Adaptive Randomized Smoothing for Certified Multi-Step Defence

Shadab Shaikh[1], Saiyue Lyu[1], Frederick Shpilevskiy[1], Evan Shelhamer[2], Mathias Lécuyer[1]
[1]University of British Columbia, [2]Imaginary Number
{shadabs3, saiyuel, fshpil}@cs.ubc.ca, shelhamer@imaginarynumber.net, mathias.lecuyer@ubc.ca

Abstract

We propose Adaptive Randomized Smoothing (ARS) to certify the predictions of test-time adaptive models against adversarial examples. ARS extends the analysis of randomized smoothing using f-differential privacy to certify the adaptive composition of several steps. We instantiate ARS on deep image classification to provide certified predictions against adversarial examples of bounded L_∞ norm. We create an adaptivity benchmark from CIFAR-10 and show that adapting improves certified accuracy by up to 9.6%.

1. Introduction

Despite impressive accuracy, deep learning models still show a worrying susceptibility to adversarial attacks. Such attacks have been shown for a large number of tasks and models [4, 6], including in security and safety critical areas such as fraud detection [16] or self-driving [3].

Several rigorous defenses have been proposed to provide robustness guarantees. Randomized Smoothing (RS) [5, 13] is one such approach that averages predictions over a noisy version of the input at test time. However, RS has its limitations: it is inflexible and either degrades accuracy or only certifies against small attacks.

We (re)connect RS to Differential Privacy (DP), after its abandonment for a tighter analysis via hypothesis testing [5], to address these shortcomings while preserving tight bounds. In doing so we introduce **Adaptive Randomized Smoothing (ARS)**, a two step method for defending against L_∞ adversaries on image classification, which is a challenging setting for RS [2]. The first step computes an input mask that focuses on task-relevant information. This reduces the dimension of the input, which is then passed to the second step for prediction. Thanks to this adaptive dimension reduction, the second step makes its prediction on a less noisy image, improving the performance and certification radius.

We evaluate our adaptive method in two settings. On a challenging adaptivity benchmark that we derive from CIFAR10, we show that ARS can improve accuracy by up to 9.6% on high-dimensional inputs.

2. Theory

2.1. Background

We introduce the necessary background on RS and DP.

Adversarial Examples: Consider a classification model $g : \mathcal{X} \to \mathcal{Y}$, and input X. An adversarial example of radius r in the L_p threat model, for model g on input X, is an input $X + e$ such that $g(X + e) \neq g(X)$, where $e \in B_p(r)$. This is a test time attack vector on ML models.

Randomized Smoothing (RS) [5, 13] certifies the predictions of a model against L_p-norm adversaries. The algorithm randomizes a base model g by adding spherical Gaussian noise to its input and predicting the class with highest expectation: $y_+ \triangleq \arg\max_{y\in\mathcal{Y}} \mathbb{P}_{z\sim\mathcal{N}(0,\sigma^2\mathbb{I}^d)}\big(g(X+z) = y\big)$. The tightest analysis [5] uses theory from hypothesis testing to show that for a prediction y on input X and with $\underline{p_+}, \overline{p_-} \in [0, 1]$ such that $\mathbb{P}(g(X + z) = y_+) \geq \underline{p_+} \geq \overline{p_-} \geq \max_{y_-\neq y_+} \mathbb{P}(g(X + z) = y_-)$, the certificate size r_X for prediction y_+ is:

$$r_X = \frac{\sigma}{2}\big(\Phi^{-1}(\underline{p_+}) - \Phi^{-1}(\overline{p_-})\big), \tag{1}$$

where Φ^{-1} is the inverse of the standard Gaussian CDF. Note that, $\underline{p_+}$ is a lower-bound on the probability that $g(X+z) = y_+$ (highest probability class), and $\overline{p_-}$ an upper-bound on the probability of any other class.

The RS algorithm was initially analyzed using Differential Privacy [13]. Intuitively, one can see the randomized classifier $g(X+z),\ z \sim \mathcal{N}(0, \sigma^2\mathbb{I}^d)$ as a privacy preserving mechanism (the Gaussian mechanism) that provably "hides" small variations in the input X. This privacy guarantee yields a robustness certificate for the model's expected predictions. This original analysis uses the notion of (ϵ, δ)-DP.

Differential Privacy (DP) is a rigorous notion of privacy. A randomized mechanism $\mathcal{M}$ is (ϵ, δ)-DP if, for any neighbouring inputs X and X', and any subset of possible outputs $\mathcal{Y} \subset \text{Range}(\mathcal{M})$, $\mathbb{P}(\mathcal{M}(X) \in \mathcal{Y}) \leq e^\epsilon \mathbb{P}(\mathcal{M}(X') \in \mathcal{Y}) + \delta$. Following Lécuyer et al. [13], we use a different definition based on L_p norms, and say that X and X' in $\mathbb{R}^d$ are neighbours at radius r for L_p norm if $X - X' \in B_p(r)$, where $B_p(r) \triangleq \{x \in \mathbb{R}^d : \|x\|_p \leq r\}$ is the L_p ball of radius r.

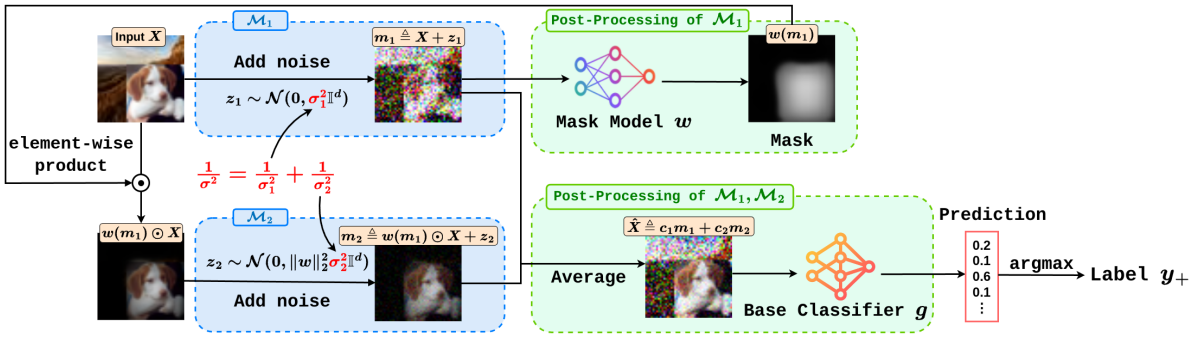


Figure 1. Two-step Adaptive Randomized Smoothing (ARS). First step $\mathcal{M}_1$ it adds noise to input X and post-processes the result into a mask $w(m_1)$. Second step $\mathcal{M}_2$ takes the element-wise masked input $w(m_1) \odot X$ and adds noise to get m_2. ARS post-processes the weighted average of m_1, m_2 using the base classifier to output a label. More details in Sec. 3. Vanilla RS sets $\sigma_2 = \sigma$ and a mask $w(.) = 1$ (no $\mathcal{M}_1$).

In this paper we show that we can use f-DP, a more recent formulation of DP, to analyze RS with results as tight as those of Cohen et al. [5]. We next introduce f-DP, and will formally connect it to RS and use it to analyze test-time adaptive models in §2.2.

***f*-DP** from Dong et al. [8, 9] is an extension of DP that defines privacy as a bound on hypothesis tests. We leverage Theorem 2.7 from [9] that a Gaussian mechanism of the form $\mathcal{M}(X) = \theta(X) + z, z \sim \mathcal{N}(0, \frac{r^2}{\mu^2})$, such that $\forall X, X' : X - X' \in B_2(r) \Rightarrow \theta(X) - \theta(X') \in B_2(r)$ (i.e., the L_2 sensitivity of θ is r), then $\mathcal{M}$ is G_μ-DP, with:

$$G_\mu(\alpha) = \Phi\Big(\Phi^{-1}(1-\alpha) - \mu\Big), \tag{2}$$

where Φ is the standard normal CDF.

We leverage two key properties of f-DP. First, f-DP is resilient to post-processing. That is, if mechanism $\mathcal{M}$ is f-DP, proc $\circ \mathcal{M}$ is also f-DP. Second, f-DP is closed under adaptive composition. We refer the interested reader to §3 in Dong et al. [9] for the precise definition. We use Corollary 3.3 in [9]: the adaptive composition of two Gaussian mechanisms G_{μ_1}-DP and G_{μ_2}-DP is itself G_μ-DP, with:

$$\mu = \sqrt{\mu_1^2 + \mu_2^2} \tag{3}$$

2.2. Adaptive Randomized Smoothing (ARS)

Our analysis of RS using f-DP is in Appendix 6.1. At a high level, we show general robustness results for classifiers that predict the expected classification of f-DP mechanisms (Proposition 6.1), which lets us analyze RS classifiers (Proposition 6.2) with a result as strong as that of Equation (1).

We next detail our key insight: this connection between RS and f-DP lets us extend RS to adaptive muti-step architectures, an approach we name ARS. Adaptive composition of mechanisms is at the core of DP algorithms, and we can leverage known results for f-DP to create certifiable adaptive prediction models that adapt their behaviour based on the input. This is an approach that has seen recent empirical interest [1, 7], but to the best of our knowledge lacks a flexible end-to-end analysis [17]. We then show one instantiation of ARS, for L_∞ adversaries.

ARS Formulation. Consider k randomized Gaussian mechanisms $\mathcal{M}_1, \dots, \mathcal{M}_k$, such that mechanism i outputs $m_i \sim \mathcal{M}_i(X|m_{<i})$, and for any $r \geq 0$ is G_{r/σ_i}-DP for the $B_p(r)$ neighboring definition. Note that the computation $\mathcal{M}_i$ can depend on previous results, as long as it is G_{r/σ_i}-DP. Further consider a (potentially randomized) post-processing function mapping the outputs of $\mathcal{M}_1, \dots, \mathcal{M}_k$ to a classification: $g(m_1, \dots, m_k) = y \in \mathcal{Y}$.

Proposition 2.1 (Adaptive RS). *Let* $\mathcal{M} : X \to g(m_1, \dots, m_k) \in \mathcal{Y}$, $(m_1, \dots, m_k) \sim (\mathcal{M}_1(X), \dots, \mathcal{M}_k(X|m_{<k}))$, *and the associated smooth model* $M_S : X \to \arg\max_{y \in \mathcal{Y}} \mathbb{P}(\mathcal{M}(X) = y)$. *Let* $y_+ \triangleq M_S(X)$ *be the prediction on input* X, *and let* $\underline{p_+}, \overline{p_-} \in [0,1]$ *be such that* $\mathbb{P}(\mathcal{M}(X) = y_+) \geq \underline{p_+} \geq \overline{p_-} \geq \max_{y_- \neq y_+} \mathbb{P}(\mathcal{M}(X) = y_-)$. *Then* $\forall e \in B_2(r_x)$, $M_S(X + e) = y_+$, *with:*

$$r_X = \frac{1}{2\sqrt{\sum_{i=1}^k \frac{1}{\sigma_i^2}}} \Big(\Phi^{-1}(\underline{p_+}) - \Phi^{-1}(\overline{p_-})\Big).$$

Proof. In Appendix 6.2. □

***L_∞* Adversary.** We use ARS to design a two-steps certified defence against an L_∞-bounded adversary. Previous work already noticed that RS applies to L_∞-bounded adversary [5, 13, 18], using the fact that any $X, X' \in \mathbb{R}^d$ such that $X - X' \in B_\infty(r^\infty)$, then $X - X' \in B_2(\sqrt{d} \cdot r^\infty)$, using the fact that $\forall X \in \mathbb{R}^d, \|X\|_2 \leq \sqrt{d}\|X\|_\infty$. This, coupled with Equation (1), yields:

$$r_X^\infty = \frac{\sigma}{2\sqrt{d}} \big(\Phi^{-1}(\underline{p_+}) - \Phi^{-1}(\overline{p_-})\big) \tag{4}$$

While there is L_∞-specific theory for RS [18], further work by Blum et al. [2] finds that Gaussian RS performs

advantageously in practice, but that the $\sqrt{d}$ dependency cannot be avoided. They therefore speculate that RS might be inherently limited for L_∞ certification on high dimensional images. Our ARS architecture side-steps this issue by leveraging adaptivity over two steps, to first to select subsets of the image important to the classification task (thereby reducing dimension), and the second to make the prediction based on the selected subset. Formally:

Proposition 2.2 (Adaptive RS for L_∞)**.** *Define the following pair of (adaptive) mechanisms:*

$$\mathcal{M}_1 : X \to X + z_1 \triangleq m_1, \quad z_1 \sim \mathcal{N}(0, \sigma_1^2 \mathbb{I}^d) \tag{5}$$

Then defining $w : \mathbb{R}^d \to [0,1]^d$*:*

$$\begin{aligned} &\mathcal{M}_2 : X, m_1 \to w(m_1) \odot X + z_2 \triangleq m_2, \\ &z_2 \sim \mathcal{N}(0, \frac{\|w(m_1)\|_2^2}{d} \sigma_2^2 \mathbb{I}^d), \end{aligned} \tag{6}$$

where $\odot$ *is the element-wise product; and the final prediction function* $g : m_1, m_2 \to \mathcal{Y}$*. Consider the mechanism* $\mathcal{M}$ *that samples* $m_1 \sim \mathcal{M}_1$*, then* $m_2 \sim \mathcal{M}_2$*, and finally outputs* $g(m_1, m_2)$*; and the associated smooth classifier* $M_S : X \to \arg\max_{y \in \mathcal{Y}} \mathbb{P}(\mathcal{M}(X) = y)$*. Let* $y_+ \triangleq M_S(X)$ *be the prediction on input* X*, and let* $\underline{p_+}, \overline{p_-} \in [0,1]$ *be such that* $\mathbb{P}(\mathcal{M}(X) = y_+) \geq \underline{p_+} \geq \overline{p_-} \geq \max_{y_- \neq y_+} \mathbb{P}(\mathcal{M}(X) = y_-)$*. Then* $\forall e \in B_\infty(r_X^\infty),\ M_S(X+e) = y_+$*, with:*

$$r_X^\infty = \frac{1}{2\sqrt{d\left(\frac{1}{\sigma_1^2} + \frac{1}{\sigma_2^2}\right)}} \left(\Phi^{-1}(\underline{p_+}) - \Phi^{-1}(\overline{p_-}) \right).$$

Proof. In Appendix 6.3. □

Remarks. 1. $w(\cdot)$ acts as a masking function, adaptively reducing (if $w_i(m_1) << 1$) the value of X_i. Intuitively, this reduces the effective dimension of the input, and hence the attack surface of an adversary, in the second mechanism. 2. This dimension reduction by masking enables reducing the variance of the second mechanism's at fixed privacy guarantee (fixed G_{μ_2}). This variance reduction is enabled *for all dimensions in the input*, even those that are not masked ($w_i(m_1) \approx 1$). As a result, the variance of the noise in $\mathcal{M}_2$ scales as $\|w(m_1)\|_2^2 \leq d$. The more masking, the lower the variance. This gain is specific to an L_∞ adversary, and does not apply to L_2. This achieves higher accuracy, as well as further apart $\underline{p_+}$ and $\overline{p_-}$, for a larger r_X^∞.

3. Architecture

Figure 1 summarizes our two-step ARS instantiation for L_∞ attacks. A standard RS approach adds Gaussian noise of standard deviation σ (a hyper-parameter) to the input, before feeding this noisy input to the base classifier g. The final predictions are averaged over the noise to create the smooth classifier. Our ARS architecture introduces several new components, and is trained end-to-end on the same classification task and with the same procedure as RS.

Budget Splitting: the noise budget, hyper-parameter σ, is split to assign individual noise levels to the two steps: mechanisms $\mathcal{M}_1$ and $\mathcal{M}_2$. Note that σ is the total amount of noise in the our two-steps model, and can be interpreted as the standard deviation of noise in RS. We hence parameterize ARS with the same σ as standard RS, and use the composition formula (Equation (3)) to split it. In practice, we assign $\sigma_1 \geq \sigma$ to $\mathcal{M}_1$, and then $\sigma_2 = 1/\sqrt{\frac{1}{\sigma^2} - \frac{1}{\sigma_1^2}}$. We experiment with two ways to set σ_1: (a) fixing it to a constant (b) learning it: add it as a trainable parameter to our model.

Masking: the mask model w takes the noisy image from $\mathcal{M}_1$ and makes a mask (one value in $[0,1]$ per input pixel) that is multiplied with the input element-wise (denoted $\odot$ in Proposition 2.2). $w(\cdot)$ is a segmentation network with residual blocks, and acts as a post-processing of $\mathcal{M}_1$ in the f-DP analysis. The masking for $\mathcal{M}_2$ enables test-time adaptivity via its dependence on the noisy input image y_1.

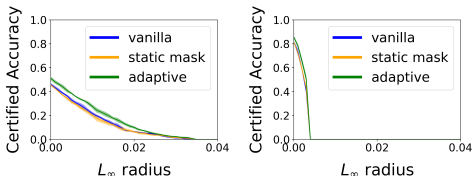
Mechanism output averaging: to fully leverage both steps' information, we take a weighted average of the outputs m_1 and m_2 before passing the result to the base classifier g. For a particular input pixel i, denote X_i the value of pixel, $w_i \in [0,1]$ its mask weight (we omit the explicit dependency on m_1 in w for compactness), and $m_{1,i}, m_{2,i}$ the respective values output by $\mathcal{M}_1$ and $\mathcal{M}_2$. Then, the final value of pixel i in the averaged input will be $\hat{X}_i \triangleq c_{1,i} m_{1,i} + c_{2,i} m_{2,i}$, where

$$c_{1,i} = \frac{\|w\|_2^2 \sigma_2^2}{\sigma_1^2 w_i^2 + \|w\|_2^2 \sigma_2^2},\ c_{2,i} = \frac{\sigma_1^2 w_i}{\sigma_1^2 w_i^2 + \|w\|_2^2 \sigma_2^2}.$$

Details about the derivation can be found in Appendix 7.1. The averaged noisy input $\hat{X}$ is finally fed to the base classifier g for prediction. The smooth classifier averages predictions over the entire pipeline. The parameters of w, g, and the budget split (if not fixed) are learned during training and are fixed at inference/certification time.

4. Experiments

We evaluate our ARS classifier by its certified and standard test accuracy. The certified test accuracy, at a specified radius r, is the percentage of test samples correctly classified *and* certifies an L_∞ radius $r_X^\infty \geq r$. Standard test accuracy is equal to to certified test accuracy at $r = 0$. To measure certified test accuracy, we need to compute r_X^∞ for each test point X. We follow the Monte Carlo procedure for L_2 from Cohen et al. [5] and, for each input X, we first use n samples $\mathcal{M}(X)$ to measure y_+, and then use $N >> n$ samples to estimate $\underline{p_+}$ and use $\overline{p_-} = 1 - \underline{p_+}$. We then divide the radius by $\sqrt{d}$ to convert it to L_∞, following Proposition 2.2. We make predictions by the majority vote of 100 noisy samples



(a) $\sigma = 0.75$ (b) $\sigma = 0.075$

Figure 2. Mean certified test accuracy for different noise levels on our 20kBG CIFAR-10 ($k = 48$, edges). One standard deviation is shaded.

Setting/Approach	Vanilla	Static Mask	ARS
20kBG, 2 loc., $\sigma = .075$	81.9 (.007)	80.3 (.01)	**83.6 (.01)**
20kBG, edges, $\sigma = .075$	81.8 (.008)	81.8 (0.008)	**85.7 (.01)**
20kBG, 2 loc., $\sigma = .75$	47.5 (.02)	48.8 (.04)	**57.1 (.01)**
20kBG, edges, $\sigma = .75$	45.9 (.01)	45.6 (0.01)	**51.2 (.01)**

Table 1. Standard test accuracy ($r = 0$) for **20kBG**. Reported numbers are percentage points in the form: **mean (standard deviation).**

for each test input. We report the mean certified test accuracy and standard deviation over 5 random seeds.

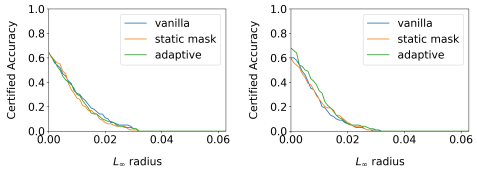
Datasets: We design challenging benchmarks for L_∞ certification based on the CIFAR-10 [12] and CelebA [15].

Models: We choose ResNet-110 adapted for the CIFAR-10 dataset [10] as the base classifier g and a modified ResNet-110 for our mask model w. We compare two baselines: (1) the RS approach of Cohen et al. [5], which we refer to as "vanilla", and (2) RS with a static mask. The static mask is learned during training and does not adapt to test inputs. We directly train a pixelwise parameterization of the mask, then test by masking the input to the noise layer and classifier.

CIFAR-10: We design a benchmark to vary input dimension, as it is a key challenge in L_∞ certification using RS (see §2.2). We take CIFAR-10 images and superimpose them onto a larger background (see appendix 7.2 for an illustration). We sample these backgrounds from BG-20k, a dataset of $20k$ background images [14], and re-scale to the desired dimensionality. We sample backgrounds from the BG-20k train set for training and the BG-20k test set for testing. We refer to this evaluation as 20kBG. For 5BG (only 5 background images) see appendix 7.4 for more details.

Since the randomly-sampled background is unrelated to CIFAR-10 classification, our mask model ($\mathcal{M}_1$) needs to learn to ignore this background information unrelated to the task and generalize across inputs. While this increase in input dimension is spurious, it nevertheless makes L_∞ certification with RS more challenging since RS noise scales with the dimension. ARS can help in this complex setting by reducing the effective input dimension.

We make the benchmark challenging by, firstly, varying the number of positions the CIFAR-10 image can be placed on the background image. Specifically, we vary the CIFAR-10 image in either 2 pre-set locations, or against a randomly chosen location against a randomly chosen edge. Secondly, we vary σ (0.75, 0.075) in order to test our approach in different noise regimes. We fix the background dimensionality to $48 \times 48 \times 3$ ($k = 48$) for these experiments.



(a) $s = 10$ (b) $s = 20$

Figure 3. Certified test accuracy for crop margin s, where higher is harder, with $s = 10$ and $s = 20$ on the CelebA benchmark for $\sigma = 3.077$.

Results: Figure 2 shows certified test accuracy plots for $\sigma = 0.75$ and $\sigma = 0.075$. Table 1 shows the standard test accuracy results for all approaches in our settings.

We highlight three points: (1) the more positions, the harder the task, and the vanilla RS accuracy slightly degrades (from 47.5% to 45.9%). (2) on a small number of positions, a static mask provides gains, as it can systematically rule out part of the background. Appendix 7.3 shows the static masks. With more positions, there are no gains and the mask learning struggles to improve over the baseline. (3) ARS's test-time adaptivity focuses on important parts of the input (see 7.3), yielding an increase of up to 9.6% absolute over vanilla RS (see third row in Tab. 1). This shows that the mask model $\mathcal{M}_1$ generalizes to new background images.

CelebA: We choose to evaluate ARS on CelebA because it is a more realistic task. CelebA includes a variety of face annotations, and we focus on binary label 21 "mouth slightly open", because this is a well-localized attribute. We thus hypothesize that masking may reduce the input dimension without loss of accuracy. We consider the "random crop" input transformation (of same dimension as input), whereby we first add a black padding of size s then take a random crop of the same size. The larger s, the more difficult the task, as the face is less central. We use $\sigma = 3.077$.

Results: Figure 3 shows the certified accuracy in two settings (more results in appendix 7.5). We observe that, as the images shifts more (higher s), the static mask fails to learn, and the accuracy severely degrades (see masks in appendix 7.3). (2) ARS, though less precise, adapts to mask non-face pixels, for accuracy gains over vanilla RS and static mask for up to 8% percentage points (see appendix 7.5).

5. Conclusion

We introduced Adaptive Randomized Smoothing (ARS), a certification procedure for test-time adaptive models against adversarial examples that leverages RS and f-DP composition. Empirically, on our adaptivity benchmark for CIFAR-10, ARS shows an improvement of certified accuracy by up to 9.6%. More generally, we hope that our method will enable the design of a new class of certified, test-time adaptive defences against adversarial examples.

References

[1] Motasem Alfarra, Adel Bibi, Philip HS Torr, and Bernard Ghanem. Data dependent randomized smoothing. In *Uncertainty in Artificial Intelligence*, pages 64–74. PMLR, 2022. 2

[2] Avrim Blum, Travis Dick, Naren Manoj, and Hongyang Zhang. Random smoothing might be unable to certify ℓ_∞ robustness for high-dimensional images. *The Journal of Machine Learning Research*, 2020. 1, 2

[3] Yulong Cao, Ningfei Wang, Chaowei Xiao, Dawei Yang, Jin Fang, Ruigang Yang, Qi Alfred Chen, Mingyan Liu, and Bo Li. Invisible for both camera and lidar: Security of multi-sensor fusion based perception in autonomous driving under physical-world attacks. In *2021 IEEE Symposium on Security and Privacy (SP)*, 2021. 1

[4] Anirban Chakraborty, Manaar Alam, Vishal Dey, Anupam Chattopadhyay, and Debdeep Mukhopadhyay. Adversarial attacks and defences: A survey. *arXiv preprint arXiv:1810.00069*, 2018. 1

[5] Jeremy Cohen, Elan Rosenfeld, and Zico Kolter. Certified adversarial robustness via randomized smoothing. In *international conference on machine learning*, pages 1310–1320. PMLR, 2019. 1, 2, 3, 4

[6] Joana C Costa, Tiago Roxo, Hugo Proença, and Pedro RM Inácio. How deep learning sees the world: A survey on adversarial attacks & defenses. *arXiv preprint arXiv:2305.10862*, 2023. 1

[7] Francesco Croce, Sven Gowal, Thomas Brunner, Evan Shelhamer, Matthias Hein, and Taylan Cemgil. Evaluating the adversarial robustness of adaptive test-time defenses. In *International Conference on Machine Learning*, pages 4421–4435. PMLR, 2022. 2

[8] Jinshuo Dong, Aaron Roth, and Weijie J Su. Gaussian differential privacy. *arXiv preprint arXiv:1905.02383*, 2019. 2

[9] Jinshuo Dong, Aaron Roth, and Weijie J Su. Gaussian differential privacy. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 2022. 2

[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016. 4

[11] James Honaker. Efficient use of differentially private binary trees. *Theory and Practice of Differential Privacy (TPDP 2015), London, UK*, 2015. 2

[12] Alex Krizhevsky. Learning multiple layers of features from tiny images, 2009. 4

[13] Mathias Lécuyer, Vaggelis Atlidakis, Roxana Geambasu, Daniel Hsu, and Suman Jana. Certified robustness to adversarial examples with differential privacy. In *2019 IEEE symposium on security and privacy (SP)*, pages 656–672. IEEE, 2019. 1, 2

[14] Jizhizi Li, Jing Zhang, Stephen J Maybank, and Dacheng Tao. Bridging composite and real: towards end-to-end deep image matting. *International Journal of Computer Vision*, 2022. 4

[15] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, 2015. 4

[16] Apapan Pumsirirat and Yan Liu. Credit card fraud detection using deep learning based on auto-encoder and restricted boltzmann machine. *International Journal of advanced computer science and applications*, 2018. 1

[17] Peter Súkeník, Aleksei Kuvshinov, and Stephan Günnemann. Intriguing properties of input-dependent randomized smoothing. *arXiv preprint arXiv:2110.05365*, 2021. 2

[18] Greg Yang, Tony Duan, J Edward Hu, Hadi Salman, Ilya Razenshteyn, and Jerry Li. Randomized smoothing of all shapes and sizes. In *International Conference on Machine Learning*. PMLR, 2020. 2

Adaptive Randomized Smoothing for Certified Multi-Step Defence

Supplementary Material

6. Theory

6.1. Randomized Smoothing from f-DP

We reconnect RS with DP, using f-DP to yield results as strong as that of Equation (1). We start with a general robustness result on f-DP classifiers. Consider a randomized classification model $\mathcal{M} : \mathcal{X} \to \mathcal{Y}$ that is f-DP for the $B_p(r)$ neighbouring definition. Define the smoothed model $M_S : X \to \mathbb{E}(\mathcal{M}(X))$. Then the following holds:

Proposition 6.1 (f-DP Robustness)**.** *Let $\mathcal{M} : \mathcal{X} \to \mathcal{Y}$ be f-DP for $B_p(r)$ neighbourhoods, and let $M_S : X \to \arg\max_{y\in\mathcal{Y}} \mathbb{P}(\mathcal{M}(X) = y)$ be the associated smooth model. Let $y_+ \triangleq M_S(X)$ be the prediction on input X, and let $\underline{p_+}, \overline{p_-} \in [0,1]$ be such that $\mathbb{P}(\mathcal{M}(X) = y) \geq \underline{p_+} \geq \overline{p_-} \geq \max_{y'\neq y} \mathbb{P}(\mathcal{M}(X) = y')$. Then:*

$$f(1-\underline{p_+}) \geq 1-f(\overline{p_-}) \Rightarrow \forall e \in B_p(r),\ M_S(X+e) = y_+$$

Proof. Let us first consider any runner-up class y_-. Calling M the random variable for $\mathcal{M}$'s prediction, consider the rejection rule $\phi = \mathbb{1}\{M = y_-\}$, where $\mathbb{1}$ is the indicator function. Denoting $\alpha \triangleq \mathbb{E}_{\mathcal{M}(X)}(\phi)$, and using the fact that $\mathcal{M}$ is f-DP for $B_p(r)$ neighbourhoods, we have that $\forall e \in B_p(r)$:

$$\begin{aligned}\mathbb{P}(\mathcal{M}(X+e) = y_-) &= \mathbb{E}_{\mathcal{M}(X+e)}(\phi) \\ &\leq 1 - f(\alpha) \leq 1 - f(\overline{p_-}),\end{aligned} \tag{7}$$

where the last inequality is because $\alpha = \mathbb{E}_{\mathcal{M}(X)}(\phi) = \mathbb{P}(\mathcal{M}(X) = y_-) \leq \overline{p_-}$, and f is non-increasing so $f(\alpha) \geq f(\overline{p_-})$ and hence $1 - f(\alpha) \leq 1 - f(\overline{p_-})$.

Let us now consider the predicted class y_+. Keeping the same notations, and defining the rule $\phi' = \mathbb{1}\{M \neq y_+\} = 1-\mathbb{1}\{M = y_+\}$. Then $\alpha' = \mathbb{E}_{\mathcal{M}(X)}(\phi') = 1-\mathbb{P}(\mathcal{M}(X) = y_+) \leq 1-\underline{p_+}$, and $\mathbb{E}_{\mathcal{M}(X+e)}(\phi') \leq 1-f(\alpha') \leq 1-f(1-\underline{p_+})$, yielding:

$$\begin{aligned}\mathbb{P}(\mathcal{M}(X+e) = y_+) &= 1 - \mathbb{E}_{\mathcal{M}(X+e)}(\phi') \\ &\geq f(1-\underline{p_+}).\end{aligned} \tag{8}$$

Putting Equations (7) and (8) together, we have that $\mathbb{P}(\mathcal{M}(X+e) = y_+) \geq f(1-\underline{p_+}) \geq 1-f(\overline{p_-}) \geq \mathbb{P}(\mathcal{M}(X+e) = y_-)$ and thus $M_S(X+e) = y_+$. □

Let us now instantiate the Proposition 6.2 on the Gaussian based RS algorithm (see §2.1):

Proposition 6.2 (RS from f-DP)**.** *Let $\mathcal{M} : X \to M(X+z)$, $z \sim \mathcal{N}(0, \sigma^2\mathbb{I}^d)$, and $M_S : X \to \arg\max_{y\in\mathcal{Y}} \mathbb{P}(\mathcal{M}(X) = y)$ be the associated smooth model. Let $y_+ \triangleq M_S(X)$ be the prediction on input X, and let $\underline{p_+}, \overline{p_-} \in [0,1]$ be such that $\mathbb{P}(\mathcal{M}(X) = y_+) \geq \underline{p_+} \geq \overline{p_-} \geq \max_{y_-\neq y_+} \mathbb{P}(\mathcal{M}(X) = y_-)$. Then $\forall e \in B_2(r_x),\ M_S(X+e) = y_+$, with:*

$$r_X = \frac{\sigma}{2}\big(\Phi^{-1}(\underline{p_+}) - \Phi^{-1}(\overline{p_-})\big).$$

Proof. $X :\to X + z$, $z \sim \mathcal{N}(0, \sigma^2)$ is a Gaussian mechanism. By Equation (2), for any $r \geq 0$ this mechanism is $G_{\frac{r}{\sigma}}$-DP for the $B_r(r)$ neighbouring definition. By post-processing $\mathcal{M}$ is also $G_{\frac{r}{\sigma}}$-DP.

Applying Proposition 6.1, we have that $G_{\frac{r}{\sigma}}(1-\underline{p_+}) \geq 1 - G_{\frac{r}{\sigma}}(\overline{p_-}) \Rightarrow \forall e \in B_2(r),\ M_S(X+e) = y_+$. Let us find $r_X = \sup\{r : G_{\frac{r}{\sigma}}(1-\underline{p_+}) \geq 1 - G_{\frac{r}{\sigma}}(\overline{p_-})\}$. Since $G_{\frac{r}{\sigma}}(.)$ as a function of r is monotonously decreasing this will happen at $G_{\frac{r_X}{\sigma}}(1-\underline{p_+}) = 1 - G_{\frac{r_X}{\sigma}}(\overline{p_-})$, that is:

$$\begin{aligned}&\Phi\Big(\Phi^{-1}(\underline{p_+}) - \frac{r_X}{\sigma}\Big) = 1 - \Phi\Big(\Phi^{-1}(1-\overline{p_-}) - \frac{r_X}{\sigma}\Big) \\ \Rightarrow\ &\Phi^{-1}(\underline{p_+}) - \frac{r_X}{\sigma} = -\Phi^{-1}(1-\overline{p_-}) + \frac{r_X}{\sigma} \\ \Rightarrow\ &\Phi^{-1}(\underline{p_+}) - \frac{r_X}{\sigma} = \Phi^{-1}(\overline{p_-}) + \frac{r_X}{\sigma} \\ \Rightarrow\ &r_X = \frac{\sigma}{2}\big(\Phi^{-1}(\underline{p_+}) - \Phi^{-1}(\overline{p_-})\big),\end{aligned}$$

where the first implication holds because by symmetry of the standard normal $1-\Phi(x) = \Phi(-x)$, and because Φ is strictly monotonous ; the second because similarly, $\Phi^{-1}(1-p) = -\Phi^{-1}(p)$. □

6.2. Adaptive Randomized Smoothing

Proposition 2.1 (Adaptive RS). *Let $\mathcal{M} : X \to g(m_1, \ldots, m_k) \in \mathcal{Y}$, $(m_1, \ldots, m_k) \sim (\mathcal{M}_1(X), \ldots, \mathcal{M}_k(X|m_{<k}))$, and the associated smooth model $M_S : X \to \arg\max_{y\in\mathcal{Y}} \mathbb{P}(\mathcal{M}(X) = y)$. Let $y_+ \triangleq M_S(X)$ be the prediction on input X, and let $\underline{p_+}, \overline{p_-} \in [0,1]$ be such that $\mathbb{P}(\mathcal{M}(X) = y_+) \geq \underline{p_+} \geq \overline{p_-} \geq \max_{y_-\neq y_+} \mathbb{P}(\mathcal{M}(X) = y_-)$. Then $\forall e \in B_2(r_x),\ M_S(X+e) = y_+$, with:*

$$r_X = \frac{1}{2\sqrt{\sum_{i=1}^k \frac{1}{\sigma_i^2}}}\Big(\Phi^{-1}(\underline{p_+}) - \Phi^{-1}(\overline{p_-})\Big).$$

Proof. Conditioned on $m_{<i}$, and for any $r \geq 0$, mechanism $\mathcal{M}_i$ is $G_{\frac{r}{\sigma_i}}$-DP. By adaptive composition of Gaussian DP mechanisms (Equation (3)), $\mathcal{M}$ is G_μ-DP with $\mu = \sqrt{\sum_{i=1}^k \frac{r^2}{\sigma_i^2}} = r\sqrt{\sum_{i=1}^k \frac{1}{\sigma_i^2}}$. We can then apply Proposition 6.2 with $\sigma = 1/\sqrt{\sum_{i=1}^k \frac{1}{\sigma_i^2}}$. □

For Gaussian noise, Proposition 2.1 leverages strong results from DP to provide a perhaps surprising result: there is no cost to adaptivity, in the sense that k independent measurements of input X with Gaussian noise (without adaptivity) of respective variance σ_i^2 can be averaged to one measurement of variance $\sigma^2 = 1/\sum_{i=1}^k \sigma_i^{-2}$. To show this, we can use a weighted average to minimize variance (see e.g., [11], Equation 4), with $c_j = \sigma_j^{-2}/\sum_{i=1}^k \sigma_i^{-2}$ yielding $\sigma^2 = \sum_{j=1}^k c_j^2\sigma_j^2 = \sum_{j=1}^k \sigma_j^{-2}/\left(\sum_{i=1}^k \sigma_i^{-2}\right)^2 = 1/\sum_{i=1}^k \sigma_i^{-2}$. Since this is exactly the equivalent variance of an adaptive multi-step RS model shown on Proposition 2.1, that is r_X is equivalent to that of a one step RS from Proposition 6.2 with variance $\sigma^2 = 1/\sum_{i=1}^k \sigma_i^{-2}$, adaptivity over multiple steps comes with no increase in certified radius.

6.3. Adaptive RS for L_∞

Proposition 2.2 (Adaptive RS for L_∞). *Define the following pair of (adaptive) mechanisms:*

$$\begin{aligned} &\mathcal{M}_1 : X \to X + z_1 \triangleq m_1, \\ &z_1 \sim \mathcal{N}(0, \sigma_1^2 \mathbb{I}^d) \end{aligned} \tag{9}$$

Then defining $w : \mathbb{R}^d \to [0,1]^d$:

$$\begin{aligned} &\mathcal{M}_2 : X, y_1 \to w(m_1) \odot X + z_2 \triangleq m_2, \\ &z_2 \sim \mathcal{N}(0, \frac{\|w(m_1)\|_2^2}{d}\sigma_2^2 \mathbb{I}^d), \end{aligned} \tag{10}$$

where $\odot$ is the element-wise product; and the final prediction function $g : m_1, m_2 \to \mathcal{Y}$.

Consider the mechanism $\mathcal{M}$ that samples $m_1 \sim \mathcal{M}_1$, then $m_2 \sim \mathcal{M}_2$, and finally outputs $g(m_1, m_2)$; and the associated smooth classifier $M_S : X \to \arg\max_{y\in\mathcal{Y}} \mathbb{P}(\mathcal{M}(X) = y)$. Let $y_+ \triangleq M_S(X)$ be the prediction on input X, and let $\underline{p_+}, \overline{p_-} \in [0,1]$ be such that $\mathbb{P}(\mathcal{M}(X) = y_+) \geq \underline{p_+} \geq \overline{p_-} \geq \max_{y_- \neq y_+} \mathbb{P}(\mathcal{M}(X) = y_-)$. Then $\forall e \in B_\infty(r_X^\infty)$, $M_S(X + e) = y_+$, with:

$$r_X^\infty = \frac{1}{2\sqrt{d\left(\frac{1}{\sigma_1^2} + \frac{1}{\sigma_2^2}\right)}}\left(\Phi^{-1}(\underline{p_+}) - \Phi^{-1}(\overline{p_-})\right).$$

Proof. Consider any X, X' s.t. $X - X' \in B_\infty(r^\infty)$, for any r^∞,. We analyze $\mathcal{M}_1$ and $\mathcal{M}_2$ in turn. First, $\|X - X'\|_2 \leq \sqrt{d}\|X - X'\|_\infty$ and $X - X' \in B_2(\sqrt{d}r^\infty)$, $\mathcal{M}_1$ is $G_{\frac{r^\infty\sqrt{d}}{\sigma_1}}$-DP.

Second, $\|w(m_1) \odot X - w(m_1) \odot X'\|_2 = \|w(m_1) \odot (X - X')\|_2 \leq \|w(m_1)\|_2\|X - X'\|_\infty$ and $X - X' \in B_2(\|w(m_1)\|_2 r^\infty)$. Conditioned on m_1, $\mathcal{M}_2$ is thus G_{μ_2}-DP with $\mu_2 = \frac{\|w(m_1)\|_2 r^\infty}{\|w(m_1)\|_2\sigma_2/\sqrt{d}} = \frac{r^\infty\sqrt{d}}{\sigma_2}$.

Noticing that $\sqrt{\frac{(r^\infty)^2 d}{\sigma_1^2} + \frac{(r^\infty)^2 d}{\sigma_2^2}} = r^\infty\sqrt{d\left(\frac{1}{\sigma_1^2} + \frac{1}{\sigma_2^2}\right)}$ and applying Proposition 2.1 concludes the proof. □

7. Experiments

7.1. Mechanism output averaging

For a particular input pixel i, denote X_i the value of pixel, $w_i \in [0,1]$ its mask weight (we omit the explicit dependency on y_1 in w for compactness), and $m_{1,i}, m_{2,i}$ the respective values output by $\mathcal{M}_1$ and $\mathcal{M}_2$. Then, the final value of pixel i in the averaged input will be $\hat{X}_i \triangleq c_{1,i}m_{1,i} + c_{2,i}m_{2,i}$, with $c_{1,i} + w_i c_{2,i} = 1$ such that $\mathbb{E}[\hat{X}_i] = c_{1,i}X_i + c_{2,i}w_i X_i = X_i$ (i.e., the final averaged input is an unbiased estimator of X_i).

We set $c_{1,i}$ and $c_{2,i}$ to minimize the variance of $\hat{X}_i$. Notice that $\mathbb{V}[\hat{X}_i] = c_{1,i}^2\sigma_1^2 + c_{2,i}^2\|w\|_2^2\sigma_2^2$. Using the constraint that $c_{1,i} + w_i c_{2,i} = 1$, we have $\mathbb{V}[\hat{X}_i] = (1 - w_i c_{2,i})^2\sigma_1^2 + c_{2,i}^2\|w\|_2^2\sigma_2^2$: this is a convex function in $c_{2,i}$ minimized when its gradient in $c_{2,i}$ is zero, yielding:

$$c_{1,i} = \frac{\|w\|_2^2\sigma_2^2}{\sigma_1^2 w_i^2 + \|w\|_2^2\sigma_2^2}, \quad c_{2,i} = \frac{\sigma_1^2 w_i}{\sigma_1^2 w_i^2 + \|w\|_2^2\sigma_2^2}.$$

7.2. Superposition illustration

Fig.4 illustrates how we create the superpositioned input images for our benchmarks.

Figure 4. CIFAR-10 image on a BG-20k background and the ARS mask.

7.3. Masks

Fig.5, Fig.6 and Fig.7 show some samples of static and adaptive masks learnt for different settings and datasets in our benchmarks.

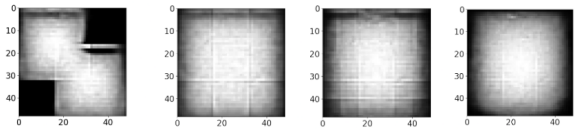


Figure 5. Static masks for 2, 4, 8 and edges locations

7.4. 5BG

We evaluate ARS on a benchmark, where instead of sampling a background image from the entire 20kBG dataset, we sample from a fixed set of 5 background images (all taken randomly from 20kBG dataset). We use the same model for our base classifier and mask model as 20kBG setup. We vary

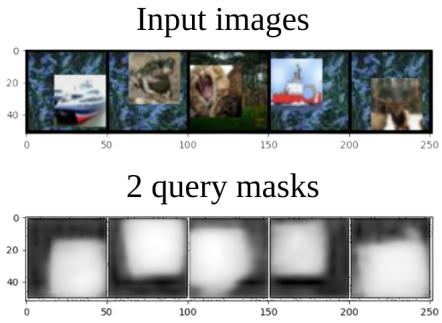


Figure 6. Adaptive masks for random (edges) locations for a random sample of 5 test inputs

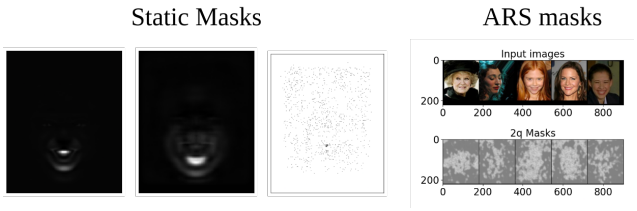


Figure 7. Static and adaptive masks for CelebA benchmark

the benchmark along 3 aspects: dimensionality, number of CIFAR-10 image locations and noise levels. The top group of Tab. 2 shows the standard test accuracy and Figure 8 shows the entire certified test accuracy curve for different settings in our **5BG** benchmark. We elaborate on each aspect of the benchmark that we vary here:

Varying input dimension (5BG). We vary the dimension of background images of dimensions $k \times k \times 3$, with $k \in \{36, 40, 48\}$. We keep the original CIFAR-10 dimension of $32 \times 32 \times 3$ when superimposing the image. To compare results across different input dimensions $d \triangleq 3k^2$, we scale σ as $\sqrt{d}$ (recall from Equation (4) that $r \propto \sigma/\sqrt{d}$). This leads to $\sigma = 0.56, 0.62, 0.75$ for $k = 36, 40, 48$, respectively. In these experiments, we use a simple setup for the location of the CIFAR-10 image on the background: we randomly sample one of two locations, either bottom right (e.g., Fig. 1), or top-left (e.g., Fig. 4).

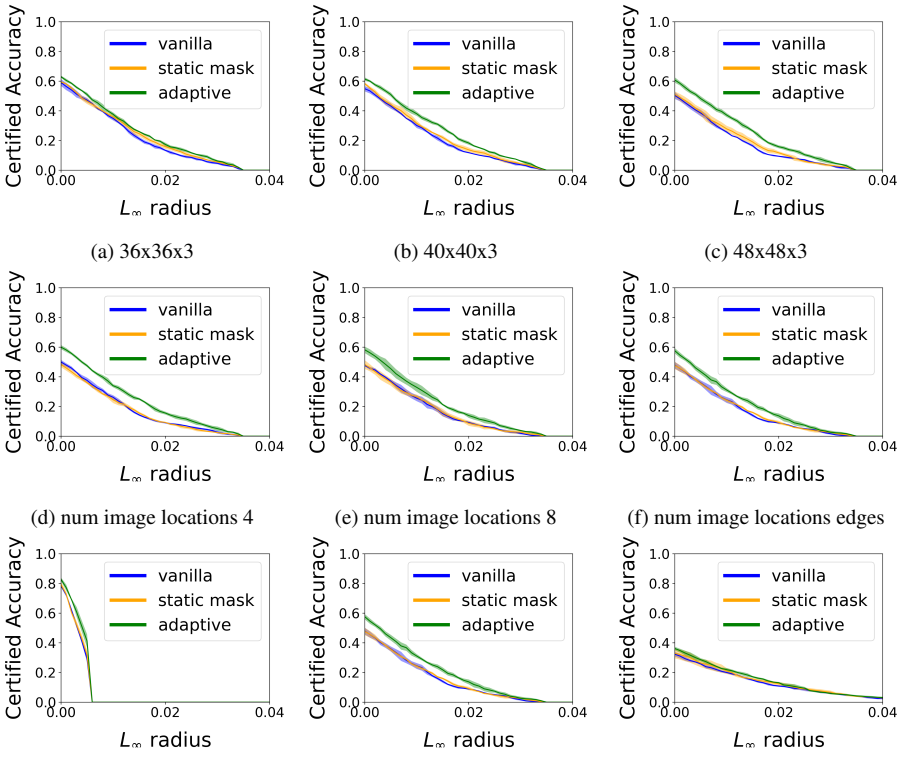
We make three observations. First, ARS always outperforms the baselines reaching an accuracy up to 10 percentage points higher than vanilla RS, a 20% improvement (see 5BG, 4 loc. in Tab. 2). Second, the larger the input dimension, the more ARS improves over both vanilla and static mask baselines. For instance, for $k = 36$ the gap between adaptive and best single query approach (static mask) is 3.6 percentage points, whereas the gap between the same approaches reaches 9.8 percentage points when $k = 48$. This is because ARS's mask is able to rule out spurious background information, reducing the noise in the second mechanism, as shown on Figure 6. Thanks to this masking, ARS is much less sensitive to increases in dimensionality, with an accuracy that remains stable whereas baselines' accuracy drops (see Tab. 2). Third, this improved clean accuracy translates to an improved certified accuracy at all certification levels. This is because ARS makes more accurate and confident predictions on more test examples, leading to a larger radius (see Proposition 2.1).

Varying the position of CIFAR-10 images (5BG). Similar to **20kBG** benchmark, in order to increase the difficulty of the mask learning task, we consider a setting with four and eight possible positions, and one where the CIFAR-10 image is positioned against a random edge, at a uniformly random position along that edge. We fix the background dimensionality to $48 \times 48 \times 3$ ($k = 48$). The total σ remains fixed to 0.75 throughout this experiment. We make three observations. First, the more positions, the harder the task, and the vanilla RS model's accuracy slightly degrades (from 50.6% to 47%). Second, on a small number of positions, a static mask provides some gains, as it can systematically rule out part of the background. Figure 5 shows the static masks. With more positions, there is no gains and the mask learning struggles, barely improving over the baseline. Third, ARS's test time adaptivity lets the model focus on important parts of the input (see Figure 6), yielding an accuracy up to 9 percentage points higher than vanilla RS.

Varying the noise levels σ (5BG). Finally, we experiment with different noise values $\sigma = 0.12, 0.25, 0.5, 1.5$ ($k = 48$, random edge locations). The best improvements are on medium noise, for which masking has a high impact (large noise) but a good mask is still possible to predict (not too large).

7.5. CelebA Benchmark

Table 3 shows the standard test accuracy for $s = 0, 10, 20$ for all 3 setups we evaluate on. Based on these results, we make one additional observation as compared to the main body of the paper. When images are well centred, i.e. $s = 0$, the static mask is able to delineate the most important pieces of the face relevant for classification task, leading to high accuracy.



(a) 36x36x3 (b) 40x40x3 (c) 48x48x3

(d) num image locations 4 (e) num image locations 8 (f) num image locations edges

(g) $\sigma = 0.12$ (h) $\sigma = 0.75$ (i) $\sigma = 1.5$

Figure 8. Certified test accuracy results for 5BG benchmark. (a), (b) and (c) correspond to 36x36x3, 40x40x3 and 48x48x3 image dimensionality for varying dimensionality experiments. (d), (e) and (f) correspond to 4, 8 and edges for varying the number of CIFAR-10 image locations experiments. (g), (h) and (i) correspond to $\sigma = 0.12$, $\sigma = 0.75$ and $\sigma = 1.5$ for varying noise level experiments

Setting/Approach	Vanilla	Static Mask	ARS
5BG, $k = 36$	59.1(.02)	59.4(.007)	**63 (.007)**
5BG, $k = 40$	54.8(.01)	58.3(.009)	**61.4 (.01)**
5BG, $k = 48$	50.3(.02)	51.0(.01)	**60.8 (.01)**
5BG, 4 loc.	50.0(.01)	48.2(.01)	**60.0 (.01)**
5BG, 8 loc.	47.7(.006)	48.3(.02)	**58.2 (.02)**
5BG, edges	47.9(.02)	47.7(.02)	**57.6 (.01)**
5BG, $\sigma = 0.12$	78.5(.01)	79.3(.01)	**82.8 (.01)**
5BG, $\sigma = 0.25$	70.9(.01)	71(.01)	**74.5 (.01)**
5BG, $\sigma = 0.5$	58.9(.01)	57.6(.04)	**64.8 (.02)**
5BG, $\sigma = 1.5$	32.3(.01)	32.9(.03)	**36.2 (.01)**
20kBG, $\sigma = .075$	83.6 (.01)	83.2 (.01)	**84.5 (.01)**
20kBG, $\sigma = .75$	44.4 (0.02)	51 (.03)	**52.9 (.01)**

Table 2. Standard test accuracy ($r = 0$) in different settings. Reported numbers are percentage points in the form: **mean (standard deviation).**

Setting/Approach	Vanilla	Static Mask	ARS
CelebA, $s = 0$	74%	**85%**	82%
CelebA, $s = 10$	64%	**65%**	64%
CelebA, $s = 20$	60%	60%	**68%**

Table 3. Standard test accuracy ($r = 0$) in different settings.