A Broader Impact

Adversarial attacks have been investigated a lot and people are worried that the vulnerability of machine learning models may affect their application in real world. This paper, as a potential countermeasure against the adversarial attacks, proposes a type of model architecture that can guarantee the model robustness under certain adversarial attacks. This could help to secure the safe deployment of ML models in real applications, thus promoting the development of various ML systems. In addition, better understanding of the Lipschitz property of machine learning models may help people understand and explain how the models work, which is a key concern when applying these algorithms in practice. On the other hand, the misuse of our technique may also lead to negative impact. For example, if an attacker is fully familiar with this work, he/she may discover some specific attack to fool this model (beyond ℓ_2 attacks which we can certify against). Therefore, in real applications, it is recommended that different defense techniques are applied together to secure the model safety.

B Pseudocode of LOT Layer

We show the detailed pseudocode of our LOT layer in Algorithm 1.

Algorithm 1 LOT layer.

Require: Unconstrained convolution kernel $V \in \mathbb{R}^{c_{out} \times c_{in} \times k \times k}$; Input tensor $X \in \mathbb{R}^{c_{in} \times w \times w}$.

$X^{pad} = \text{zero\_pad}(X, (k, k, k, k)) \in \mathbb{R}^{c_{in} \times (w+2k) \times (w+2k)}$.
$V^{pad} = \text{zero\_pad}(W, (0, 0, k + w, k + w)) \in \mathbb{R}^{c_{out} \times c_{in} \times (w+2k) \times (w+2k)}$.
// Calculate the Fourier transformation:
for all $i \in \{1, \dots, c_{in}\}$ **do**
  $\tilde{X}_i = \text{FFT}(X_i^{pad})$.
  for all $j \in \{1, \dots, c_{out}\}$ **do**
    $\tilde{V}_{j,i} = \text{FFT}(V_{j,i}^{pad})$.
  end for
end for
// Calculate the output on frequency domain:
for all $a, b \in \{1, \dots, w + 2k\}$ **do**
  $\hat{V} = \frac{\tilde{V}_{:,:,a,b}}{\sqrt{||\tilde{V}_{:,:,a,b}\tilde{V}^{\intercal}_{:,:,a,b}||_F}}$ // Rescale $\tilde{V}$.
  Calculate $\tilde{W}_{:,:,a,b} = (\hat{V}\hat{V}^*)^{-\frac{1}{2}}\hat{V}$ with Newton's iteration.
  $\tilde{Y}_{:,a,b} = \tilde{W}_{:,:,a,b}\tilde{X}_{:,a,b}$.
end for
// Get the final output:
for all $i \in \{1, \dots, c_{out}\}$ **do**
  $Y_i = \text{FFT}^{-1}(\tilde{Y}_i)$.
end for
return $(Y_{:,k:w+k,k:w+k}).real$

C Proof of Theorem 5.1 and Theorem 5.2

C.1 Proof

To prove Theorem 5.1 and Theorem 5.2, we first define the loss on a subset, so that we can divide the loss into different subset losses.

Definition C.1 (Subset loss)**.** Define conditional loss:

$$L_{\mathsf{m}}^{cond}(G, G^*|S) = E_{x \in S}[G^*(x)(1 - G(x)) + (1 - G^*(x))G(x)]$$

to bethe loss function calculated only over the subset $S \subseteq \mathcal{X}$, and define the subset loss:

$$L_{\mathsf{m}}(G, G^*|S) = L_{\mathsf{m}}^{cond}(G, G^*|S) \cdot P(S).$$

A property of the subset loss is that, if $S = S_1 \cup S_2$ where S_1 and S_2 are disjoint, then $L_{\mathsf{m}}(G, G^*|S) = L_{\mathsf{m}}(G, G^*|S_1) + L_{\mathsf{m}}(G, G^*|S_2)$.

First, we provide some facts on the relationship between subset loss $L_{\mathsf{m}}(G, G^*|S)$ and $L_{\mathsf{m}}(G, G_{pl}|S)$ when $S \subseteq \mathcal{M}(G_{pl})$ or $S \subseteq \overline{\mathcal{M}(G_{pl})}$.

Fact C.1. *When $S \subseteq \overline{\mathcal{M}(G_{pl})}$, then $L_{\mathsf{m}}(G, G^*|S) = L_{\mathsf{m}}(G, G_{pl}|S)$; when $S \subseteq \mathcal{M}(G_{pl})$, then $L_{\mathsf{m}}(G, G^*|S) + L_{\mathsf{m}}(G, G_{pl}|S) = P(S)$.*

Proof. This is easy to see by noticing that when $x \in \overline{\mathcal{M}(G_{pl})}$, then $G_{pl}(x) = G^*(x)$; when $x \in \mathcal{M}(G_{pl})$, then $G_{pl}(x) \neq G^*(x)$, so $G_{pl}(x) + G^*(x) = 1$. □

Now, we define $S_1 = S_B^{\mathsf{m}}(G) \cap \mathcal{M}(G_{pl})$ and $S_2 = S_B^{\mathsf{m}}(G) \cap \overline{\mathcal{M}(G_{pl})}$. Base on the fact, we will be able to derive the relationship between $L_{\mathsf{m}}(G, G^*|S_1)$ and $L_{\mathsf{m}}(G, G^*|S_2)$:

Lemma C.2. *We have the following relationship between the losses on the sets in which G_{pl} is correct vs. G_{pl} is wrong:*

$$L_{\mathsf{m}}(G, G^*|S_2) + P(S_1) = L_{\mathsf{m}}(G, G^*|S_1) + L_{\mathsf{m}}(G, G_{pl}|S_B^{\mathsf{m}}(G))$$

And therefore,

$$L_{\mathsf{m}}(G, G^*|S_2) \leq L_{\mathsf{m}}(G, G^*|S_1) + L_{\mathsf{m}}(G, G_{pl}) - \mathit{Err}(G_{pl}) + R_B^{\mathsf{m}}(G)$$

Proof. For the first equation, note that $S_2 \subseteq \overline{\mathcal{M}(G_{pl})}$ and $S_1 \subseteq \mathcal{M}(G_{pl})$, so based on the previous fact, we have:

$$\begin{aligned}
&L_{\mathsf{m}}(G, G^*|S_2) + P(S_1) \\
=&L_{\mathsf{m}}(G, G^*|S_2) + L_{\mathsf{m}}(G, G_{pl}|S_1) + L_{\mathsf{m}}(G, G^*|S_1) \\
=&L_{\mathsf{m}}(G, G_{pl}|S_2) + L_{\mathsf{m}}(G, G_{pl}|S_1) + L_{\mathsf{m}}(G, G^*|S_1) \\
=&L_{\mathsf{m}}(G, G_{pl}|S_B^{\mathsf{m}}(G)) + L_{\mathsf{m}}(G, G^*|S_1)
\end{aligned}$$

For the second inequation, notice $\mathcal{M}(G_{pl}) \backslash \overline{S_B^{\mathsf{m}}(G)} \subseteq S_1$, so $P(S_1) \geq P(\mathcal{M}(G_{pl})) - P(\overline{S_B^{\mathsf{m}}(G)}) = \text{Err}(G_{pl}) - R_B^{\mathsf{m}}(G)$. So:

$$\begin{aligned}
L_{\mathsf{m}}(G, G^*|S_2) =&L_{\mathsf{m}}(G, G^*|S_1) + L_{\mathsf{m}}(G, G_{pl}|S_B^{\mathsf{m}}(G)) - P(S_1) \\
\leq&L_{\mathsf{m}}(G, G^*|S_1) + L_{\mathsf{m}}(G, G_{pl}) - (\text{Err}(G_{pl}) - R_B^{\mathsf{m}}(G)) \\
=&L_{\mathsf{m}}(G, G^*|S_1) + L_{\mathsf{m}}(G, G_{pl}) - \text{Err}(G_{pl}) + R_B(G)
\end{aligned}$$

□

Fact C.3. *Given $\delta \in [0, \frac{1}{c}]$ and $\beta \in (0, \frac{c-1}{2}]$, we can verify that:*

$$\frac{\beta - 1}{c(1-\delta) - 2} \leq \frac{\beta}{c-1}$$

Proof. This can be verified by substituting $\delta = \frac{1}{c}$ into LHS, noticing that LHS is monotonically increasing w.r.t. δ. □

Lemma C.4. *For any $\beta \in (0, \frac{c-1}{2}]$, define $q = \frac{\beta}{c-1}\mathit{Err}(G_{pl})$ and $\alpha = (\beta - 1)\mathit{Err}(G_{pl})$. If G fits the pseudolabels with suffcient accuracy and consistency:*

$$L_{\mathsf{m}}(G, G_{pl}) + 2R_B^{\mathsf{m}}(G) \leq \mathit{Err}(G_{pl}) + \alpha$$

Then G satisfies the following error bound:

$$\mathit{Err}_{\mathsf{m}}(G) \leq 2(q + R_B^{\mathsf{m}}(G)) + L_{\mathsf{m}}(G, G_{pl}) - \mathit{Err}(G_{pl})$$

The intuition of the proof is as follows. Lemma C.2 provides a relationship between the loss $L_{\mathsf{m}}(G, G^*|S_1)$ and $L_{\mathsf{m}}(G, G^*|S_2)$. On the other hand, the expansion of S_1 is also related with S_2 by $(\mathcal{N}(S_1) \backslash S_1) \cap S_B^{\mathsf{m}}(G) \subseteq S_2$. Note that the expansion $P(\mathcal{N}(S_1)) > c \cdot P(S_1)$, so S_1 cannot be too large or otherwise $\mathcal{N}(S_1) \backslash S_1$ will be too large to be within S_2. We will show that $L_{\mathsf{m}}(G, G^*|S_1) < q$.

Proof. Consider the expansion of S_1, $\mathcal{N}(S_1)$. Since $S_1 \subseteq \mathcal{M}(G_{pl})$ and $P(G_{pl}) < 1/c$, we know that $c \cdot P(S_1) < 1$, so by the assumption of expansion, $P(\mathcal{N}(S_1)) \geq c \cdot P(S_1)$. In addition, notice that $(\mathcal{N}(S_1) \backslash S_1) \cap S_B^{\mathsf{m}}(G) \subseteq S_2$. Therefore, we have:

$$\begin{aligned} L_{\mathsf{m}}(G, G^*|S_2) \geq & L_{\mathsf{m}}(G, G^*|(\mathcal{N}(S_1) \backslash S_1) \cap S_B^{\mathsf{m}}(G)) \\ \geq & L_{\mathsf{m}}(G, G^*|\mathcal{N}(S_1) \cap S_B^{\mathsf{m}}(G)) - L_{\mathsf{m}}(G, G^*|S_1) \end{aligned}$$

For the first term, we notice that $\mathcal{N}(S_1) \cap S_B^{\mathsf{m}}(G) \subseteq \mathcal{N}(S_1)$ and $S_1 \subseteq S_B^{\mathsf{m}}(G)$, so the conditional loss satisfies $L(G, G^*|\mathcal{N}(S_1) \cap S_B^{\mathsf{m}}(G))/P(\mathcal{N}(S_1) \cap S_B^{\mathsf{m}}(G)) \geq (1-\delta)L(G, G^*|S_1)/P(S_1)$. Therefore,

$$\begin{aligned} L_{\mathsf{m}}(G, G^*|S_2) \geq & L_{\mathsf{m}}(G, G^*|\mathcal{N}(S_1) \cap S_B^{\mathsf{m}}(G)) - L_{\mathsf{m}}(G, G^*|S_1) \\ \geq & (1-\delta) \cdot \frac{P(\mathcal{N}(S_1) \cap S_B^{\mathsf{m}}(G))}{P(S_1)} \cdot L_{\mathsf{m}}(G, G^*|S_1) - L_{\mathsf{m}}(G, G^*|S_1) \\ \geq & (1-\delta) \cdot \frac{P(\mathcal{N}(S_1)) - P(\overline{S_B^{\mathsf{m}}(G)})}{P(S_1)} \cdot L_{\mathsf{m}}(G, G^*|S_1) - L_{\mathsf{m}}(G, G^*|S_1) \\ \geq & (1-\delta) \cdot \frac{P(\mathcal{N}(S_1))}{P(S_1)} \cdot L_{\mathsf{m}}(G, G^*|S_1) - \frac{L_{\mathsf{m}}(G, G^*|S_1)}{P(S_1)} \cdot P(\overline{S_B^{\mathsf{m}}(G)}) - L_{\mathsf{m}}(G, G^*|S_1) \\ \geq & (1-\delta)c \cdot L(G, G^*|S_1) - P(\overline{S_B^{\mathsf{m}}(G)}) - L_{\mathsf{m}}(G, G^*|S_1) \\ = & (c(1-\delta) - 1)L_{\mathsf{m}}(G, G^*|S_1) - R_B^{\mathsf{m}}(G) \end{aligned}$$

Now, substituting $L(G, G^*|S_2)$ on the LHS with Lemma C.2 and noticing $\mathcal{M}(G_{pl}) \backslash \overline{S_B^{\mathsf{m}}(G)} \subseteq S_1$, with simple transformation we have:

$$\begin{aligned} (c(1-\delta) - 2)L(G, G^*|S_1) \leq & L_{\mathsf{m}}(G, G_{pl}) - \text{Err}(G_{pl}) + 2 * R_B^{\mathsf{m}}(G) \\ \leq & \alpha \end{aligned}$$

The last inequality comes from the condition in the lemma. Thus, with Fact C.3, we know $L(G, G^*|S_1) \leq \alpha/((c(1-\delta) - 2)) \leq q$. Now, we can bound the overall error:

$$\begin{aligned} \text{Err}_{\mathsf{m}}(G) = & L_{\mathsf{m}}(G, G^*|S_1) + L_{\mathsf{m}}(G, G^*|S_2) + L_{\mathsf{m}}(G, G^*|\overline{S_B^{\mathsf{m}}(G)}) \\ \leq & q + (q + L_{\mathsf{m}}(G, G_{pl}) - \text{Err}(G_{pl}) + R_B^{\mathsf{m}}(G)) + R_B^{\mathsf{m}}(G) \\ \leq & 2(q + R_B^{\mathsf{m}}(G)) + L_{\mathsf{m}}(G, G_{pl}) - \text{Err}(G_{pl}) \end{aligned}$$

□

Now, we will prove our main lemma, based on which we will be able to derive Theorem 5.1 and Theorem 5.2.

Lemma C.5. *Suppose Assumption 5.2 holds true. Then we can bound:*

$$\mathit{Err}_{\mathsf{m}}(G) \leq L(G) \triangleq \frac{c+3}{c-1} L_m(G, G_{pl}) + \frac{2c+2}{c-1} R_{\mathcal{B}}^m(G; \delta) - \mathit{Err}(G_{pl}).$$

for any $\delta \in [0, \frac{1}{c}]$.

Proof. First, we consider the case where $L_{\mathsf{m}}(G, G_{pl}) + 2R_B^{\mathsf{m}}(G) \leq \frac{c-1}{2} \cdot \text{Err}(G_{pl})$. In this case, we can find some $\beta \in (0, \frac{c-1}{2}]$ such that.

$$L_{\mathsf{m}}(G, G_{pl}) + 2R_B^{\mathsf{m}}(G) = \beta \text{Err}(G_{pl}) = \text{Err}(G_{pl}) + (\beta - 1)\text{Err}(G_{pl})$$

Thus, by lemma C.4, we have:

$$\begin{aligned} \text{Err}_{\mathsf{m}}(G) \leq & 2(\frac{\beta}{c-1}\text{Err}(G_{pl}) + R_B^{\mathsf{m}}(G)) + L_{\mathsf{m}}(G, G_{pl}) - \text{Err}(G_{pl}) \\ = & \frac{2}{c-1}\beta\text{Err}(G_{pl}) + 2R_B^{\mathsf{m}}(G) + L_{\mathsf{m}}(G, G_{pl}) - \text{Err}(G_{pl}) \\ = & \frac{2}{c-1}(L_{\mathsf{m}}(G, G_{pl}) + 2R_B^{\mathsf{m}}(G)) + 2R_B^{\mathsf{m}}(G) + L_{\mathsf{m}}(G, G_{pl}) - \text{Err}(G_{pl}) \\ \leq & \frac{c+3}{c-1} L_{\mathsf{m}}(G, G_{pl}) + \frac{2c+2}{c-1} R_B^{\mathsf{m}}(G) - \text{Err}(G_{pl}) \\ = & L(G) \end{aligned}$$

Next, we consider the case where $L_{\mathsf{m}}(G, G_{pl}) + 2R_B^{\mathsf{m}}(G) > \frac{c-1}{2} \cdot \text{Err}(G_{pl})$. By triangle inequality, we have:

$$\begin{aligned}
\text{Err}_{\mathsf{m}}(G) = L_{\mathsf{m}}(G, G^*) &\leq L_{\mathsf{m}}(G, G_{pl}) + L_{\mathsf{m}}(G_{pl}, G^*) \\
&= L_{\mathsf{m}}(G, G_{pl}) + 2\text{Err}(G_{pl}) - \text{Err}(G_{pl}) \\
&< L_{\mathsf{m}}(G, G_{pl}) + \frac{4}{c-1}(L_{\mathsf{m}}(G, G_{pl}) + 2R_B^{\mathsf{m}}(G)) - \text{Err}(G_{pl}) \\
&= \frac{c+3}{c-1} L_{\mathsf{m}}(G, G_{pl}) + \frac{8}{c-1} R_B^{\mathsf{m}}(G) - \text{Err}(G_{pl}) \\
&\leq \frac{c+3}{c-1} L_{\mathsf{m}}(G, G_{pl}) + \frac{2c+2}{c-1} R_B^{\mathsf{m}}(G) - \text{Err}(G_{pl}) \qquad \text{(using } c > 3\text{)} \\
&= L(G)
\end{aligned}$$

□

Now, we provide the proof of Theorem 5.1 and Theorem 5.2 based on Lemma C.5.

Proof of Theorem 5.1. Since $\hat{G}$ is an optimizer of $L(G)$, we know $\text{Err}_{\mathsf{m}}(\hat{G}) \leq L(\hat{G}) \leq L(G^*)$. Substituting G^* into $L(G)$ gives the bound in the theorem. □

Proof of Theorem 5.2. Note that $\text{CertR}(G) \geq \frac{0.5 - Err(G)}{Lip(G)}$ by its definition. Substituting $Err(G) \leq L(G)$ gives the inequality in the theorem. □

D Error Control of Newton's Method

Recall that given an unconstrained matrix $V \in \mathbb{R}^{n \times n}$, we know that $W = (VV^\intercal)^{-\frac{1}{2}} V$ is orthogonal, i.e., $||W||_2 = 1$, which provides certified robustness for the resulting model. In practice, we use a finite number of Newton's iteration steps to approximate $(VV^\intercal)^{-\frac{1}{2}}$. In this section, we provide the following theorem which rigorously control the spectral norm under finite Newton's iteration steps.

Theorem D.1. *Given a matrix $V \in \mathbb{R}^{n \times n}$ such that $||I - VV^\intercal||_2 < 1$, if we use Newton's iteraton for $k^\star$ steps following Equation* (1) *with initialization $Y_0 = VV^\intercal$ and $Z_0 = I$, then we have*

$$||Z_{k^\star} V||_2 \leq 1 + \frac{||V||_2}{\sqrt{\rho_{\min}(VV^\intercal)}}(1 - \sqrt{1 - ||I - VV^\intercal||_2^{2^{k^\star}}}) \leq 1 + \frac{||V||_2}{\sqrt{\rho_{\min}(VV^\intercal)}} ||I - VV^\intercal||_2^{2^{k^\star}}, \tag{2}$$

where $\rho_{\min}$ is the smallest eigenvalue of the matrix.

Remark. We make sure the condition $||I - VV^\intercal||_2 < 1$ is satisfied by rescaling as discussed in Section 4.1. As the theorem shows, along with the increase of Newton's iteration step $k^\star$, the spectral norm of $Z_{k^\star} V$ approaches 1 where the additional term $||I - VV^\intercal||_2^{2^k}$ decays exponentially. Hence, we can rigorously bound the error in the orthogonalization process caused by finite steps and use the bound to determine how many finite steps are needed. Indeed, in practice, we apply singular value decomposition to the computed matrix $Z_{k^\star} V$, and find its maximum singular value always approaches 1 from the left side, i.e., the actual spectral norm is equal to or smaller than 1. Detail experimental verification is in Appendix E.4.

D.1 Proof of Theorem D.1

For brevity, throughout this section, for $V \in \mathbb{R}^{n \times n}$, we define $A = VV^\intercal$. Note that A is a real symmetric matrix. Then, we recursively define

$$\begin{aligned}
B_0 &= I, \\
B_{k+1} &= \frac{1}{2}\left(3B_k - B_k^3 A\right).
\end{aligned} \tag{3}$$

Before proving the main theorem, we first present the following three lemmas.

Lemma D.1. *For any $k \in \mathbb{N}$, $B_k A = A B_k$ and $B_k A^{\frac{1}{2}} = A^{\frac{1}{2}} B_k$.*

Proof of Lemma D.1. We prove the lemma by induction. Since $B_0 = I$, for $k = 0$ the lemma holds. Suppose that the lemma holds for k, then we have

$$B_{k+1}A = \frac{1}{2}(3B_k - B_k^3 A)A = \frac{1}{2}(3B_k A - B_k^3 A^2) \overset{(*)}{=} \frac{1}{2}(3AB_k - AB_k^3 A) = A \cdot \frac{1}{2}(3B_k - B_k^3 A) = AB_{k+1},$$
$$B_{k+1}A^{\frac{1}{2}} = \frac{1}{2}(3B_k - B_k^3 A)A^{\frac{1}{2}} = \frac{1}{2}(3B_k A^{\frac{1}{2}} - B_k^3 A^{\frac{3}{2}}) \overset{(*)}{=} \frac{1}{2}(3A^{\frac{1}{2}} B_k - A^{\frac{1}{2}} B_k^3 A) = A^{\frac{1}{2}} \cdot \frac{1}{2}(3B_k - B_k^3 A) = A^{\frac{1}{2}} B_{k+1}. \tag{4}$$

In above equations, $(*)$ is due to the induction assumption that $B_k A = AB_k$ or $B_k A^{\frac{1}{2}} = A^{\frac{1}{2}} B_k$. Therefore, the lemma holds with $k + 1$ and by induction the lemma holds for any $k \in \mathbb{N}$. □

Lemma D.2. *For any $k \in \mathbb{N}$, $Y_k = B_k A$ and $Z_k = B_k$.*

Proof of Lemma D.2. We prove the lemma by induction. Since $Y_0 = VV^\intercal = A$, $Z_0 = I$, and $B_0 = I$, for $k = 0$ the lemma holds. Suppose that the lemma holds for k, then we have

$$\begin{aligned} Y_{k+1} &= \frac{1}{2} Y_k (3I - Z_k Y_k) = \frac{1}{2} B_k A (3I - B_k^2 A) \overset{(*)}{=} \frac{1}{2}(3B_k - B_k^3 A) \cdot A = B_{k+1} A, \\ Z_{k+1} &= \frac{1}{2}(3I - Z_k Y_k) Z_k = \frac{1}{2}(3I - B_k^2 A) B_k \overset{(*)}{=} \frac{1}{2}(3B_k - B_k^3 A) = B_{k+1}, \end{aligned} \tag{5}$$

where $(*)$ leverages $B_k A = AB_k$ from Lemma D.1. Therefore, by induction the lemma holds for any $k \in \mathbb{N}$. □

Lemma D.3. *When $||I - A||_2 < 1$, for any $k \in \mathbb{N}$, the eigenvalue $\lambda \in \mathbb{R}$ of matrix $A^{\frac{1}{2}} B_k$ is positive.*

Proof of Lemma D.3. We define $C_k = A^{\frac{1}{2}} B_k$, then by leveraging the commutability between B_k and $A/A^{\frac{1}{2}}$ (Lemma D.1) we have the following iteration:

$$\begin{cases} C_0 = A^{\frac{1}{2}}, \\ C_{k+1} = \frac{1}{2}(3C_k - C_k^3). \end{cases} \tag{6}$$

Since $||I - A||_2 < 1$ and A is a real symmetric matrix, any eigenvalue of C_0, denoted by $\lambda_i^{C_0}$, $\in (0, \sqrt{2})$. Denoting the diagonalization of C_0 by $C_0 = P^{-1} D_0 P$ where $D_0 = \mathrm{diag}(\lambda_1^{C_0}, \cdots, \lambda_n^{C_0})$. Then, from the iteration, we have $C_{k+1} = P^{-1}\left(\frac{1}{2}(3D_k - D_k^3)\right) P$ and therefore $\lambda_i^{C_{k+1}} = \lambda_i^{C_k}\left(\frac{3}{2} - \frac{1}{2}(\lambda_i^{C_k})^2\right)$. Define function $f : \mathbb{R} \to \mathbb{R}$ such that $f(x) = x(\frac{3}{2} - \frac{1}{2}x^2)$. We find $f'(x) = 0 \Rightarrow x \in \{0, 1\}$. Thus, when $x \in (0, \sqrt{2})$, $f(x) \in (0, 1) \subseteq (0, \sqrt{2})$. Now we apply the induction. When $k = 0$, we have $\lambda_i^{C_k} \in (0, \sqrt{2})$. Suppose $\lambda_i^{C_k} \in (0, \sqrt{2})$, from above result we have $\lambda_i^{C_{k+1}} = f(\lambda_i^{C_k}) \in (0, \sqrt{2})$. Thus, for any $k \in \mathbb{N}$, all eigenvalues of $C_k = A^{\frac{1}{2}} B_k$ are positive. □

Now we are ready to prove the main theorem.

Proof of Theorem D.1. Since $Z_k = B_k$ for any $k \in \mathbb{N}$ acccording to Lemma D.2, we focus on B_k and its expression of iteration (Equation (3)) henceforth. According to [3, Section 6], define $R_k = I - B_k^2 A$, we have $R_{k+1} = \frac{3}{4}R_k^2 + \frac{1}{4}R_k^3$. We have $||I - A||_2 = ||R_0||_2 < 1$. By induction,

$$\begin{aligned} ||R_{k+1}||_2 &\le \frac{3}{4}||R_k||_2^2 + \frac{1}{4}||R_k||_2^3 \\ &\le \frac{3}{4}||R_k||_2^2 + \frac{1}{4}||R_k||_2^2 \quad \text{(by induction condition } ||R_k||_2 \le 1) \\ &= ||R_k||_2^2. \end{aligned} \tag{7}$$

Therefore, $||R_{k^\star}||_2 \le ||I - A||_2^{2^{k^\star}}$, i.e., the eigenvalues of the symmetric matrix $I - B_{k^\star}^2 A$ are in the range $[-||I - A||_2^{2^{k^\star}}, ||I - A||_2^{2^{k^\star}}]$.

Given an eigenvalue $\lambda \in \mathbb{R}$ with eigenvector $x \in \mathbb{R}^n$ of the real symmetric matrix $A^{\frac{1}{2}} B_{k^\star}$, we have

$$\begin{aligned} & A^{\frac{1}{2}} B_{k^\star} x = \lambda x \\ \iff & B_{k^\star}^2 A x = \lambda^2 x \qquad \text{(by Lemma D.1)} \\ \iff & (I - B_{k^\star}^2 A) x = \left(1 - \lambda^2\right) x. \end{aligned} \tag{8}$$

Thus, $(1-\lambda^2)$ is an eigenvalue of matrix $(I - B_{k^\star}^2 A)$. Since $(1-\lambda^2) \in [-||I-A||_2^{2^{k^\star}}, ||I-A||_2^{2^{k^\star}}]$, we get

$$\min\{|\lambda - 1|, |\lambda + 1|\} \leq 1 - \sqrt{1 - ||I - A||_2^{2^{k^\star}}}. \tag{9}$$

Then, according to Lemma D.3, we know that $\lambda > 0$ and hence $\lambda \in \left[\sqrt{1 - ||I-A||_2^{2^{k^\star}}}, 2 - \sqrt{1 - ||I-A||_2^{2^{k^\star}}}\right]$. Now we apply diagonalization to $A^{\frac{1}{2}} B_{k^\star}$:

$$A^{\frac{1}{2}} B_{k^\star} := P^\intercal \Lambda P. \tag{10}$$

As a result,

$$\begin{aligned} & ||B_{k^\star} V||_2 \\ =& ||A^{-\frac{1}{2}} P^\intercal \Lambda P V||_2 \\ =& ||A^{-\frac{1}{2}} P^\intercal (\Lambda - I) P V + A^{-\frac{1}{2}} P^\intercal I P V||_2 \\ \leq& ||A^{-\frac{1}{2}}||_2 \cdot (1 - \sqrt{1 - ||I-A||_2^{2^{k^\star}}}) \cdot ||V||_2 + ||A^{-\frac{1}{2}} V||_2 \\ =& ||A^{-\frac{1}{2}}||_2 \cdot ||V||_2 \cdot (1 - \sqrt{1 - ||I-A||_2^{2^{k^\star}}}) + 1 \end{aligned} \tag{11}$$

where the last equality uses the fact $A^{-\frac{1}{2}} V$ is a orthogonal matrix with spectral norm 1.

Since any eigenvector with eigenvalue λ of $A^{-\frac{1}{2}}$ corresponds to the eigenvector with eigenvalue $(1/\lambda^2)$ of $A = VV^\intercal$, and VV^T as a symmetric real matrix only has real eigenvalues,

$$||A^{-\frac{1}{2}}||_2 = \frac{1}{\sqrt{\rho_{\min}(VV^\intercal)}}. \tag{12}$$

Plug it into Equation (11), we get

$$||Z_{k^\star} V||_2 = ||B_{k^\star} V||_2 = 1 + \frac{||V||_2}{\sqrt{\rho_{\min}(VV^\intercal)}} (1 - \sqrt{1 - ||I-A||_2^{2^{k^\star}}}). \tag{13}$$

Notice that $1 - \sqrt{1-x} \leq x$ for $x = ||I-A||_2^{2^{k^\star}} \in [0, 1]$, we conclude the proof.

□

E Addition Exp Results

E.1 Visualization

We show the representation visualization on 16 neurons for SOC and LOT in Figure 3.

E.2 Residual Connection

We show the loss landscape for both LOT and SOC in Figure 4

E.3 Circular vs. Zero padding

As we discussed in Section 4.1, the default parametrization leads to the convolution result with circular-padding, while we will pre-process the input to get the final result with zero-padding. In Table 5, we show the performance comparison between circular and zero padding. We can see that the performance significantly improves when we do the pre-processing and use the zero-padding instead of default circular-padding.

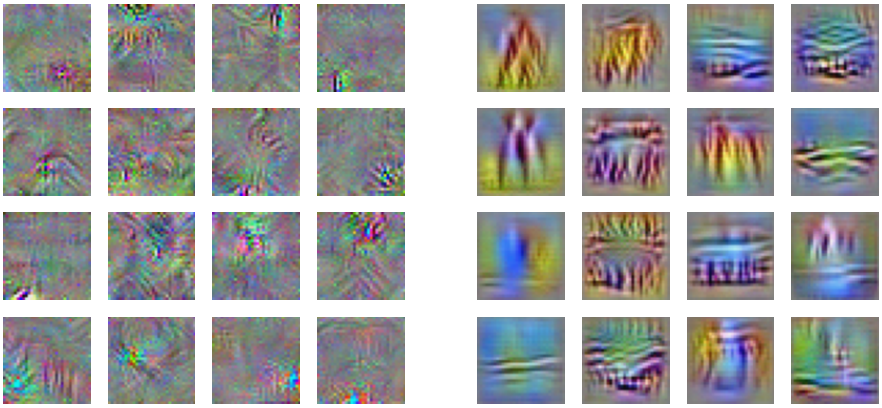
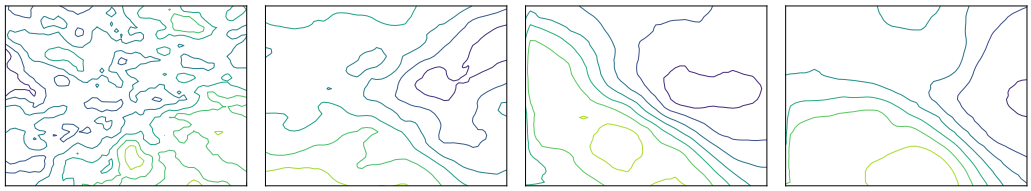


Figure 3: Visualizing the features in the last hidden layer of LipConvnet-20 for SOC (left) and LOT (right). Each image corresponds to one randomly chosen neuron from the last hidden layer and is optimized to maximize the value of the neuron.



(a) LOT, without residual connection. (b) LOT, with residual connection. (c) SOC, without residual connection. (d) SOC, with residual connection.

Figure 4: The loss landscape [12] with respect to the parameters of LOT and SOC network with and without residual connections. The figure is plotted by calculating the loss contour on two randomly chosen directions of parameters. We can observe that the residual connection greatly smoothifies the LOT network, while both SOCs with and without residual connection are smooth.

E.4 Error Control of Newton's Iterations

As we discuss in Section D, we use a finite number of Newton's iteration to approximate the inverse square root of a matrix $Z_{k^*} \approx (VV^\intercal)^{-\frac{1}{2}}$. Theoretically, we have shown that the error of approximated orthogonal matrix $Z_{k^*}V$ will decay exponentially with number of iterations. Furthermore, we observe in practice that during the Newton's iteration, the maximum singular value of Z_kV will always approach 1 from the left side. As an example, we show the maximum singular value (σ_{max}) for all LOT layers in LipConvnet-20 at different steps (k) during the Newton's Iteration in Figure 5. We can see that $\sigma_{max} < 1$ for all the layers, so we can safely assume that the Lipschitz bound is no larger than 1. In addition, $\sigma_{max} > 1 - 10^{-4} = 0.9999$ after $k = 8$ iterations for all the layers, which indicates that the Newton's iteration converges well.

E.5 Supervised Learning without CReg Loss and HH Activation

We show the results of semi-supervised learning under standard setting (without CReg Loss, HH Activation, and LLN on CIFAR-100) in Table 6 and Table 7. We can observe that we still achieve a good performance compared with SOC, and the gap is sometimes larger than with the different optimization techniques.

E.6 Full semi-supervised

We show the results for all architectures with semi-supervised learning in Table 8 (with CReg and HH) and 9 (without CReg and HH). We can see observe that we still achieve a better performance

Table 5: Performance comparison of LOT network with default circular padding and pre-processed zero padding.

Model	Conv. Type	Padding	Vanilla Accuracy	Certified Accuracy at $\rho =$ 36/255	72/255	108/255
LipConvnet-20	LOT	Circular-pad	76.65%	61.15%	43.83%	28.78%
		Zero-pad	**77.86%**	**63.54%**	**47.15%**	**32.12%**

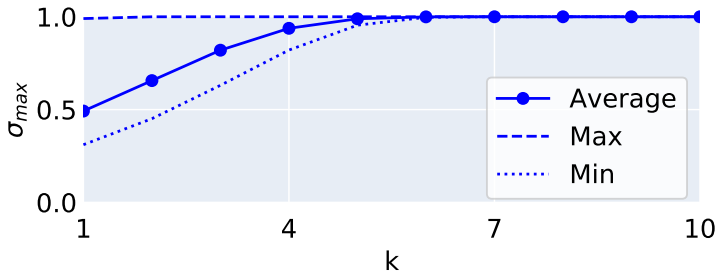


Figure 5: The maximum singular value σ_{max} of $Z_k V$ for each of the LOT layer in LipConvnet-20 during the Newton's iteration. We verify that all the σ_{max}'s are smaller than 1, and after the 8-th iteration, all the σ_{max}'s are larger than 0.9999.

compared with SOC. In addition, these results also have a larger certified accuracy at larger radius compared with the supervised learning setting.

F Results on TinyImageNet

To further validate the performance of LOT, we evaluate it on the TinyImageNet dataset which consists of 100,000 64×64 images in 200 classes. We use the same model architecture and training setting as before and show the comparison between LOT and SOC in Table 10. We can observe that LOT still outperforms the existing SOC approach in most cases. Meanwhile, we observe that all 1-Lipschitz models have a performance drop on the larger dataset compared with vanilla models, and we leave the breakthrough as future work.

G Discussion of Model Architectures

We would like to emphasize that there are two major differences in designing 1-Lipschitz networks and standard networks. First, standard architectures focus a lot on regularizing the network so that it does not overfit (e.g. with dropout), while in 1-Lipschitz networks we do not need strong regularization because the 1-Lipschitz constraint is already strong enough. Second, standard architectures are carefully designed so that the gradient will propagate in a proper manner (e.g. residual connection, BatchNorm), while 1-Lipschitz networks have an intrinsic gradient-norm-preserving property (see [16]). Thus, it may not be appropriate to directly utilize modern architectures in the design of 1-Lipschitz networks. In Table 11, we show the results of a 1-Lipschitz ResNet18 model, where we (1) replace all the conv layers with 1-Lipschitz convolution, (2) replace all the residual connection with average and (3) enforce the variance parameter of BatchNorm layers to be 1 so that the model is 1-Lipschitz. We can observe that LOT still outperforms SOC on both CIFAR-10 and TinyImageNet, while the performance of the ResNet18 architecture is not as good as that of LipConvNet.

Table 6: Certified accuracy of 1-Lipschitz model without CReg loss and HH activation on CIFAR-10 in supervised setting.

Model	**Conv. Type**	**Vanilla Accuracy**	**Certified Accuracy at $\rho =$** 36/255	72/255	108/255	**Evaluation Time (sec)**
LipConvnet-5	SOC	75.78%	59.18%	42.01%	27.09%	2.117
	LOT	**77.20%**	**61.76%**	**44.45%**	**29.61%**	**1.406**
LipConvnet-10	SOC	76.45%	60.86%	44.15%	29.15%	3.170
	LOT	**77.30%**	**62.54%**	**46.03%**	**30.64%**	**1.420**
LipConvnet-15	SOC	76.68%	61.36%	44.28%	29.66%	3.993
	LOT	**77.34%**	**63.40%**	**46.54%**	**31.75%**	**1.453**
LipConvnet-20	SOC	76.90%	61.87%	45.79%	31.08%	4.752
	LOT	**77.86%**	**63.54%**	**47.15%**	**32.12%**	**1.558**
LipConvnet-25	SOC	75.24%	60.17%	43.55%	28.60%	5.613
	LOT	**77.76%**	**62.77%**	**46.06%**	**31.20%**	**1.834**
LipConvnet-30	SOC	74.51%	59.06%	42.46%	28.05%	6.438
	LOT	**77.34%**	**62.76%**	**46.24%**	**31.07%**	**2.219**
LipConvnet-35	SOC	73.73%	58.50%	41.75%	27.20%	7.400
	LOT	**77.54%**	**62.62%**	**46.28%**	**31.64%**	**2.620**
LipConvnet-40	SOC	71.63%	54.39%	37.92%	24.13%	8.175
	LOT	**77.79%**	**62.69%**	**46.34%**	**31.32%**	**2.910**

Table 7: Certified accuracy of 1-Lipschitz model without CReg loss, HH activation and LLN on CIFAR-100 in supervised setting.

Model	**Conv. Type**	**Vanilla Accuracy**	**Certified Accuracy at $\rho =$** 36/255	72/255	108/255
LipConvnet-5	SOC	42.71%	27.86%	17.45%	9.99%
	LOT	**46.07%**	**31.28%**	**19.86%**	**12.17%**
LipConvnet-10	SOC	43.72%	29.39%	18.56%	11.16%
	LOT	**44.68%**	**30.59%**	**19.69%**	**12.33%**
LipConvnet-15	SOC	42.92%	28.81%	17.93%	10.73%
	LOT	**46.01%**	**32.08%**	**20.72%**	**12.92%**
LipConvnet-20	SOC	43.06%	29.34%	18.66%	11.20%
	LOT	**46.05%**	**32.17%**	**20.81%**	**13.16%**
LipConvnet-25	SOC	43.37%	28.59%	18.18%	10.85%
	LOT	**46.21%**	**31.81%**	**21.01%**	**12.83%**
LipConvnet-30	SOC	42.87%	28.74%	18.47%	11.21%
	LOT	**45.71%**	**32.23%**	**20.87%**	**13.03%**
LipConvnet-35	SOC	42.42%	28.34%	18.10%	10.96%
	LOT	**45.38%**	**31.03%**	**20.02%**	**12.46%**
LipConvnet-40	SOC	41.84%	28.00%	17.40%	10.28%
	LOT	**45.30%**	**30.91%**	**19.97%**	**12.36%**

Table 8: Certified accuracy of 1-Lipschitz model with CReg loss and HH activation on CIFAR-10 in semi-supervised setting.

Model	**Conv. Type**	**Vanilla Accuracy**	**Certified Accuracy at $\rho =$** 36/255	72/255	108/255
LipConvnet-5 + CReg + HH	SOC	69.67%	59.28%	48.02%	38.30%
	LOT	**71.52%**	**61.25%**	**50.66%**	**40.31%**
LipConvnet-10 + CReg + HH	SOC	71.10%	60.81%	50.61%	41.03%
	LOT	**71.82%**	**62.60%**	**51.76%**	**42.31%**
LipConvnet-15 + CReg + HH	SOC	71.15%	61.65%	51.78%	42.53%
	LOT	**71.89%**	**62.37%**	**52.24%**	**42.71%**
LipConvnet-20 + CReg + HH	SOC	70.95%	61.72%	51.78%	42.01%
	LOT	**71.86%**	**62.86%**	**52.24%**	**42.39%**
LipConvnet-25 + CReg + HH	SOC	70.58%	61.40%	51.46%	42.05%
	LOT	**71.64%**	**62.67%**	**52.00%**	**42.34%**
LipConvnet-30 + CReg + HH	SOC	70.37%	60.74%	51.23%	41.81%
	LOT	**72.08%**	**62.84%**	**51.99%**	**41.94%**

Table 9: Certified accuracy of 1-Lipschitz model without CReg loss and HH activation on CIFAR-10 in semi-supervised setting.

Model	**Conv. Type**	**Vanilla Accuracy**	**Certified Accuracy at $\rho =$** 36/255	72/255	108/255
LipConvnet-5	SOC	70.67%	59.14%	45.88%	34.21%
	LOT	**72.86%**	**61.66%**	**48.84%**	**36.67%**
LipConvnet-10	SOC	71.86%	60.15%	48.01%	35.66%
	LOT	**73.57%**	**62.62%**	**50.26%**	**37.82%**
LipConvnet-15	SOC	73.05%	62.11%	50.44%	38.50%
	LOT	**73.74%**	**63.01%**	**50.66%**	**38.73%**
LipConvnet-20	SOC	72.77%	62.51%	50.40%	38.05%
	LOT	**73.64%**	**63.37%**	**51.00%**	**38.52%**
LipConvnet-25	SOC	72.45%	62.03%	50.12%	38.28%
	LOT	**73.62%**	**63.61%**	**51.21%**	**38.77%**
LipConvnet-30	SOC	71.04%	61.06%	49.30%	38.11%
	LOT	**71.71%**	**63.32%**	**51.27%**	**39.42%**

Table 10: Certified accuracy of 1-Lipschitz model without CReg loss and HH activation on TinyImageNet in supervised setting.

Model	**Conv. Type**	**Vanilla Accuracy**	**Certified Accuracy at $\rho =$** 36/255	72/255	108/255
LipConvnet-5	SOC	30.77%	19.74%	11.60%	6.89%
	LOT	**32.71%**	**21.44%**	**12.96%**	**7.92%**
LipConvnet-10	SOC	31.94%	21.21%	12.80%	**7.79%**
	LOT	**32.31%**	**21.22%**	**12.96%**	7.75%
LipConvnet-15	SOC	32.26%	21.36%	12.94%	7.80%
	LOT	**33.14%**	**22.21%**	**13.34%**	**8.12%**
LipConvnet-20	SOC	32.44%	21.27%	12.90%	7.63%
	LOT	**33.19%**	**22.02%**	**13.42%**	**8.12%**

Table 11: Certified accuracy of 1-Lipschitz ResNet-18 model without CReg loss and HH activation on CIFAR-10/TinyImageNet in supervised setting

Model	**Conv. Type**	CIFAR-10				TinyImageNet			
		Vanilla Accuracy	**Certified Accuracy at $\rho =$** 36/255	72/255	108/255	**Vanilla Accuracy**	**Certified Accuracy at $\rho =$** 36/255	72/255	108/255
ResNet-18	SOC	66.43%	43.00%	23.00%	9.52%	23.26%	11.64%	5.21%	2.17%
	LOT	**68.85%**	**45.46%**	**25.43%**	**11.35%**	**25.09%**	**12.83%**	**5.90%**	**2.62%**