Countering Adversarial Images using Input Transformations

Chuan Guo, Mayank Rana, Moustapha Cisse, Laurens Van Der Maaten

Presented by

Hari Venugopalan, Zainul Abi Din
Motivation: Why is this a hard problem to solve?

- Adversarial Examples are very easy to generate and very difficult to defend against - problem with neural nets, piecewise linear functions

- Adversarial Examples transfer from one model to another very easily

- This problem is not well understood, researchers are divided on the nature of these samples.
Problem Definition: Adversarial Example

- $x$ is the original input, $x'$ is the perturbed input which is found by adding some noise to $x$.

- Given a classifier $h(x)$, the score $h(x)$ and $h(x')$ should not be the same.

- While $d(x, x')$ is smaller than some threshold.

- Keep the distortion low and move the decision boundary.

Source: Towards Evaluating the Robustness of Neural Networks, Carlini and Wagner
Measure of distortion

- Normalized L2-Dissimilarity
- X is the actual image
- X' is the adversarial image
- N is the total number of images
- Low L2-Dissimilarity

\[
\frac{1}{N} \sum_{n=1}^{N} \frac{\| x_n - x'_n \|_2}{\| x_n \|_2}.
\]

Source: Countering Adversarial Images using Input Transformations, Chuan Guo
Intuitively? What is an attack

X: Original Input

Classifier

h(X)

Y: probabilities

X': Perturbed Image

Classifier

h(X')

Y: probabilities

Image taken from: Towards Evaluating the Robustness of Neural Networks, Carlini and Wagner
Types of attacks in DNNs

- **White-Box**: The attacker is assumed to have complete knowledge of the networks weights and architecture
- **Black-Box**: Does not require internal model access
- **Gray-Box**: Attacker doesn’t know the defense strategy used.
- **Targeted Attack**: Attacker selects the class they want the example to be misclassified as.
- **Untargeted Attack**: Any misclassification is the goal

Source: ZOO: Zeroth Order Optimization Based Black-box Attacks to Deep Neural Networks without Training Substitute Models, Pi-Yu Chen
How do you find this noise? Fast Gradient Sign Method

1. $x' = x + \eta$
2. $\eta = \text{sign} \left( \nabla_x J(\theta, x, y) \right)$.

- Ian J. Goodfellow, Jonathon Shlens, Christian Szegedy - 2015

$x$

“panda”
57.7% confidence

$\text{sign}(\nabla_x J(\theta, x, y))$

“nematode”
8.2% confidence

$\epsilon \text{sign}(\nabla_x J(\theta, x, y))$

“gibbon”
99.3% confidence

$\epsilon = 0.007$
Deep fool:

- $f(x)$ is the classifier
- Given a sample $x_0$, change $f(x_0)$ sign
- Project $x_0$ orthogonally on $f(x)$
- $r = \{f(x_0) / [||\nabla x f(x_0)||^2] \nabla x f(x_0)\}$
- Keep adding $r$ to $x_0$ till sign of $f(x_0)$ changes

Source: DeepFool: A simple and accurate method to fool deep neural networks, Moosavi Dezfooli
Adversarial Transformative Networks

\[ f_w(X) \]

Learn to generate and adversarial sample for Target NN

Image Source: DeepFool: A simple and accurate method to fool deep neural networks, Moosavi Dezfooli

Concept Source: Learning to generate adversarial examples, Shumeet Baluja
ATN Cost- Gradient W.R.T $\theta$

Cost = $\sum \beta L_x(g_{f,\theta}(x), x) + L_y(f_w(g_{f,\theta}(x)), r(f_w(x), t))$

$r(f(x), t) = \text{norm}(\alpha*\max(f(x); k = t, f(x); otherwise)$

Source: Learning to generate adversarial examples, Shumeet Baluja
Carlini and Wagner Attack:

\[
\min_{x'} [ \| x - x' \|_2 + \lambda_f \max(-k, Z(x')_{h(x)} - \max\{Z(x')_k : k \neq h(x)\})]
\]

- \(Z(x')\): Input to the softmax layer
- \(Z(x')_k\): \(k\) th component of \(Z(x')\)
- \(\max\{Z(x')_k : k \neq h(x)\}\): second largest logit
- \(Z(x')_{h(x)} - \max\{Z(x')_k : k \neq h(x)\}\): Difference between second largest logit and largest logit
- \(k\): Confidence

Source: Towards Evaluating the Robustness of Neural Networks, Carlini and Wagner
Countering Adversarial Images

- Improve robustness of model
- Exploit randomness and non-differentiability
- **Model specific** and **model agnostic** defenses
Model Specific Defenses

- Based on Robust Optimization
- Minimization-maximization approach is followed
- Learning algorithm and regularization scheme
- Makes assumptions on nature of adversary
- Do not satisfy Kerckhoff's principle
Model Agnostic Defenses

- Transforms images to remove perturbations
- JPEG compression and image re-scaling are examples
- Paper aims to increase effectiveness of model agnostic defenses
Feature Squeezing

- Proposed by Xu et al
- Detects adversarial inputs
- Input space is reduced by “squeezing out” features
- Outputs are compared in original and reduced spaces
- Different outputs means adversarial inputs

Source: Feature Squeezing: Detecting Adversarial Examples in Deep Neural Networks
JPEG Compression

- Proposed by Dziugaite et al
- Removes perturbations by compressing images
- Compression could remove aspects of perturbation
- Effective against small-magnitude perturbation
- With larger perturbation, compression is unable to recover non-adversarial image

Source: A study of the effect of JPG compression on adversarial images
Total Variation Minimization

- Transformation is taken as an optimization problem
- Transformation is close to input, but also has low variation
- Inspired by noise removal

\[
\min_z \|(1 - X) \odot (z - x)\|_2 + \lambda_{TV} \cdot TV_p(z).
\]

\[
TV_p(z) = \sum_{k=1}^{K} \left[ \sum_{i=2}^{N} \|z(i, ;, k) - z(i - 1, ;, k)\|_p + \sum_{j=2}^{N} \|z(:, j, k) - z(:, j - 1, k)\|_p \right]
\]

Source: Countering Adversarial Images using Input Transformations
Image Quilting

- Images pieced together from small patches taken from database
- Non-parametric technique
- Database only has clean patches
- Patches are placed over predefined points and edges are smoothed
- Patches selected using KNN on database
- Resulting image does not have any perturbations
Experiments

- Performed in black box and gray box setting
- Performed on Imagenet dataset
- ResNet-50 model was attacked
- Attacks included FGSM, I-FGSM, Deepfool and Carlini and Wagner
- Top1 classification accuracy was reported for varying normalized L2 dissimilarities
Gray Box: Image Transformations at Test Time

Source: Countering Adversarial Images using Input Transformations
Black Box: Image transformations at test time

Source: Countering Adversarial Images using Input Transformations
# Black Box: Ensembling and model transfer

| Source: Countering Adversarial Images using Input Transformations |

## Table

<table>
<thead>
<tr>
<th>Attack Method</th>
<th>Quilting RN50</th>
<th>Quilting RN101</th>
<th>Quilting DN169</th>
<th>Quilting Iv4</th>
<th>TVM + Quilting RN50</th>
<th>TVM + Quilting RN101</th>
<th>TVM + Quilting DN169</th>
<th>TVM + Quilting Iv4</th>
<th>Cropping + TVM + Quilting RN50</th>
<th>Cropping + TVM + Quilting RN101</th>
<th>Cropping + TVM + Quilting DN169</th>
<th>Cropping + TVM + Quilting Iv4</th>
</tr>
</thead>
<tbody>
<tr>
<td>No Attack</td>
<td>70.07</td>
<td>72.56</td>
<td>70.18</td>
<td>73.01</td>
<td>72.38</td>
<td>74.74</td>
<td>73.10</td>
<td>75.55</td>
<td>72.14</td>
<td>74.53</td>
<td>72.92</td>
<td>75.10</td>
</tr>
<tr>
<td>FGSM</td>
<td>65.45</td>
<td>68.50</td>
<td>65.96</td>
<td>67.53</td>
<td>65.70</td>
<td>68.77</td>
<td>67.09</td>
<td>69.19</td>
<td>66.65</td>
<td>69.75</td>
<td>67.86</td>
<td>70.37</td>
</tr>
<tr>
<td>I-FGSM</td>
<td>65.59</td>
<td>68.72</td>
<td>66.16</td>
<td>69.29</td>
<td>65.84</td>
<td>69.10</td>
<td>67.32</td>
<td>71.05</td>
<td>67.03</td>
<td>70.14</td>
<td>68.20</td>
<td>71.52</td>
</tr>
<tr>
<td>DeepFool</td>
<td>65.20</td>
<td>68.73</td>
<td>65.86</td>
<td>68.70</td>
<td>65.80</td>
<td>69.34</td>
<td>67.40</td>
<td>71.03</td>
<td>67.11</td>
<td>70.49</td>
<td>68.62</td>
<td>71.47</td>
</tr>
<tr>
<td>CW-L2</td>
<td>64.11</td>
<td>67.72</td>
<td>65.00</td>
<td>68.14</td>
<td>63.99</td>
<td>68.20</td>
<td>66.08</td>
<td>70.13</td>
<td>65.31</td>
<td>69.14</td>
<td>66.96</td>
<td>70.50</td>
</tr>
</tbody>
</table>
Gray Box: Image transformations at test time

Source: Countering Adversarial Images using Input Transformations
Comparison to Ensemble Adversarial Training

<table>
<thead>
<tr>
<th>Attack</th>
<th>Cropping</th>
<th>TVM</th>
<th>Quilting</th>
<th>Ensemble Training (Tramèr et al., 2017)</th>
</tr>
</thead>
<tbody>
<tr>
<td>No Attack</td>
<td>65.41</td>
<td>66.29</td>
<td>69.66</td>
<td>80.3</td>
</tr>
<tr>
<td>FGSM</td>
<td>49.52</td>
<td>31.37</td>
<td>39.55</td>
<td>69.15</td>
</tr>
<tr>
<td>I-FGSM</td>
<td>43.89</td>
<td>40.99</td>
<td>33.22</td>
<td>5.07</td>
</tr>
<tr>
<td>DeepFool</td>
<td>44.92</td>
<td>44.69</td>
<td>34.54</td>
<td>1.84</td>
</tr>
<tr>
<td>CW-L2</td>
<td>41.06</td>
<td>48.41</td>
<td>30.51</td>
<td>22.23</td>
</tr>
</tbody>
</table>

Source: Countering Adversarial Images using Input Transformations
Our experiment: Running TVM with an ATN

- We built an ATN that broke an ANN from error rate 0.34 to 0.91
- Error rate went from 0.91 to 0.90 when ATN perturbation was transformed using TVM
- Dataset consisted of 20000 non-MNIST images
Conclusions and Questions

- Image transformations are a more generic defense
- Benefits from randomization
- Benefits from non-differentiability
- Why were they not successful against ATN attack?
- Like ATNs, can the best transformation be “learned”?
- How good are these transformations in complete white box settings?
Merci Beaucoup