Feature Squeezing: Detecting Adversarial Examples in Deep Neural Networks

Weilin Xu  David Evans  Yanjun Qi
Background: Classifiers are Easily Fooled

Original Example + Perturbations = Adversarial Examples

100% confidence

1

“1”

BIM

“4” 100%

CW₂

“2” 99.9%

JSMA

“2” 83.8%

Solution Strategy

Solution Strategy 1: Train a perfect vision model.
   Infeasible yet.

Solution Strategy 2: Make it harder to find adversarial examples.
   Arms race!

Feature Squeezing: A general framework that reduces the search space available for an adversary and detects adversarial examples.
Roadmap

• Feature Squeezing Detection Framework

• Feature Squeezers
  • Bit Depth Reduction
  • Spatial Smoothing

• Detection Evaluation
  • Oblivious adversary
  • Adaptive adversary
Feature Squeezer coalesces similar samples into a single one.

- Barely change legitimate input.
- Destruct adversarial perturbations.
Detection Framework: Multiple Squeezers

Input

Model

Prediction

Model

Prediction

Model

Prediction

\[ \max (d_{\downarrow 1}, d_{\downarrow 2}) > T \]

Adversarial

Yes

No

Legitimate

• Bit Depth Reduction
• Spatial Smoothing
Bit Depth Reduction

Signal Quantization

Reduce to 1-bit
\[ f_{\downarrow i} = \text{round}(f_{\downarrow i} \times 2)/2 \]

Original value

Target value

8-bit

3-bit

1-bit
Bit Depth Reduction

Eliminating adversarial perturbations while preserving semantics.
# Accuracy with Bit Depth Reduction

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Squeezer</th>
<th>Adversarial Examples (FGSM, BIM, CW_∞, Deep Fool, CW_2, CW_0, JSMA)</th>
<th>Legitimate Images</th>
</tr>
</thead>
<tbody>
<tr>
<td>MNIST</td>
<td>None</td>
<td>13.0%</td>
<td>99.43%</td>
</tr>
<tr>
<td></td>
<td>1-bit Depth</td>
<td>62.7%</td>
<td>99.33%</td>
</tr>
<tr>
<td></td>
<td><strong>Baseline</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ImageNet</td>
<td>None</td>
<td>2.78%</td>
<td>69.70%</td>
</tr>
<tr>
<td></td>
<td>4-bit Depth</td>
<td>52.11%</td>
<td>68.00%</td>
</tr>
</tbody>
</table>
Spatial Smoothing: Median Filter

- Replace a pixel with median of its neighbors.
- Effective in eliminating "salt-and-pepper" noise.

* Image from https://sultanofswing90.wordpress.com/tag/image-processing/
Spatial Smoothing: Non-local Means

• Replace a patch with weighted mean of similar patches.
• Preserve more edges.

\[ p' = \sum \hat{w}(p, q_i) \times q_i \]
Median Filter (2*2)

Non-local Means (13-3-4)
## Accuracy with Spatial Smoothing

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Squeezer</th>
<th>Adversarial Examples (FGSM, BIM, CW∞, Deep Fool, CW2, CW0)</th>
<th>Legitimate Images</th>
</tr>
</thead>
<tbody>
<tr>
<td>ImageNet</td>
<td>None</td>
<td>2.78%</td>
<td>69.70%</td>
</tr>
<tr>
<td></td>
<td>Median Filter 2*2</td>
<td>68.11%</td>
<td>65.40%</td>
</tr>
<tr>
<td></td>
<td>Non-local Means 11-3-4</td>
<td>57.11%</td>
<td>65.40%</td>
</tr>
</tbody>
</table>
Other Potential Squeezers

- Thermometer Encoding (learnable bit depth reduction)

- Image denoising using bilateral filter, autoencoder, wavelet, etc.

- Image resizing
Experimental Setup

• Datasets and Models
  MNIST, 7-layer-CNN
  CIFAR-10, DenseNet
  ImageNet, MobileNet

• Attacks (100 examples for each attack)
  • Untargeted: FGSM, BIM, DeepFool
  • Targeted (Next/Least-Likely): JSMA, Carlini-Wagner $L_2/L_\infty/L_0$

• Detection Datasets
  • A balanced dataset with legitimate examples.
  • 50% for training the detector, the remaining for validation.
Threat Models

• **Oblivious adversary**: The adversary has full knowledge of the target model, but is not aware of the detector.

• **Adaptive adversary**: The adversary has full knowledge of the target model and the detector.
Train a detector (MNIST)

Select a threshold value with FPR 5%.
Detect Successful Adv. Examples (MNIST)

Bit Depth Reduction is more effective on $L_\infty$ and $L_2$ attacks.

Median Smoothing is more effective on $L_0$ attacks.

<table>
<thead>
<tr>
<th>Squeezer</th>
<th>$L_\infty$ Attacks</th>
<th>$L_2$ Attacks</th>
<th>$L_0$ Attacks</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>FGSM</td>
<td>BIM</td>
<td>$CW_\infty$</td>
</tr>
<tr>
<td>1-bit Depth</td>
<td>100%</td>
<td>97.9%</td>
<td>100%</td>
</tr>
<tr>
<td>Median 2*2</td>
<td>73.1%</td>
<td>27.7%</td>
<td>100%</td>
</tr>
<tr>
<td>[Best Single]</td>
<td>100%</td>
<td>97.9%</td>
<td>100%</td>
</tr>
<tr>
<td>Joint</td>
<td>100%</td>
<td>97.9%</td>
<td>100%</td>
</tr>
</tbody>
</table>

Joint detection improves performance.
### Aggregated Detection Results

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Squeezers</th>
<th>Threshold</th>
<th>False Positive Rate</th>
<th>Detection Rate (SAEs)</th>
<th>ROC-AUC Exclude FAEs</th>
</tr>
</thead>
<tbody>
<tr>
<td>MNIST</td>
<td>Bit Depth (1-bit), Median (2x2)</td>
<td>0.0029</td>
<td>3.98%</td>
<td>98.2%</td>
<td>99.44%</td>
</tr>
<tr>
<td>CIFAR-10</td>
<td>Bit Depth (5-bit), Median (2x2), Non-local Mean (13-3-2)</td>
<td>1.1402</td>
<td>4.93%</td>
<td>84.5%</td>
<td>95.74%</td>
</tr>
<tr>
<td>ImageNet</td>
<td>Bit Depth (5-bit), Median (2x2), Non-local Mean (11-3-4)</td>
<td>1.2128</td>
<td>8.33%</td>
<td>85.9%</td>
<td>94.24%</td>
</tr>
</tbody>
</table>
Threat Models

• **Oblivious attack**: The adversary has full knowledge of the target model, but is not aware of the detector.

• **Adaptive attack**: The adversary has full knowledge of the target model and the detector.
Adaptive Adversary

Adaptive CW\textsubscript{2} attack, unbounded adversary.

\[
\begin{align*}
\text{minimize} & \quad \|g(x') - t\| + \lambda \Delta(x, x') \\
\text{Misclassification term} & \quad \text{Distance term} & \quad \text{Detection term}
\end{align*}
\]

Warren He, James Wei, Xinyun Chen, Nicholas Carlini, Dawn Song, 
Adversarial Example Defense: Ensembles of Weak Defenses are not Strong, USENIX WOOT’17.
Adaptive Adversarial Examples

No successful adversarial examples were found for images originally labeled as 3 or 8.

Mean $L_2$

- Untargeted: $2.80$
- Targeted (Next): $4.14$
- Targeted (LL): $4.67$
Adaptive Adversary Success Rates

Adversary’s Success Rate vs. Clipped $\varepsilon$

- Common $\varepsilon$
- Unbounded
- Untargeted
- Targeted (Next)
- Targeted (LL)

Points:
- $0.01$
- $0.06$
- $0.01$
- $0.44$
- $0.68$
Counter Measure: Randomization

- Binary filter threshold $= 0.5$
  
  $\text{threshold} := \mathcal{N}(0.5, 0.0625)$

- Strengthen the adaptive adversary
  - Attack an ensemble of 3 detectors with thresholds $= [0.4, 0.5, 0.6]$
Attack Deterministic Detector

0 1 2 3 4 5 6 7 8 9
0 1 2 3 4 5 6 7 8 9
0 1 2 3 4 5 6 7 8 9

Mean $L_2$

2.80, Untargeted
4.14, Targeted-Next
4.67, Targeted-LL

Attack Randomized Detector

0 1 2 3 4 5 6 7 8 9
0 1 2 3 4 5 6 7 8 9
0 1 2 3 4 5 6 7 8 9

3.63, Untargeted
5.48, Targeted-Next
5.76, Targeted-LL
Conclusion

• Feature Squeezing hardens deep learning models.
• Feature Squeezing gives advantages to the defense side in the arms race with adaptive adversary.
Thank you!
Reproduce our results using EvadeML-Zoo: https://evadeML.org/zoo
Backup Slides
NIPS’17 AML Defense Challenge

• Different threat model: Unknown target model and defense.
• Top 4 defense submissions:

<table>
<thead>
<tr>
<th>Username</th>
<th>Basic Idea</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>liaofz Denoise autoencoder trained with adv. examples + model ensemble</td>
<td>95.32</td>
</tr>
<tr>
<td>2</td>
<td>cihangxie Random resizing + random padding.</td>
<td>92.35</td>
</tr>
<tr>
<td>3</td>
<td>anlthms JPEG compression + random affine transformation + model ensemble.</td>
<td>91.48</td>
</tr>
<tr>
<td>4</td>
<td>erkowa 2x2 Median filter + model ensemble.</td>
<td>91.20</td>
</tr>
</tbody>
</table>

None of them is robust against adaptive adversary.