BlurNet: Defense by Filtering the Feature Maps

Adversarial Robustness Workshop

Ravi Raju and Mikko Lipasti
19 May 2020

UW Madison ECE Dept.
Adversarial defense has been mostly focused against pixel-bound adversaries but not physical ones!

- \( R^2 \) attacks are physically executed by placing stickers on road signs.
- Spectral analysis shows \( R^2 \) attacks introduce high frequency components.
- Filtering feature maps is more effective than input blurring.
- Multiple loss penalties to induce low pass filtering.

Defenses should be tailored to be defend specific classes of attacks rather than being universal.
Adversarial defense has been mostly focused against pixel-bound adversaries but not physical ones!

- $R^{2}$ attacks are physically executed by placing stickers on road signs

Filtering feature maps is more effective than input blurring

Defenses should be tailored to be defend specific classes of attacks rather than being universal.
Executive Summary

Adversarial defense has been mostly focused against pixel-bound adversaries but not physical ones!

- $RP_2$ attacks are physically executed by placing stickers on road signs
- Spectral analysis show $RP_2$ attacks introduce high frequency components
Executive Summary

Adversarial defense has been mostly focused against pixel-bound adversaries but not physical ones!

- $RP_2$ attacks are physically executed by placing stickers on road signs
- Spectral analysis show $RP_2$ attacks introduce high frequency components
- Filtering feature maps is more effective than input blurring
Adversarial defense has been mostly focused against pixel-bound adversaries but not physical ones!

- $RP_2$ attacks are physically executed by placing stickers on road signs
- Spectral analysis show $RP_2$ attacks introduce high frequency components
- Filtering feature maps is more effective than input blurring
- Multiple loss penalties to induce low pass filtering

Defenses should be tailored to be defend specific classes of attacks rather than being universal.
Introduction
Vulnerabilities in NNs
Vulnerabilities in NNs

- STOP
- Neural Network
- 25 Speed Limit Sign
- Log-shifted and normalized frequency spectrum of R channel of a stop sign image
- Lower frequencies correspond to the center and higher ones to the edge.
FFT Spectrum of Input Channels

- Log-shifted and normalized frequency spectrum of RGB channels of a natural and perturbed stop sign image
- Filtering the input channels doesn’t seem promising
FFT of First Layer Feature Maps

- Each row corresponds to a unique feature map from L1 layer of network.
FFT of First Layer Feature Maps

<table>
<thead>
<tr>
<th></th>
<th>Natural</th>
<th>Adv</th>
<th>Difference</th>
<th>Blurred</th>
</tr>
</thead>
<tbody>
<tr>
<td>Feature Map 1</td>
<td><img src="image1.png" alt="Image" /></td>
<td><img src="image2.png" alt="Image" /></td>
<td><img src="image3.png" alt="Image" /></td>
<td><img src="image4.png" alt="Image" /></td>
</tr>
<tr>
<td>Feature Map 2</td>
<td><img src="image5.png" alt="Image" /></td>
<td><img src="image6.png" alt="Image" /></td>
<td><img src="image7.png" alt="Image" /></td>
<td><img src="image8.png" alt="Image" /></td>
</tr>
</tbody>
</table>

- Each row corresponds to a unique feature map from L1 layer of network.
- Difference image suggests that the perturbations induce high frequency components not found in natural stop signs.
Background
Let $x$ be an image and a neural network, $F(x) = y$, such that $F : \mathbb{R}^{w \times h \times c} \rightarrow \mathbb{R}^n$ where $y$ is the correct label.

- Main goal of an attacker - generating an image, $x_{adv}$ such that $F(x_{adv}) \neq y$ by perturbing the input pixels.
Let $x$ be an image and a neural network, $F(x) = y$, such that $F : \mathbb{R}^{w \times h \times c} \rightarrow \mathbb{R}^n$ where $y$ is the correct label.

- Main goal of an attacker - generating an image, $x_{adv}$ such that $F(x_{adv}) \neq y$ by perturbing the input pixels.
- Main goal of a defense algorithm - classify the perturbed image with the correct label.
Let $x$ be an image and a neural network, $F(x) = y$, such that $F : \mathbb{R}^{w \times h \times c} \rightarrow \mathbb{R}^n$ where $y$ is the correct label.

- Main goal of an attacker - generating an image, $x_{adv}$ such that $F(x_{adv}) \neq y$ by perturbing the input pixels.
- Main goal of a defense algorithm - classify the perturbed image with the correct label
- Threat Model - rules which dictate what kind of information an attacker
  1. White-box attacks
Let $x$ be an image and a neural network, $F(x) = y$, such that $F : \mathbb{R}^{w \times h \times c} \rightarrow \mathbb{R}^n$ where $y$ is the correct label.

- Main goal of an attacker - generating an image, $x_{adv}$ such that $F(x_{adv}) \neq y$ by perturbing the input pixels.
- Main goal of a defense algorithm - classify the perturbed image with the correct label.
- Threat Model - rules which dictate what kind of information an attacker
  1. White-box attacks
  2. Black-box attacks (Transfer Attacks)
Let $x$ be an image and a neural network, $F(x) = y$, such that $F : \mathbb{R}^{w \times h \times c} \rightarrow \mathbb{R}^n$ where $y$ is the correct label.

- Main goal of an attacker - generating an image, $x_{adv}$ such that $F(x_{adv}) \neq y$ by perturbing the input pixels.
- Main goal of a defense algorithm - classify the perturbed image with the correct label
- Threat Model - rules which dictate what kind of information an attacker
  1. White-box attacks
  2. Black-box attacks (Transfer Attacks)
  3. Adaptive Attacks
1. Attack Success Rate - number of predictions altered by an attack, \( \frac{1}{N} \sum_{n=1}^{N} I[F(x_n) \neq F(x_{nadv})] \).
1. Attack Success Rate - number of predictions altered by an attack, \( \frac{1}{N} \sum_{n=1}^{N} \mathbb{1}[F(x_n) \neq F(x_{n\text{adv}})] \).

2. \(L_2\) Dissimilarity Distance - how different is the adversarial image from the original, \( \frac{1}{N} \sum_{n=1}^{N} \frac{||x-x_{\text{adv}}||_p}{||x||_p} \).
1. Attack Success Rate - number of predictions altered by an attack, \( \frac{1}{N} \sum_{n=1}^{N} \mathbb{I}[F(x_n) \neq F(x_{n_{adv}})] \).

2. \( L_2 \) Dissimilarity Distance - how different is the adversarial image from the original, \( \frac{1}{N} \sum_{n=1}^{N} \frac{||x-x_{adv}||_p}{||x||_p} \).

3. An attacker is considered strong if its attack success rate is high while having a low dissimilarity metric.
Attacker: Robust Physical Perturbation ($RP_2$) Attack

$\text{arg min} \; \delta \lambda \| \mathbf{M} \mathbf{x} \cdot \delta \|_p + E \mathbf{x}_i \sim \mathcal{X} V \ell (f_\theta (\mathbf{x}_i + T_i (\mathbf{M} \mathbf{x} \cdot \delta)), y^*)$

$^1$Eykholt et. al
Attacker: Robust Physical Perturbation ($RP_2$) Attack

\[
\arg\min_\delta \lambda \|M_x \cdot \delta\|_p + \mathbb{E}_{x_i \sim X} \ell(f_\theta(x_i + T_i(M_x \cdot \delta)), y^*)
\]

\(^1\)Eykholdt et. al
Dataset and Model

- Victim model 4-layer CNN
- LISA traffic sign dataset - consider the top 18 classes only\(^2\)
- Attacks focused on victim set of 40 stop signs

\(^2\)Mogelmose et. al
Low-pass filtering the feature maps
Induce low-pass filtering with standard blur kernels by inserting a depthwise convolution after the first convolution layer.

\[ K = \frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \]
Filtering input vs. Filtering Feature Maps

Table 1: Results from black box evaluation

<table>
<thead>
<tr>
<th></th>
<th>Accuracy</th>
<th>Attack Success Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline</td>
<td>100%</td>
<td>90%</td>
</tr>
<tr>
<td>Input filter 3x3</td>
<td>100%</td>
<td>87.5%</td>
</tr>
<tr>
<td>Input filter 5x5</td>
<td>100%</td>
<td>67.5%</td>
</tr>
<tr>
<td>3x3 filter on L1 feature maps</td>
<td>100%</td>
<td>65%</td>
</tr>
<tr>
<td>5x5 filter on L1 feature maps</td>
<td>87.5%</td>
<td>17.5%</td>
</tr>
</tbody>
</table>

Filtering features maps gives larger gains but trades off accuracy as filter width increases.
Filtering in the higher layers

1. The FFT Spectrum of a subsampling of feature maps from the second layer of the network.
Filtering in the higher layers

1. The FFT Spectrum of a subsampling of feature maps from the second layer of the network.

2. High frequency information is relevant to maintain decent classification.
Filtering in the higher layers

1. The FFT Spectrum of a subsampling of feature maps from the second layer of the network.

2. High frequency information is relevant to maintain decent classification.

Filtering at the layers of the network does not make sense since spatial information is not preserved.
Learning the Filter Parameters
Minimizing accuracy loss

• Can we learn the filter parameters so that we gain low-pass filtering without a significant degradation in accuracy?
Minimizing accuracy loss

- Can we learn the filter parameters so that we gain low-pass filtering without a significant degradation in accuracy?
- Regularization
  1. $L_\infty$ regularization of the depthwise convolution layer
Can we learn the filter parameters so that we gain low-pass filtering without a significant degradation in accuracy?

Regularization

1. $L_\infty$ regularization of the depthwise convolution layer
2. Total Variation (TV) regularization of the Layer 1 feature maps
Minimizing accuracy loss

• Can we learn the filter parameters so that we gain low-pass filtering without a significant degradation in accuracy?

• Regularization
  1. $L_\infty$ regularization of the depthwise convolution layer
  2. Total Variation (TV) regularization of the Layer 1 feature maps
  3. Generalized Tikhonov regularization of the Layer 1 feature maps

• Train all models with Adam with learning rate $10^{-3}$
$L_\infty$ Regularization

$$\min \frac{\alpha}{K} \sum_{j=1}^{K} \| W_{\text{depthwise}}[:, :, j] \|_\infty + \ell(f_\theta(x, y))$$
$L_\infty$ Regularization

$$\min \alpha \frac{1}{K} \sum_{j=1}^{K} \| W_{\text{depthwise}}[:, :, j] \|_\infty + \ell(f_\theta(x, y))$$

$L_\infty$ norm is a natural choice for the depthwise weights. This will ensure that the weights in the kernel take similar values to each much like a low pass filter.

$$K = \frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$
Total Variation Regularization

\[ TV(x) = \sum_{i,j} |x_{i+1,j} - x_{i,j}| + |x_{i,j+1} - x_{i,j}|. \]
Total Variation Regularization

\[ TV(x) = \sum_{i,j} |x_{i+1,j} - x_{i,j}| + |x_{i,j+1} - x_{i,j}|. \]

\[ \min_\alpha \frac{1}{N \cdot K} \sum_{i=1}^{N} \sum_{k=1}^{K} TV(\mathcal{F}[i, :, :, k]) + \ell(f_\theta(x, y)), \]
Total Variation Regularization

$TV(x) = \sum_{i,j} |x_{i+1,j} - x_{i,j}| + |x_{i,j+1} - x_{i,j}|.$

$$\min \alpha TV \frac{1}{N \cdot K} \sum_{i=1}^{N} \sum_{i=1}^{K} TV(\mathcal{F}[i, :, :, k]) + \ell(f_\theta(x, y)),$$

TV encourages the neighboring values in the feature maps to be similar so the high spike introduced by $RP_2$ would be diminished.
Tikhonov Regularization

\[ \min \mu \| L \cdot w_0 \|_2 + \ell(f_\theta(x, y, w_0)), \]
Tikhonov Regularization

\[
\begin{align*}
\min & \quad \mu \| L \cdot w_0 \|_2 + \ell(f_\theta(x, y, w_0)), \\
\min & \quad \alpha_{hf} \frac{1}{N \cdot K} \sum_{i=1}^{N} \sum_{i=1}^{K} \| L_{hf} \cdot F[i, :, :, k] \|_2 + \ell(f_\theta(x, y))
\end{align*}
\]

L is a regularization operator \( L_{hf} = I - L_{avg} \), defense is referred as Tik_{hf}
Tikhonov Regularization

\[
\min \mu \| L \cdot w_0 \|_2 + \ell(f_\theta(x, y, w_0)),
\]

\[
\min \alpha_{hf} \frac{1}{N \cdot K} \sum_{i=1}^{N} \sum_{i=1}^{K} \| L_{hf} \cdot F[i, :, :, k] \|_2 + \ell(f_\theta(x, y))
\]

\[L\] is a regularization operator \( L_{hf} = I - L_{avg} \), defense is referred as \( \text{Tik}_{hf} \)

\[
\min \alpha_{pseudo} \frac{1}{N \cdot K} \sum_{i=1}^{N} \sum_{i=1}^{K} \| L_{diff}^+ \cdot F[i, :, :, k] \|_2 + \ell(f_\theta(x, y))
\]

We refer to this defense as \( \text{Tik}_{pseudo} \)
Total Variation Regularization and both variants of Tikhonov Regularization are able to perform well against a white-box $RP_2$ adversary.
Table 2: Results from white box evaluation

<table>
<thead>
<tr>
<th>Method</th>
<th>$\alpha$</th>
<th>Legitimate Acc.</th>
<th>Average Success Rate</th>
<th>Worst Success Rate</th>
<th>$L_2$ Dissimilarity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline</td>
<td>0</td>
<td>91%</td>
<td>49.18%</td>
<td>90%</td>
<td>0.207</td>
</tr>
<tr>
<td>Gaussian aug ($\sigma = 0.1$)</td>
<td>-</td>
<td>84.3%</td>
<td>19.44%</td>
<td>62.5%</td>
<td>0.238</td>
</tr>
<tr>
<td>Gaussian aug ($\sigma = 0.2$)</td>
<td>-</td>
<td>84.4%</td>
<td>55.97%</td>
<td>80%</td>
<td>0.196</td>
</tr>
<tr>
<td>Gaussian aug ($\sigma = 0.3$)</td>
<td>-</td>
<td>85.6%</td>
<td>21.39%</td>
<td>25%</td>
<td>0.198</td>
</tr>
<tr>
<td>Rand. sm ($\sigma = 0.1$)</td>
<td>-</td>
<td>84.3%</td>
<td>19.3%</td>
<td>67.5%</td>
<td>0.236</td>
</tr>
<tr>
<td>Rand. sm ($\sigma = 0.2$)</td>
<td>-</td>
<td>84.4%</td>
<td>55%</td>
<td>70%</td>
<td>0.189</td>
</tr>
<tr>
<td>Rand. sm ($\sigma = 0.3$)</td>
<td>-</td>
<td>85.6%</td>
<td>22.5%</td>
<td>22.5%</td>
<td>0.198</td>
</tr>
<tr>
<td>Adv-train</td>
<td>-</td>
<td>77.9%</td>
<td>11.94%</td>
<td>20%</td>
<td>0.244</td>
</tr>
<tr>
<td>3x3 conv</td>
<td>$10^{-4}$</td>
<td>86.3%</td>
<td>30%</td>
<td>55%</td>
<td>0.201</td>
</tr>
<tr>
<td>5x5 conv</td>
<td>0.1</td>
<td>86.3%</td>
<td>24.11%</td>
<td>47.5%</td>
<td>0.189</td>
</tr>
<tr>
<td>7x7 conv</td>
<td>0.1</td>
<td>87%</td>
<td>11.61%</td>
<td>30%</td>
<td>0.203</td>
</tr>
<tr>
<td>TV</td>
<td>$10^{-4}$</td>
<td>85.6%</td>
<td>7.92%</td>
<td>17.5%</td>
<td>0.224</td>
</tr>
<tr>
<td>TV</td>
<td>$10^{-5}$</td>
<td>82.3%</td>
<td>8.47%</td>
<td>22.5%</td>
<td>0.199</td>
</tr>
<tr>
<td>Tik$_{hf}$</td>
<td>$10^{-4}$</td>
<td>84.5%</td>
<td>5.416%</td>
<td>10%</td>
<td>0.214</td>
</tr>
<tr>
<td>Tik$_{pseudo}$</td>
<td>$10^{-6}$</td>
<td>83.6%</td>
<td>13.9%</td>
<td>35%</td>
<td>0.222</td>
</tr>
</tbody>
</table>

TV and Tik$_{hf}$ are able to outperform norm-bound defenses like adversarial training and randomized smoothing.
Include the regularizer into attacker’s loss function to reveal knowledge of defense

Low Frequency attack: Basic idea is restrict the search space of adversary so perturbations added are low-frequency noise

\[
\arg \min_{\delta} \lambda \| M_x \cdot \delta \|_p + \ell(f_{\theta}(x_i + T_i(IDCT(M_{\text{dim}} \cdot DCT(M_x \cdot \delta))))), y^*)
\]
Adaptive Attacks

\[
\arg\min_{\delta} \ell_{adv} + \frac{1}{N \cdot K} \sum_{i=1}^{N} \sum_{i=1}^{K} TV(\mathcal{F}[i, :, :, k])
\]

\[
\arg\min_{\delta} \ell_{adv} + \frac{1}{N \cdot K} \sum_{i=1}^{N} \sum_{i=1}^{K} \|L_{hf} \cdot \mathcal{F}[i, :, :, k]\|_2,
\]

\[
\arg\min_{\delta} \ell_{adv} + \frac{1}{N \cdot K} \sum_{i=1}^{N} \sum_{i=1}^{K} \|L_{diff}^+ \cdot \mathcal{F}[i, :, :, k]\|_2.
\]

Add the regularizer for TV, \(Tik_{hf}\), \(Tik_{pseudo}\) to the attacker's loss function and rerun evaluation.
## Adaptive Attack Evaluation

<table>
<thead>
<tr>
<th></th>
<th>Average Success Rate</th>
<th>Worst Success Rate</th>
<th>$L_2$ Dissimilarity</th>
</tr>
</thead>
<tbody>
<tr>
<td>3x3 conv</td>
<td>22.91%</td>
<td>52.5%</td>
<td>0.546</td>
</tr>
<tr>
<td>5x5 conv</td>
<td>46.25%</td>
<td>75%</td>
<td>0.539</td>
</tr>
<tr>
<td>7x7 conv</td>
<td>10.416%</td>
<td>20%</td>
<td>0.539</td>
</tr>
<tr>
<td>TV ($10^{-4}$)</td>
<td>8.33%</td>
<td><strong>20%</strong></td>
<td>0.044</td>
</tr>
<tr>
<td>TV ($10^{-5}$)</td>
<td>6.11%</td>
<td>25%</td>
<td>0.046</td>
</tr>
<tr>
<td>Tik$_{hf}$</td>
<td>23.6%</td>
<td>47.5%</td>
<td>0.147</td>
</tr>
<tr>
<td>Tik$_{pseudo}$</td>
<td>17.5%</td>
<td>45%</td>
<td>0.141</td>
</tr>
</tbody>
</table>

Total Variation Regularization is a robust against physical-based attackers like $RP_2$. 
Success rate after increased only by 5% after changing mask dimension for conv7x7 showing low pass filtering feature maps is a promising defense against physical attacks.
Concluding Remarks and Future Work
Conclusion/Future Work

- Spectral analysis of the feature maps and $RP_2$ attacks introduce high-frequency components

Future Work

- Generalize this idea to larger datasets and models
- Combine our regularization terms with methods like adversarial training

Published at: DSN Workshop in Dependable and Secure Machine Learning.

Link: https://arxiv.org/abs/1908.02256
Conclusion/Future Work

- Spectral analysis of the feature maps and $R_P^2$ attacks introduce high-frequency components.
- Adding low-pass filters after the first layers provides robustness benefits.

Future Work

- Generalize this idea to larger datasets and models.
- Combine our regularization terms with methods like adversarial training.

• Spectral analysis of the feature maps and $RP_2$ attacks introduce high-frequency components
• Adding low-pass filters after the first layers provides robustness benefits
• Introduced multiple loss penalties to induce low pass filtering

Future Work
Conclusion/Future Work

- Spectral analysis of the feature maps and $RP_2$ attacks introduce high-frequency components
- Adding low-pass filters after the first layers provides robustness benefits
- Introduced multiple loss penalties to induce low pass filtering

Future Work

- Generalize this idea to larger datasets and models
Conclusion/Future Work

• Spectral analysis of the feature maps and $RP_2$ attacks introduce high-frequency components
• Adding low-pass filters after the first layers provides robustness benefits
• Introduced multiple loss penalties to induce low pass filtering

Future Work

• Generalize this idea to larger datasets and models
• Combine our regularization terms with methods like adversarial training

Published at: DSN Workshop in Dependable and Secure Machine Learning.
Thank you for listening!