# Certifying Geometric Robustness of Neural Networks

Mislav Balunović, Maximilian Baader, Gagandeep Singh, Timon Gehr, Martin Vechev

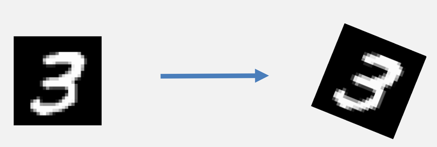
Department of Computer Science
ETH zürich

## Geometric robustness and certification

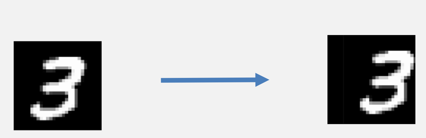
### Problem

Naturally occurring geometric transformations (e.g. rotation) can cause neural networks to misclassify images [1]:

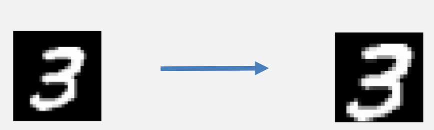
Rotation: $T_{\phi}(x,y) = (x\cos(\phi) - y\sin(\phi), x\sin(\phi) + y\cos(\phi))$



Translation: $T_{\delta_x,\delta_y}(x,y) = (x + \delta_x, y + \delta_y)$

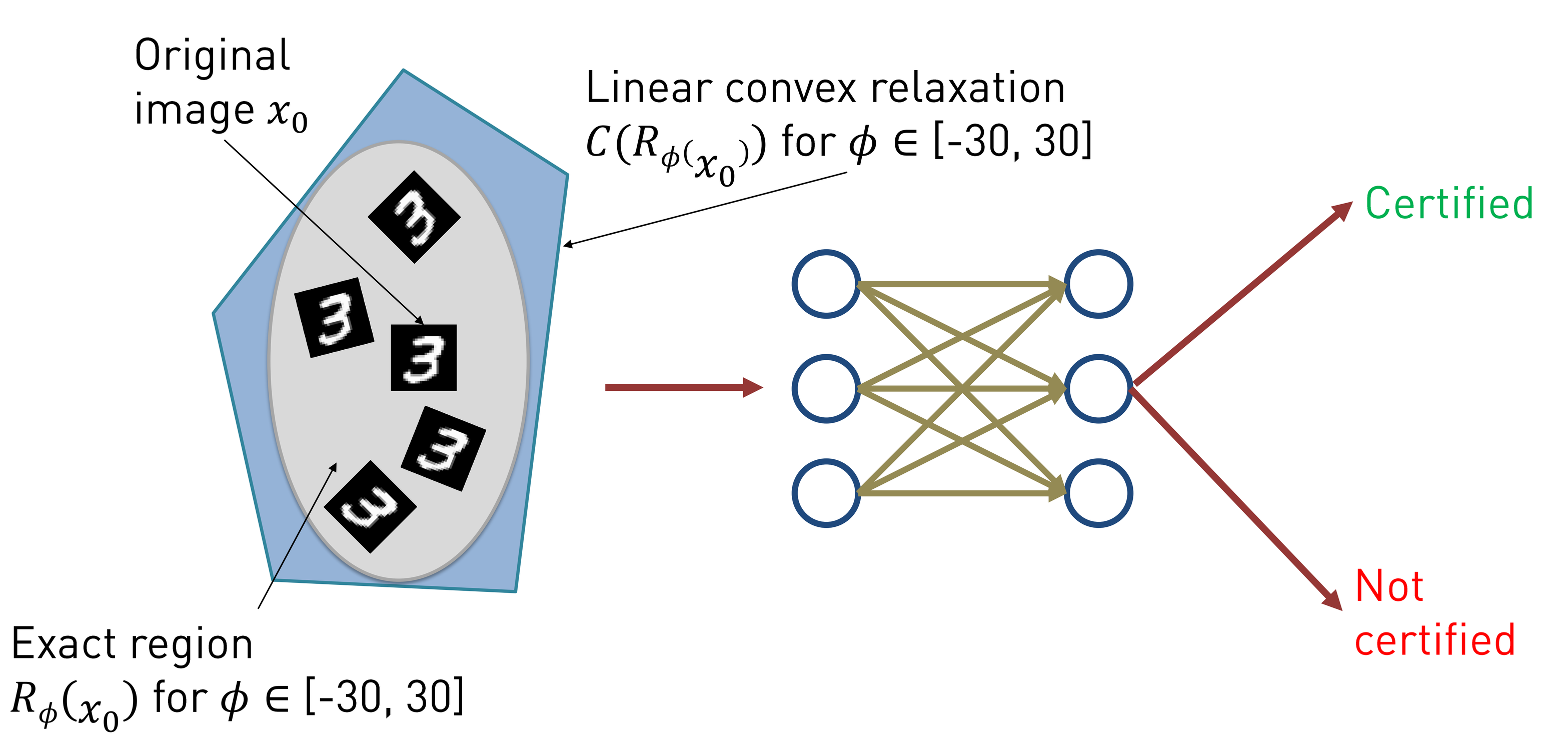


Scaling: $T_{\lambda}(x,y) = (\lambda x, \lambda y)$



Our goal is to certify that neural network correctly classifies image $I_{\boldsymbol{\kappa}}$ for each transformation parameter $\boldsymbol{\kappa} \in D$.

We build on DeepPoly [2] which requires computing linear convex relaxation capturing all possible images obtainable using specified geometric transformation.




### Optimization problem

To obtain the tight linear relaxation, our goal is to find $\boldsymbol{w}_l, b_l$ and $\boldsymbol{w}_u, b_u$ which minimize the volume

$$L(\boldsymbol{w}_l, b_l) := \int_{\boldsymbol{\kappa} \in D} \left( I_{\boldsymbol{\kappa}}(x,y) - \left(\boldsymbol{w}_l^T \boldsymbol{\kappa} + b_l\right)\right) d\boldsymbol{\kappa}$$

$$U(\boldsymbol{w}_u, b_u) := \int_{\boldsymbol{\kappa} \in D} \left( \left(\boldsymbol{w}_u^T \boldsymbol{\kappa} + b_u\right) - I_{\boldsymbol{\kappa}}(x,y)\right) d\boldsymbol{\kappa}$$

subject to the soundness constraints

$$\boldsymbol{w}_l^T \boldsymbol{\kappa} + b_l \leq I_{\boldsymbol{\kappa}}(x,y) \leq \boldsymbol{w}_u^T \boldsymbol{\kappa} + b_u, \forall \boldsymbol{\kappa} \in D.$$

## Our algorithm

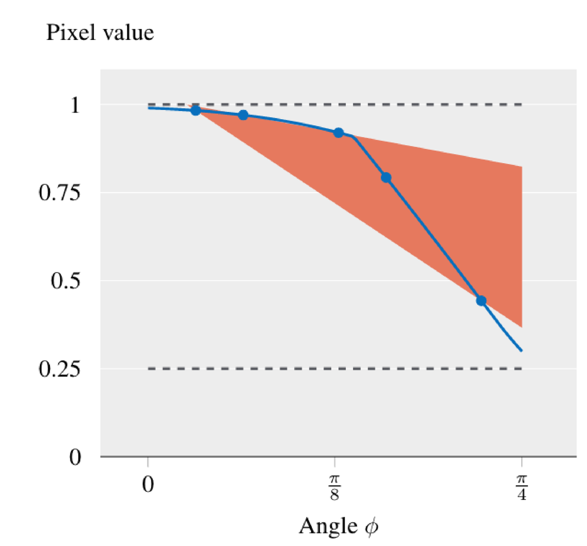
### Step 1: Approximation via Monte Carlo sampling

Replace the intractable objective with a Monte Carlo approximation and the infinite set of constraints with a finite set

$$L(\boldsymbol{w}_l, b_l) \approx \frac{1}{N} \sum_{i=1}^{N} \left( I_{\boldsymbol{\kappa}^i}(x,y) - \left(\boldsymbol{w}_l^T \boldsymbol{\kappa}^i + b_l\right)\right)$$

$$\boldsymbol{w}_l^T \boldsymbol{\kappa}^i + b_l \leq I_{\boldsymbol{\kappa}^i}(x,y)$$

We can solve the relaxed problem exactly in polynomial time using linear programming (LP) and obtain approximate solutions $\widehat{\boldsymbol{w}}_l, \hat{b}_l$ to the original problem




### Step 2: Bound the maximum violation

Next, we bound the maximum soundness violation. This requires computing an upper bound to the function $f: D \to R$,

$$f(\boldsymbol{\kappa}) = \left(\widehat{\boldsymbol{w}}_l^T \boldsymbol{\kappa} + \hat{b}_l\right) - I_{\boldsymbol{\kappa}}(x,y).$$

1) Bound $f$ by running interval propagation to obtain $l, u$ such that $f(\boldsymbol{\kappa}) \in [l, u], \forall \boldsymbol{\kappa} \in D$.

This yields an inequality:
$f(\boldsymbol{\kappa}) \leq f(\boldsymbol{\kappa}_c) + (u - f(\boldsymbol{\kappa}_c)), \forall \boldsymbol{\kappa} \in D$.

2) Bound $f$ using mean-value theorem and Lipschitz continuity:

$$f(\boldsymbol{\kappa}) = f(\boldsymbol{\kappa}_c) + 1/2 \nabla f(\boldsymbol{\kappa}')^T (\boldsymbol{\kappa} - \boldsymbol{\kappa}_c)$$
$$\leq f(\boldsymbol{\kappa}_c) + 1/2 |\boldsymbol{L}|^T (\boldsymbol{\kappa} - \boldsymbol{\kappa}_c)$$

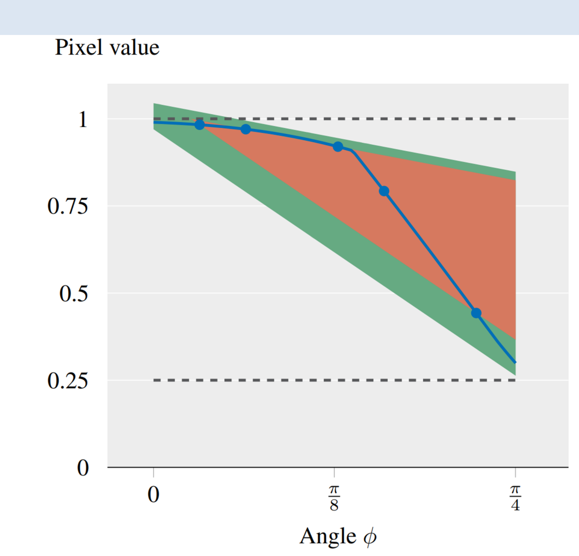
where $|\partial_i f(\boldsymbol{\kappa}')| \leq |L_i|$ for any $\boldsymbol{\kappa}' \in D$.

We also refine the bounds using branch and bound algorithm: we keep partitioning the domain into hyperrectangles as long as the obtained bound is not tight enough.

### Step 3: Sound constraints

$$\left(\widehat{\boldsymbol{w}}_l^T \boldsymbol{\kappa} + \hat{b}_l\right) - I_{\boldsymbol{\kappa}}(x,y) \leq \delta_l, \forall \boldsymbol{\kappa} \in D$$
$$I_{\boldsymbol{\kappa}}(x,y) - \left(\widehat{\boldsymbol{w}}_u^T \boldsymbol{\kappa} + \hat{b}_u\right) \leq \delta_u, \forall \boldsymbol{\kappa} \in D$$

Then, the constraints
$\boldsymbol{w}_l = \widehat{\boldsymbol{w}}_l$, $b_l = \hat{b}_l - \delta_l$ and
$\boldsymbol{w}_u = \widehat{\boldsymbol{w}}_u$, $b_u = \hat{b}_u + \delta_u$ are sound.




### Asymptotically optimal constraints

Theorem: *Let $N$ be the number of sampled points in the algorithm and $\epsilon$ tolerance in Lipschitz optimization. Let $\boldsymbol{w}_l^*, b_l^*$ be the minimum of function $L$ and $\widehat{\boldsymbol{w}}_l, \hat{b}_l$ be the constraints obtained using our method. For every $\delta$ there exists $N_\delta$ such that $|L(\boldsymbol{w}_l^*, b_l^*) - L(\widehat{\boldsymbol{w}}_l, \hat{b}_l)| < \delta + \epsilon$ for every $N > N_\delta$, with high probability. Analogous result holds for upper constraint.*

## Experimental evaluation

### Experimental evaluation

Code available at: https://github.com/eth-sri/deepg

**Properties**: Rotation, translation, scaling, shearing, brightness changes as well as compositions of these transformations.

**Networks**: 4-layer CNN with 45k neurons on CIFAR-10 dataset and 3-layer CNN on MNIST and Fashion-MNIST datasets.

| | | Accuracy (%) | Attacked (%) | Certified (%) | |
|---|---|---|---|---|---|
| | | | | Interval [9] | DeepG |
| MNIST | R(30) | 99.1 | 0.0 | 7.1 | **87.8** |
| | T(2, 2) | 99.1 | 1.0 | 0.0 | **77.0** |
| | Sc(5), R(5), B(5, 0.01) | 99.3 | 0.0 | 0.0 | **34.0** |
| | Sh(2), R(2), Sc(2), B(2, 0.001) | 99.2 | 0.0 | 1.0 | **72.0** |
| Fashion-MNIST | Sc(20) | 91.4 | 11.2 | 19.1 | **70.8** |
| | R(10), B(2, 0.01) | 87.7 | 3.6 | 0.0 | **71.4** |
| | Sc(3), R(3), Sh(2) | 87.2 | 3.5 | 3.5 | **56.6** |
| CIFAR-10 | R(10) | 71.2 | 10.8 | 28.4 | **87.8** |
| | R(2), Sh(2) | 68.5 | 5.6 | 0.0 | **54.2** |
| | Sc(1), R(1), B(1, 0.001) | 73.2 | 3.8 | 0.0 | **54.4** |

### Comparison of training techniques

We certify networks trained using different training methods:

1) Standard training
2) Training with data augmentation
3) PGD training
4) Provable defense (DiffAI)

We find that network trained using combination of data augmentation and PGD training has highest **accuracy** and highest **certification rate** with DeepG.

| | | Accuracy (%) | Attack success (%) | Certified (%) | |
|---|---|---|---|---|---|
| | | | | Interval [9] | DeepG |
| MNIST | Standard | 98.7 | 52.0 | 0.0 | 12.0 |
| | Augmented | 99.0 | 4.0 | 0.0 | 46.5 |
| | $L_\infty$-PGD | 98.9 | 45.5 | 0.0 | 20.2 |
| | $L_\infty$-DiffAI | 98.4 | 51.0 | 1.0 | 17.0 |
| | $L_\infty$-PGD + Augmented | **99.1** | **1.0** | 0.0 | **77.0** |
| | $L_\infty$-DiffAI + Augmented | 98.0 | 6.0 | **42.0** | 66.0 |

### Experiments on large networks

We certify robustness against rotations between -2 and 2 degrees.

**ResNetTiny**
- 312 000 neurons
- 91.1% certified
- 25 + 528 seconds per image

**ResNet18**
- 558 000 neurons
- 82.2% certified
- 25 + 1652 seconds per image

[1] Engstrom, Logan, Brandon Tran, Dimitris Tsipras, Ludwig Schmidt, and Aleksander Madry. "Exploring the Landscape of Spatial Robustness.", ICML 2019

[2] Singh, Gagandeep, Timon Gehr, Markus Püschel, and Martin Vechev. "An abstract domain for certifying neural networks.", POPL 2019