特徴マップの幾何変換前後に着目した敵対サンプルの検出

土松 千紗[1,a)]　足立 浩規[1,b)]　平川 翼[1,c)]　山下 隆義[1,d)]　藤吉 弘亘[1,e)]

概要

AEs を検出する手法が多数提案されているが，その多くは AEs がネットワーク内部にどのような影響を及ぼしているか十分に調査されていない．そこで本研究では，AEs を入力した際のネットワークの内部状態を把握するために，任意の幾何変換を施した画像の特徴マップを分析する．また，分析結果に基づいた AEs の検出法を提案し，従来手法と比較する．評価実験により，提案手法が従来の防御手法より AEs の認識率を向上させることを示す．

1. はじめに

コンピュータビジョンの分野で，畳み込みニューラルネットワーク (CNN) は優れた性能を発揮している一方で，入力画像に悪意のある微小摂動を付与した画像である Adversarial Examples (AEs) [1] に脆弱である．AEs の摂動は人が認知不可能であるにも関わらず，CNN は高い信頼度で誤認識するため，セキュリティの観点で問題視されている．

AEs に対する防御手法として, 推論モデルの前段に検出器を設けて，AEs か否かの判定を下す Adversarial detection [2], [3], [4] が提案されている．これらの手法はあらゆる推論モデルへの導入が容易である一方，多くの手法が通常のサンプルと，AEs の出力関係のみ重視しており，特徴マップや重みなどのモデルの内部状態に対する分析が不十分である．以降，記載がない限り通常のサンプルを Clean と呼称する．AEs の多くがモデルの勾配をもとに摂動を求めるため，モデル内部の変化を考慮することでさらなる性能向上が期待できる．

そこで本研究では，入力画像に様々な幾何変換を施した時のモデルの内部状態である特徴マップに着目して分析を行い，Clean と AEs の関係を明らかにする．そして，分析によって得られた Clean と AEs の傾向をもとにした，新たな AEs の検出手法を提案する．評価実験では，提案手法と多くの従来手法で用いられている事後確率による AEs の検出を組み合わせた検出器の性能を従来手法と比較する．

[1] 中部大学
[a)] matsuno@mprg.cs.chubu.ac.jp
[b)] ha618@mprg.cs.chubu.ac.jp
[c)] hirakawa@mprg.cs.chubu.ac.jp
[d)] takayoshi@isc.chubu.ac.jp
[e)] fujiyoshi@isc.chubu.ac.jp

2. 関連研究

2.1 Adversarial Examples

AEs は，データを $\boldsymbol{x}$，データに対する教師信号を y，摂動の許容範囲を ϵ としたとき，式 (1) を満たすように求めた摂動を付与した画像を表している．

$$\boldsymbol{\delta} = \max_{\|\boldsymbol{\delta}\|_\infty \leq \epsilon} L\left(\boldsymbol{x} + \boldsymbol{\delta}, y; \boldsymbol{\theta}\right) \tag{1}$$

この最適化によって適切な摂動を求めることができるが，途方もない時間が必要となる．そこで，効率よく摂動を求める手法として Projected Gradient Descent (PGD) [5] が提案されている．PGD は損失を入力画像に関して微分したときに得る勾配方向を利用して，各画素を徐々に微小な変化を適用することで強い摂動を求める．L_∞-norm を用いた PGD は，式 (2) で定義することができる．

$$\boldsymbol{x}_{t+1} := P(\boldsymbol{x}_t + \alpha \cdot \mathrm{sign}(\nabla_{\boldsymbol{x}_t} L(\boldsymbol{x}_t, y; \boldsymbol{\theta}))) \tag{2}$$

ここで，$\boldsymbol{\theta}$ は重みパラメータ，$L(\cdot)$ は損失関数，α はステップごとに加えるノイズの大きさである．また，$P(\cdot)$ は対象が領域外となった場合，領域内に投影する．

2.2 Adversarial detection

Adversarial detection とは，AEs に対する検出器を推論モデルの前に設けることで AEs の入力を防ぐ方法である．

PixelDefend[2] は，画像生成モデルの 1 つである PixelCNN [6] を用いて画素間の依存関係に注目し，並べ替え検定を用いて AEs を検出する．並べ替え検定は，入力画像と学習データの 2 群の要素を入れ替えた場合の PixelCNN の同時確率分布の差を調査する．

Feature Squeezing[3] は，入力画像に対して色深度の削減や平滑化の適用前後の予測クラスが異なるときに AEs として検出する．AEs と判定されたサンプルは，平滑化処理などによって摂動の影響を抑えてから推論モデルで予測する．

Dense-Layer-Analysis (DLA) [4] は，全結合層のニューロンカバレッジに注目し，推論モデルのニューロンカバ

レッジを入力とする 2 値分類モデルを用いて AEs を検出する．ニューロンカバレッジは，全結合層において活性化関数の出力が閾値以上となったニューロンの割合である．PixelDefend，Feature Squeezing は検出した AEs を，正しいクラスに識別するよう処理を行い，推論用モデルに入力するのに対し DLA は，AEs の検出のみを行う．

AEs の摂動は正解クラス確率が低下するように入力画像に対する勾配から求めるため，CNN の内部状態を考慮することは様々な AEs を高い性能で防御することに繋がると考える．しかし，Feature Squeezing や PixelDefend は入出力の関係のみに着目しているため，内部状態に対する分析が不十分である．また DLA は，AEs を検出時に入力画像の補正はせず，アラームを鳴らす仕様であるため，入力画像を即座に確認できる場面でのみ使用可能であり，用途が限られている．

そこで本研究では，Clean と AEs に様々な幾何変換を施して畳み込み処理適用後の特徴マップに着目した AEs の検出手法を提案する．

3. CNN の挙動分析

本分析では，Clean と AEs に様々な幾何変換を施して畳み込み処理適用後の特徴マップを比較することで AEs が特徴マップに与える影響を調査する．特徴マップはバッチ正規化後のものと，バッチ正規化後に活性化関数を適用したものを比較する．しかし AEs による影響は小さく，分かりづらいため特徴マップの統計により差を明確にする必要がある．本分析では単純な統計である平均と分散を算出することで扱う値の数を減らす．本分析で使用する統計値は，式 (3) に示すような特徴マップ全体の平均値と，式 (4) に示すようなチャネル方向の平均値を使用する．

$$\text{mean} = \frac{\sum_{k=1}^{C}\sum_{j=1}^{H}\sum_{i=1}^{W} n_{kij}}{W \times H \times C} \tag{3}$$

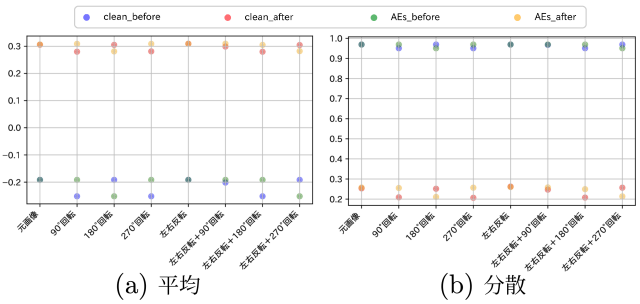
$$\text{mean} = \frac{\sum_{k=1}^{C} n_{kij}}{C} \tag{4}$$

ここで，$n \in \mathbb{R}^{H\times W\times C}$ は特徴マップ，H，W，C は，それぞれ幅，高さ，チャネル数である．また，幾何変換は左右反転と $\{90°, 180°, 270°\}$ の回転を組み合わせた 7 種類を使用する．

3.1 AEs の傾向調査

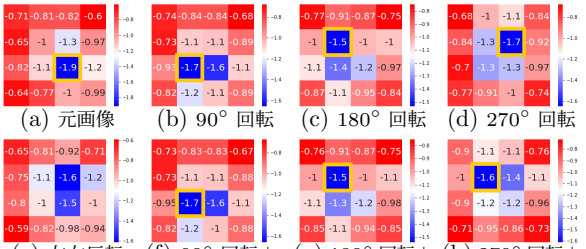
まず 1 サンプルを用いた特徴マップを分析する．データセットは CIFAR-10，モデルは ResNet-18，活性化関数は ReLU，バッチサイズは 256，学習回数は 80 エポックである．初期学習率は 0.1 であり，50 エポックで 1/10 に減衰する．攻撃手法は $\epsilon = 0.031$，$\alpha = 0.003$，反復回数は 20 の PGD を使用する．

7 層目の特徴マップ全体の平均と分散を図 1 に示す．図



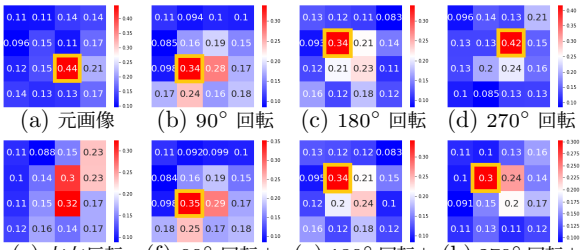
(a) 平均　(b) 分散

図 1: 特徴マップ全体の平均と分散



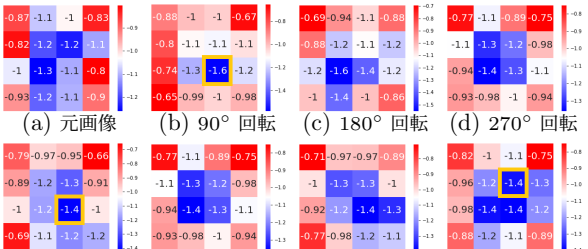
(a) 元画像　(b) 90° 回転　(c) 180° 回転　(d) 270° 回転
(e) 左右反転　(f) 90° 回転＋左右反転　(g) 180° 回転＋左右反転　(h) 270° 回転＋左右反転

図 2: 活性化前の Clean の特徴マップのチャネル方向の平均



(a) 元画像　(b) 90° 回転　(c) 180° 回転　(d) 270° 回転
(e) 左右反転　(f) 90° 回転＋左右反転　(g) 180° 回転＋左右反転　(h) 270° 回転＋左右反転

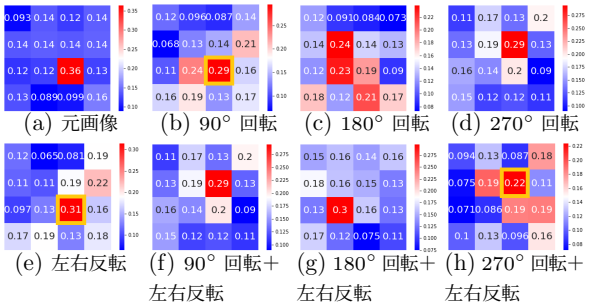
図 3: 活性化後の Clean の特徴マップのチャネル方向の平均



(a) 元画像　(b) 90° 回転　(c) 180° 回転　(d) 270° 回転
(e) 左右反転　(f) 90° 回転＋左右反転　(g) 180° 回転＋左右反転　(h) 270° 回転＋左右反転

図 4: 活性化前の AEs の特徴マップのチャネル方向の平均

1 より平均，分散ともに AEs と Clean の違いが微小であることが確認できる．次に，活性化前の特徴マップをチャネル方向に計算した結果を図 2，図 4，活性化後の結果を図 3，図 5 に示す．この時，図 2，図 3 が Clean，図 4，図 5 が AEs である．図中の黄色の枠は，活性化前の最小値と活性化後の最大値が一致した座標を示している． 図 2，図 3



(a) 元画像 (b) 90° 回転 (c) 180° 回転 (d) 270° 回転

(e) 左右反転 (f) 90° 回転＋左右反転 (g) 180° 回転＋左右反転 (h) 270° 回転＋左右反転

図 5: 活性化後の AEs の特徴マップのチャネル方向の平均

表 1: 検出に利用する特徴マップに対する AUROC

	3 層目	5 層目	7 層目
AUROC	0.4946	**0.5378**	0.4993

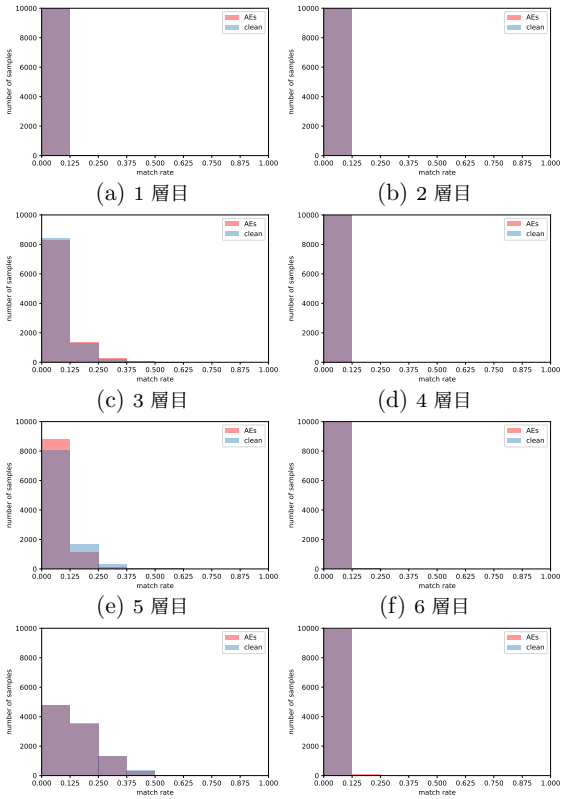
より，活性化後の最大値と活性化前の最小値の座標が，左右反転の場合を除いて一致することが確認できる．このような現象が生じる座標は，多数の負の値と少数の非常に大きな正の値で構成されているため最小値となり，ReLU を適用することで最大値となると予想できる．一方，図 4，図 5 より，活性化後の最大値と活性化前の最小値の座標が 90° 回転，左右反転，270° 回転＋左右反転の場合で一致することが確認できる．これは摂動の影響により，抽出する特徴が Clean から変化し，活性化前の最小値と活性化後の最大値の座標が一致しなくなったと考える．したがって活性化後の最大値と活性化前の最小値の座標の一致率を用いた AEs の検出が期待できる．

次に，複数サンプルの特徴マップを Area Under Receiver Operating Characteristic Curve (AUROC)，各層の特徴マップの座標の一致率をヒストグラムを用いて分析する．AUROC は，横軸を偽陽性率，縦軸を真陽性率とし，プロットしたグラフである ROC 曲線 下の面積であり，値が 1 に近づくほど検出率が高いことを示す．学習条件や PGD は先ほどと同様の設定を使用する．

各層から取得した特徴マップを用いて，座標の一致率を図 6 にヒストグラムとして表現する．図 6 より 3，5，7 層目において，一致率が 0%以外に多く分布しており，閾値の設定が可能である．また 5 層目において，Clean と AEs の分布の違いが最も大きい．次に各学習サンプルに対して座標の一致率を求めた場合の AUROC を表 1 に示す．表 1 より，層の位置に関わらず AUROC は 0.5 に近いが，5 層目が最も高い値であることが確認できる．したがって，5 層目の特徴マップを用いて検出することが適切である．また，図 6(e) より殆どのサンプルが 0 または 0.125 に分布しているため，閾値は 0.125 が適当だと考える．

4. 提案手法

分析結果をもとに，特徴マップと事後確率を利用した AEs に対する検出器を提案する．提案手法による検出の流



(a) 1 層目 (b) 2 層目

(c) 3 層目 (d) 4 層目

(e) 5 層目 (f) 6 層目

(g) 7 層目 (h) 8 層目

図 6: 各層における座標一致率の分布

れをアルゴリズム 1 に示す．ここで閾値は AEs を検出する際に使用する定数である．まず，特徴マップを用いて AEs の判定を行う．5 層目の特徴マップをチャネル方向に平均し，活性化後の最大値と活性化前の最小値の座標が一致する割合を閾値と比較して判定する．そして，事後確率を用いて AEs の判定を行う．各幾何変換画像の事後確率と元画像の事後確率の KL ダイバージェンスの中央値を閾値と比較して判定する．KL ダイバージェンスは式 (5) より求める．

$$KL(P||Q) = \sum_i P_i \log \frac{P_i}{Q_i} \tag{5}$$

また，AEs と判定された場合 2×2 の平均フィルタを用いて平滑化を行い，推論モデルへ入力する．

5. 評価実験

提案手法の有効性を調査するために評価実験を行う．比較対象には PixelDefend，Feature Squeezing，DLA を用いる．また，特徴マップのみ，また事後確率のみを検出に用いた手法とも比較する．

5.1 実験条件

データセットは CIFAR-10，モデルは ResNet-18 である．活性化関数は ReLU，学習回数は 300 エポック，バッチサイズは 256 とする．また，提案手法と Feature Squeezing の初期学習率は 0.1 であり，150 エポック，225 エポック

アルゴリズム 1 検出の流れ

Require: 1 枚の画像 x，学習済みの検出用モデル M，幾何変換 $\mathcal{G} = g_i | i = 1, \ldots, n$，変換回数 n，任意の幾何変換 g_i，事後確率の閾値 τ_1，特徴マップの閾値 τ_2
Ensure: AEs か否かを表す実数

$count \leftarrow 0$ ▷ 座標の一致回数
for $k = 1, 2, \ldots, n$ **do**
  $xp \leftarrow g_k(x)$
  $yp_k, f_{before}, f_{after} \leftarrow \mathrm{M}(xp)$ ▷ 事後確率，活性化前後の特徴マップ
  $f_{before}, f_{after} \leftarrow \mathrm{mean} = \frac{\sum_{k=1}^{C} f_{kij}}{C}$
  if $\arg\min_i f^{(i)}_{before}$ の座標 == $\arg\max_i f^{(i)}_{after}$ の座標 **then**
    $count \leftarrow count + 1$
  end if
end for
$kl_{sum} \leftarrow KL(yp_1 || yp_{k|2,3,\ldots,n}) = \sum_i yp_{1,i} \log \frac{yp_{1,i}}{yp_{k,i}}$
if $\mathrm{median}(kl_{sum}) < \tau_1$ OR $\frac{count}{n} < \tau_2$ **then**
  return 1 ▷ AEs
else
  return 0 ▷ Clean
end if

表 2: 従来手法と提案手法の検出率の F 値の比較

	FGSM	PGD
DLA	**0.815**	**0.833**
Feature Squeezing	0.667	0.667
PixelDefend	0.571	0.571
提案手法 (τ_1=0.1)	0.572	0.581
提案手法 (τ_1=1)	0.666	0.660
提案手法 (τ_1=10)	0.667	0.667
提案手法 (τ_1=15)	0.667	0.667

で 1/10 に減衰する．幾何変換として学習時はデータ拡張のためランダムな矩形切り抜きと左右反転，検出時は一致率確認のため左右反転と $\{90°, 180°, 270°\}$ の回転を組み合わせた 7 種類を適用する．攻撃手法は，Fast Gradient Sign Method (FGSM)[1] ，PGD を使用する．ここで，$\epsilon = 0.031$，$\alpha = 0.003$，摂動計算の反復回数は 20 とする．検出には，畳み込み層から出力された特徴マップを用いる．また，$\tau_2 = \{0.1, 1, 10, 15\}$，$\tau_2 = 0.125$ とする．

5.2 実験結果

提案手法および従来手法を用いた場合の検出率の F 値を表 2 に示す．そして，提案手法および PixelDefend, Feature Squeezing を用いた場合の認識率を表 3 に示す．また，提案手法および特徴マップのみ，また事後確率のみを検出に用いた手法の認識率を表 4 に示す．ここで，提案手法の事後確率を用いた検出における閾値は 0.1 とする． 表 2 より，提案手法は DLA と比較して AEs の検出率の F 値が低く，PixelDefend，Feature Squeezing と同程度である．しかし表 3 より，提案手法の Clean に対する認識率は PixelDefend や Feature Squeezing と同程度で，AEs に対する認識率が高い．よって，AEs に対して提案手法は正しいクラスに識別するように入力画像を修正できていると考える．また，表 4 より，提案手法の Clean に対する認識率は事後確率のみ，または特徴マップのみを用いたときよりも低下した．一方，AEs に対する認識率は向上した．以上より，事後確率と特徴マップによる検出を組み合わせることで，堅牢性を向上させることが可能である．

表 3: 従来手法と提案手法の認識率の比較 [%]

	Clean	FGSM	PGD
PixelDefend	85.00	46.00	46.00
Feature Squeezing	86.50	61.54	2.97
提案手法 (τ_1=0.1)	**88.08**	28.62	1.10
提案手法 (τ_1=1)	86.19	33.44	1.52
提案手法 (τ_1=10)	86.20	**61.93**	72.89
提案手法 (τ_1=15)	86.20	**61.93**	**72.90**

表 4: 検出の組み合わせごとの認識率の比較 [%]

	Clean	FGSM	PGD
防御なし	**92.39**	17.51	0.01
提案手法 (事後確率のみ)	92.30	17.82	0.01
提案手法 (特徴マップのみ)	88.11	28.48	**1.10**
提案手法	88.08	**28.62**	**1.10**

6. おわりに

本研究では，AEs を入力した際のネットワークの内部状態を分析し，事後確率と併せた AEs の検出手法を提案した．評価実験にて，従来手法と比較して，提案手法は AEs に対する堅牢性が高いことを確認した．また，事後確率と特徴マップによる検出を組み合わせることで AEs に対する堅牢性を向上させることが可能であることを確認した．今後は，特徴マップを取得する位置の変更や，統計方法の変更を行い分析を行う予定である．

参考文献

[1] I. J. Goodfellow, et al., “Explaining and Harnessing Adversarial Examples,” In International Conference on Learning Representations, pp. 1–11, 2015.

[2] Y. Song, et al., “Pixeldefend: Leveraging generative models to understand and defend against adversarial examples,” In International Conference on Learning Representations, pp. 1–20, 2018.

[3] W. Xu, et al., “Feature Squeezing: Detecting Adversarial Examples in Deep Neural Networks,” In Network and Distributed System Security Symposium, pp. 1–15, 2018.

[4] P. Sperl, et al., “DLA: Dense-Layer-Analysis for Adversarial Example Detection,” In IEEE European Symposium on Security and Privacy (EuroS&P), pp.198-215, 2020.

[5] A. Madry, et al., “Towards Deep Learning Models Resistant to Adversarial Attacks,” In International Conference on Learning Representations, pp. 1–28, 2018.

[6] A. V. Oord, et al., “Pixel Recurrent Neural Networks,” In International Conference on Machine Learning, pp. 1747–1756, 2016.