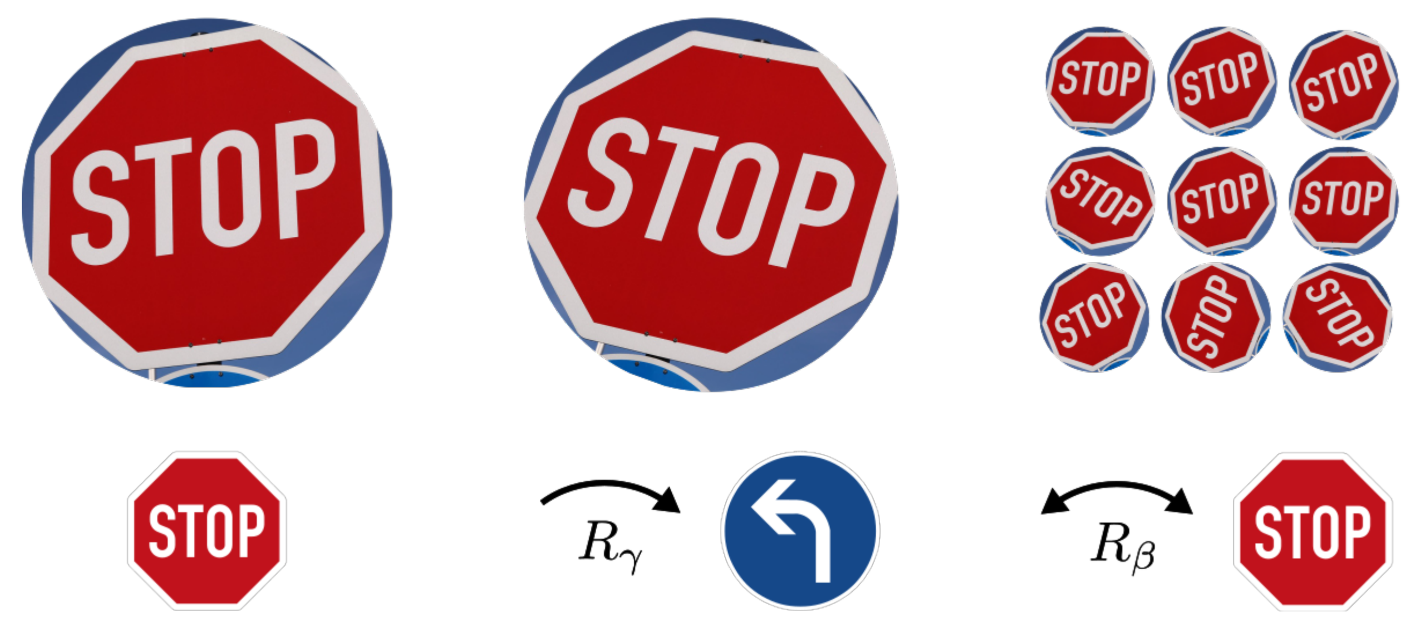
Certified Defense to Image Transformations via Randomized Smoothing

Marc Fischer,
Maximilian Baader,
Martin Vechev

ETH zürich ≡SRILAB NEURAL INFORMATION PROCESSING SYSTEMS



Randomized Smoothing for Parametric Transformations

We generalize randomized smoothing (**RS**) [Cohen et al.]

$$g(x) = \text{argmax}_{c \in \mathcal{Y}} \mathbb{P}\left(f(x+\epsilon) = c\right)$$

for classifier f, noise $\epsilon \sim \mathcal{N}(0, \sigma^2 I)$

Then $g(x+\delta) = g(x)$ for $\|\delta\|_2 \leq r_\delta$.

to randomized smoothing for parametric transformations (**SPT**):

$$g(x) = \text{argmax}_{c \in \mathcal{Y}} \mathbb{P}\left(f(\psi_\beta(x)) = c\right)$$

for classifier f, noise $\beta \sim \mathcal{N}(0, \sigma^2 I)$

Then $g(\psi_\gamma(x)) = g(x)$ for $\|\gamma\|_2 \leq r_\gamma$.

requires $\psi_\alpha \circ \psi_\beta = \psi_{\alpha+\beta}$

Heuristic best effort defense

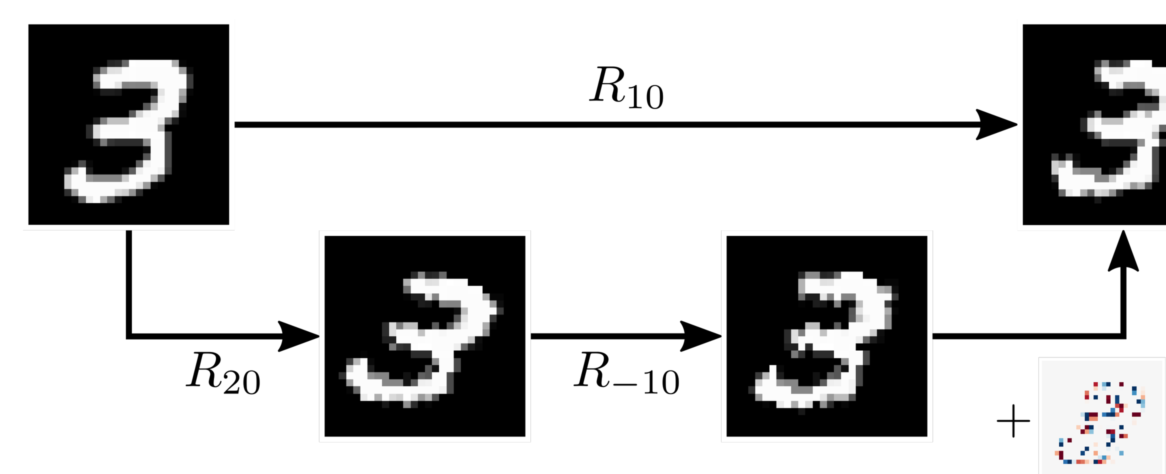
By applying SPT to image rotation we can obtain a heuristic defense as rotations don't compose as required (discussed next). Here we show results for adversarial rotations of up to 30°. In the paper we investigate the tightness of the obtained robustness radius and find counterexamples.

	acc. f	adv. acc. f	adv. acc. g
MNIST	0.99	0.73	0.99
CIFAR-10	0.91	0.26	0.85
ImageNet	0.76	0.56	0.76

Interpolation: Image Rotations don't compose

Common image transformations such as rotations or translation do not fulfill our required composition property. The reason for this are the interpolation operations employed to represent the resulting image on the pixel grid.

$$R_\alpha \circ R_\beta \neq R_{\alpha+\beta}$$

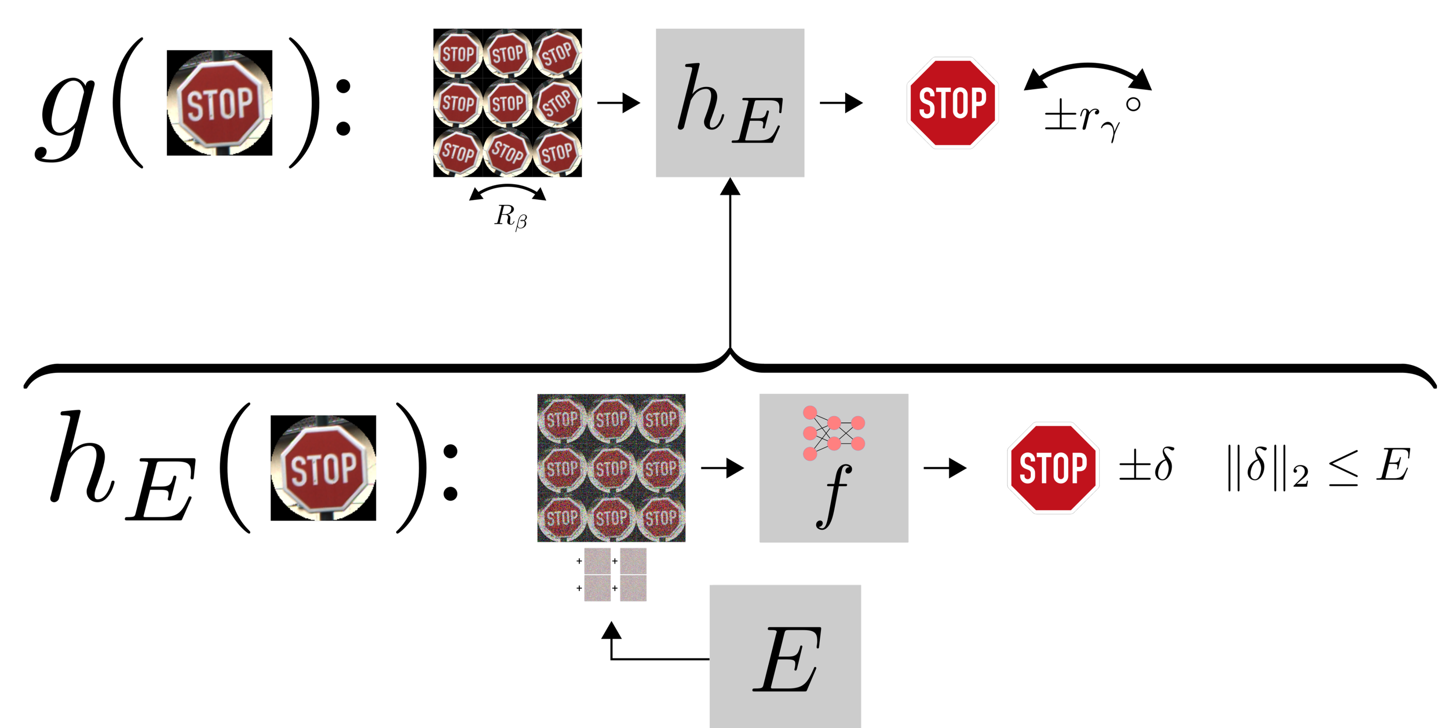


Rotation with interpolation does not compose.

Certification in the Presence of Interpolation

Over a base classifier (neural network) f we construct (via RS) a classifier h_E that is robust to the rotation error. Then via SPT we construct a classifier g that is robust to transformations (e.g., rotations).

$$\forall \beta, \gamma \in \mathbb{R}$$
$$\|R_{\beta+\gamma}(\mathbf{x}) - R_\beta \circ R_\gamma(\mathbf{x})\|_2 \leq E$$



Overview of the proposed classifier.

Computing the error bound on the training set

We obtain an error bound E on the training set, that we expect to hold for samples from the data distribution $\mathcal{D}$ with probability q_E. We use interval analysis over γ and sampling over β to derive a sound bound.

$$q_E := \mathbb{P}_{\mathbf{x} \sim \mathcal{D}} \left(\max_{\substack{\gamma \in \Gamma \\ \beta \in \mathbb{R}}} \|R_{\beta+\gamma}(\mathbf{x}) - R_\beta \circ R_\gamma(\mathbf{x})\|_2 \leq E \right)$$

	certified acc.	median r_γ
MNIST	0.99	44.90°
CIFAR-10	0.56	25.44°
ImageNet	0.23	16.17°
Restricted ImageNet	0.72	30.00°

Computing individual error bounds online

For a given possible attacked $\mathbf{x}' = R_\gamma(\mathbf{x})$ we compute E and certify $g(\mathbf{x}') = g(\mathbf{x})$, without access to γ and $\mathbf{x}$. Again we use interval analysis over γ and sampling over β to obtain the bound E.

$$\max_{\substack{\gamma \in \Gamma \\ \beta \in \mathbb{R}}} \|R_\beta(\mathbf{x}') - R_{\beta+\gamma} \circ (R_\gamma)^{-1}(\mathbf{x}')\|_2 \leq E$$

	certified acc.	$g(\mathbf{x}') = g(\mathbf{x})$ verified
MNIST	0.98	0.78

In order to calculate the above expression we need to compute the inverse rotation $(R_\gamma)^{-1}(\mathbf{x}')$. We relax this into the following set parametrized by Γ, over which we then chose the γ maximizing the outer expression:

$$(R_\Gamma)^{-1}(\mathbf{x}') := \{\mathbf{x} \mid R_\gamma(\mathbf{x}) = \mathbf{x}', \gamma \in \Gamma\}$$

We compute an overapproximating of this set by using interval analysis to invert the interpolation algorithm and obtain a lower and upper bound for each pixel in $\mathbf{x}$. By repeated application we can refine the result.

(left) Rotated Image. (middle) Lower and upper bound obtained from our inverse computation. (right) Original image.

References

J. Cohen, E. Rosenfeld, J. Kolter, (2019). "Certified Adversarial Robustness via Randomized Smoothing."
In: 36th International Conference on Machine Learning (ICML) 2019

safeai.ethz.ch

github.com/eth-sri/transformation-smoothing