tr>
<tr>
<td>Scaling, Brightness, and $f_2$</td>
<td>MNIST</td>
<td>$s \leq 50%; b \leq 50%$, $[0:b] \leq 0.05$</td>
<td>Same as Scaling and Brightness</td>
<td>$s \sim \mathcal{N}(0.12^2 I)$</td>
<td>75.1 s</td>
</tr>
<tr>
<td></td>
<td>CIFAR-10</td>
<td>$s \leq 30%; b \leq 30%$, $[0:b] \leq 0.05$</td>
<td>Same as Scaling and Brightness</td>
<td>$s \sim \mathcal{N}(0.05^2 I)$</td>
<td>50.0 s</td>
</tr>
<tr>
<td></td>
<td>ImageNet</td>
<td>$s \leq 30%; b \leq 30%$, $[0:b] \leq 0.05$</td>
<td>Same as Scaling and Brightness</td>
<td>$s \sim \mathcal{N}(0.05^2 I)$</td>
<td>1657.7 s</td>
</tr>
</tbody>
</table>