DCGANを用いたネットワークトラフィックの異常検出

日置 裕士[1,a)]　青木 茂樹[1,b)]　宮本 貴朗[1]

概要：本研究では IDS の検知精度を高めるため，画像処理の分野で高い性能を示している Deep Learning を用いて異常を検出する手法を提案する．一般的に Deep Learning においては，学習の際に教師データが必要であり，教師データの作成が課題となっている．ここでは Deep Learning の中でも教師データを必要としない DCGAN を使用する．まず正常なトラフィックデータを数値に変換し，画像とみなして DCGAN で学習する．その後，別の期間に収集したトラフィックデータに対して，学習した DCGAN を用いることで異常な通信と正常な通信の識別を行う．実験では MWS データセットを用いて本手法の有効性を確認した．

キーワード：ネットワーク異常検出, Deep Learning, GAN

1. はじめに

近年，サイバー攻撃などのネットワーク犯罪の増加に伴い，ネットワーク上の不正なトラフィックを検出する侵入検知システム (IDS: Intrusion Detection System) の研究が盛んに行なわれている．IDS はシグネチャ型とアノマリ型の 2 種類に大別することができる．代表的なシグネチャ型 IDS として，Snort[1] や Suricata[2]，The Bro[3] 等が挙げられる．シグネチャ型 IDS は異常を定義したパターンファイルに基づいて異常の検出を行う方式である．しかしパターンファイルに定義されていない攻撃については亜種を含め検出できないという欠点がある．文献 [4] では，シグネチャ型 IDS の検出結果から学習データを自動生成し，機械学習することで本来検出できない亜種攻撃を検知できる IDS を提案している．しかしこの手法では，学習データをパターンファイルに基づいて生成しているために，パターンファイルに登録されていない未知の異常については検出できないことが課題となっている．

一方，代表的なアノマリ型 IDS としては文献 [5,6,7] の手法が挙げられる．アノマリ型 IDS は正常な通信のみを含むデータを正常状態と定義し，そこから外れた状態を異常として検出する方式である．しかしアノマリ型 IDS ではシグネチャ型 IDS の問題点であった未知の異常の検出については可能であるが，シグネチャ型 IDS に比べ，検出精度が低いという欠点がある．近年，アノマリ型 IDS の検出精度を向上させるために様々な機械学習手法を用いた研究が行われている．その一つとして，文献 [8] ではマルウェアの通信挙動をモデルとして構築するために，マルウェアの通信挙動を表す特徴量を抽出し，DBSCAN によってクラスタリングを行い，異常を分類する手法を提案している．

本研究では，画像処理の分野において高い識別性能を示している Deep Learning に注目する．一般に Deep Learning による学習では，学習時に教師データが必要であるが，ネットワークの異常検出に関する研究において，正常/異常の教師ラベルが付いたデータを取得することは難しいため，一般的な Deep Learning の手法を用いることは難しい．そこで Deep Learning の手法の中でも，教師データを必要としない敵対型生成ネットワーク (GAN:Generative Adversarial Network) に注目する．GAN は文献 [9] によって提案された手法であり，Generator と Discriminator という 2 つのネットワークを用いて，学習した画像と類似した画像を精巧に生成することができるモデルである．文献 [10] では，文献 [9] の GAN に対して畳み込みニューラルネットワーク (CNN:Convolutional Neural Network) を組み込むことで，さらに精巧な画像を生成することが可能となった DCGAN(Deep Convolutional Generative Adversarial Network) を提案している．文献 [11] では．DCGAN を用いて医療画像から健康な患者と疾患を持つ患者を分類する手法を提案している．

本研究ではネットワークに流れるパケットを画像に変換して，DCGAN で学習を行い，正常な通信と異常な通信を高精度に識別する手法を提案する．また 2 種類のデータ

[1] 大阪府立大学
Osaka Prefecture University
[a)] sxa01232@edu.osakafu-u.ac.jp
[b)] aoki@kis.osakafu-u.ac.jp

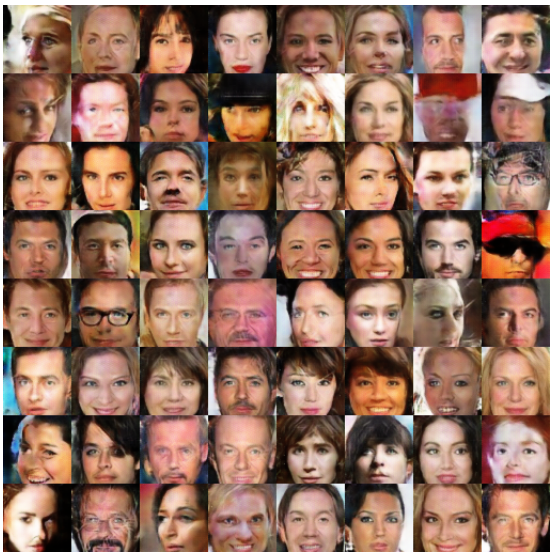

図 1　GAN による画像生成例

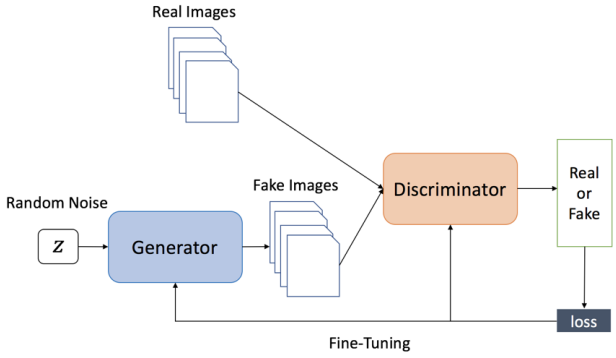

図 2　GAN の構成

セットを用いて実験を行い，提案手法の有効性を確認した．

2. 関連研究

本研究に関連する従来研究として，アノマリ型 IDS の代表的な手法である文献 [5,6,7,8] について述べる．文献 [5] では，パケットのエントロピーに基づく異常検出手法が提案されている．この手法ではまず，IP アドレスやポート番号など毎の単位時間当たりのパケット数を計測する．次に，パケットの発生確率を求め，求めた発生確率からエントロピーを算出する．その後，エントロピーの時系列変化に着目した EMMM 法により，エントロピーが大きく変化する時間を攻撃などが含まれている異常状態として検出している．

文献 [6] では，ネットワークのトラフィックは複数の正常状態で表されると考え，複数の正常状態を定義し，各状態との違いから異常検出する手法を提案している．この手法では，異常を含まないデータから単位時間当たりの ICMP や TCP パケット数等を計測してクラスタリングする．メンバが少ないクラスは削除し全てのクラスにおいて閾値以上のメンバ数となるまでクラスタリングを繰り返す．クラスタリング結果を正常状態として定義し，新たに観測されたデータから同様の特徴を抽出し，正常クラスとの距離が閾値以上かどうかで異常の判別を行っている．

文献 [7] では，複数の特徴量の組み合わせによる異常検出手法を提案している．この手法では，異常をトラフィック量の異常，通信手順の異常，通信内容の異常の 3 種類に分け，単位時間あたりのトラフィック量を数値化した特徴量，フロー毎のフラグの出現回数を数値化した特徴量，フロー内のパケットのペイロードのパターンの傾向を数値化した特徴量を学習用データからそれぞれ抽出する．そして新たなデータでこれらの特徴量を抽出し，学習用データの値と閾値以上離れている特徴量が存在する場合に異常であると判断する．

文献 [8] では，一般的な通信の特徴量を 70 種類，マルウェア通信の特徴量を 25 種類，計 95 種類の特徴量を抽出し，密度ベースのクラスタリング手法である DBSCAN を用いてクラスタリングを行い，マルウェアの通信パターンをモデル化している．このモデルに対し，新たなトラフィックから同様に抽出した特徴量を投影することで異常の検出を行っている．

本研究では IDS の検出精度を向上させるため，画像処理の分野で高い性能を示している Deep Learning に注目する．その中でも教師データを必要としない GAN を用いた手法である文献 [9,10,11] について述べる．文献 [9] では，敵対型生成ネットワーク (GAN) を発案しており，訓練データとなる画像を精巧に模倣した画像を生成できる手法を提案している．図 1 に顔画像 [12] を学習し，学習した GAN で生成された画像の例を示す．図より精巧な顔画像が生成できていることを確認できる．GAN の構成の概要を図 2 に示す．GAN は Generator と Discriminator という 2 つのネットワークによって構成されており，Generator が訓練データと同じようなデータを生成するように学習し，Discriminator は訓練データと Generator が生成したデータを正しく判別するように学習する．このときの Discriminator の判別の誤差を元に，Generator は Discriminator に訓練データと誤って判別されるような精巧に類似したデータを生成するように学習する．一方，Discriminator は Generator が生成したデータと訓練データを完全に判別できるように学習していく．このように 2 つのネットワークがお互いを高め合うように学習を行うことで精巧なデータを生成することが可能なモデルとなる．しかし GAN には学習が不安定という問題点がある．そこで文献 [10] では，GAN の 2 つのネットワークに畳み込みニューラルネットワーク (CNN:Convolutional Neural Network) を組み込むことで，GAN の学習を安定させ，より精巧な画像を生成すること

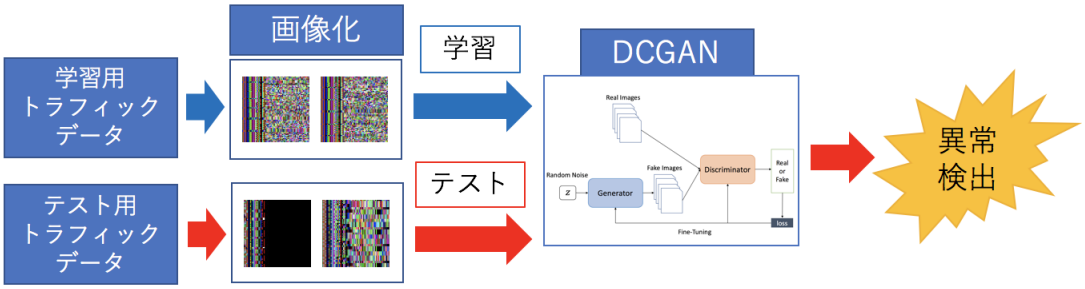

図 3　提案手法の概要

に成功している．

文献 [11] では，文献 [10] の DCGAN を用いて医療画像から異常検出を行っている．まず正常な画像を訓練データとして DCGAN で学習する．新たにテスト画像を用いて，学習後の Generator をパラメータを固定して学習する．訓練データと類似した正常な画像が含まれていれば，収束した後の誤差は小さくなると考えられる．また訓練データとは異なる異常な画像が含まれていれば収束した後の誤差は大きくなると考えられる．この性質を利用して算出された誤差を元に異常度を評価している．

本研究では，正常なトラフィックデータを収集して，画像として読み込み，文献 [10] の DCGAN によって学習を行い，文献 [11] の手法を参考にして異常度を算出し，正常な通信と異常な通信を分類する手法を提案する．

3. 提案手法

提案手法の概要を図 3 に示す．本手法は学習と異常検出の 2 つのプロセスに分かれている．まず学習処理では，正常なトラフィックデータを画像に変換し，DCGAN により学習を行う．その後，別の期間に収集したトラフィックデータを同様に画像に変換し，学習した DCGAN を用いて正常な通信と異常な通信を識別する．

3.1 トラフィックデータの画像への変換

DCGAN により学習を行うため，トラフィックデータを PNM 形式の画像に変換する．画像への変換の概要を図 4 に示す．まず対象となるトラフィックデータを pcap 形式でキャプチャし，64 パケットごとに分割する．分割された pcap ファイルを 1 パケットずつ読み込み，パケットの先頭 192 バイトを 8bit 単位で 0~255 の数値に変換する．そして変換した数値を 1 画素あたり RGB の 3 つの値に割り当てる．この処理を分割した 64 パケットに対して行い，64×64 画素の画像 1 枚に変換する．画像に変換したトラフィックデータ (以下，トラフィック画像) の例を図 5 に示す．図中 1 行が 1 パケットを表しており，画像全体で 64 パケットが表されている．また白色で表現されている部分

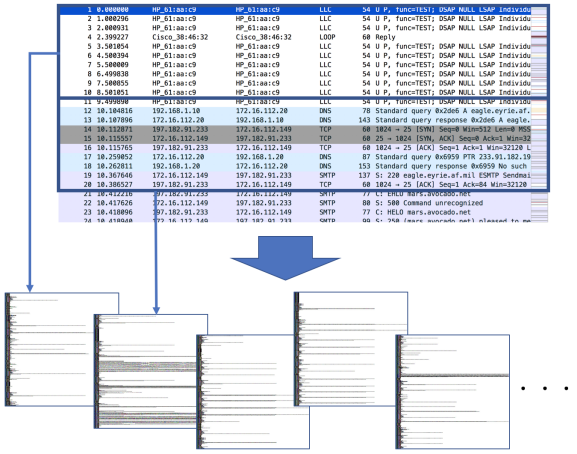

図 4　画像への変換の概要

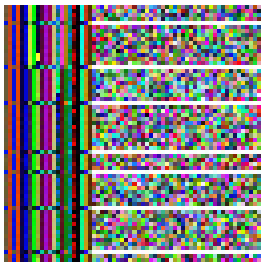

図 5　画像に変換されたパケットの例

はデータが含まれていない部分を表す．つまり画像中の左端部分はパケットのヘッダ部分であり，それ以外の部分はペイロード部分を表している．また正常通信のトラフィック画像とホストスキャン攻撃を含む通信のトラフィック画像の例を図 6 に示す．ホストスキャン攻撃は攻撃の対象となるホストを見つけるために，IP アドレスを増加させながら，ping によって応答を確認する攻撃である．そのためペイロード部分に何も含まれていないパケットが連続しているため，画像の右側部分が白色であることがわかる．このようにトラフィックデータを画像に変換することによってトラフィックの特徴を視覚化することができる．

3.2 DCGAN による学習

3.1 節で変換されたトラフィック画像の中から，攻撃通信

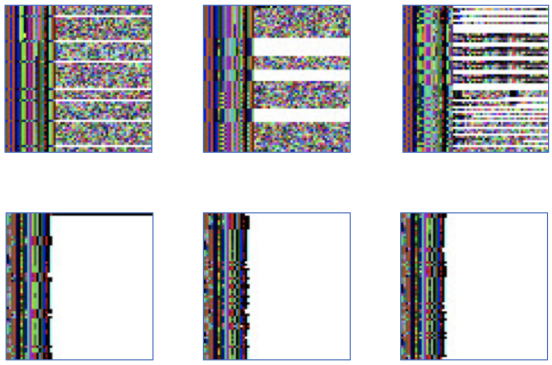

図 6 通常通信と攻撃通信のトラフィック画像の比較 (上段：正常通信 下段：ホストスキャン攻撃を含む通信)

が含まれていない正常通信のみの画像を用いて DCGAN で学習を行う．DCGAN は敵対型生成ネットワーク (GAN) に対して畳み込みニューラルネットワーク (CNN) を組み込んだモデルである．GAN は Generator と Discriminator の 2 つの学習器から構成される．Generator は潜在変数 $\boldsymbol{z}$ を入力とし，訓練データに類似したデータを生成するように学習する．Discriminator は Generator によって生成されたデータと訓練データを正しく識別できるように学習する．学習は Discriminator が判別を行う際に発生する誤差を基にこれら 2 つのネットワークを更新していくことで行っていく．次式は学習の過程を数式で表したものである．G は generator、D は discriminator、$\boldsymbol{x}$ は訓練データ、$\boldsymbol{z}$ は潜在変数を表す．

$$\min_G \max_D V(D,G) = \mathbb{E}_{\boldsymbol{x}\ p_{data}(\boldsymbol{x})}[logD(\boldsymbol{x})] + \mathbb{E}_{\boldsymbol{z}\ p_{\boldsymbol{z}}(\boldsymbol{z})}[log(1-D(G(\boldsymbol{z})))] \tag{1}$$

$G(\boldsymbol{z})$ は Generator によって潜在変数 $\boldsymbol{z}$ を入力として生成されたデータを，$D(\boldsymbol{x})$ はデータ $\boldsymbol{x}$ が訓練データである確率を表す．Discriminator は $D(\boldsymbol{x})$ を最大化しようとするのに対し，Generator は $\log(1-D(G(\boldsymbol{z}))$ を最小化しようとする．つまり Generator は Discriminator に訓練データと誤って判別されるような精巧なデータを生成することを目的に，Discriminator は Generator が生成したデータを完全に判別することを目的として学習する．このように学習を行うことで，2 つのネットワークはお互いに高め合い，訓練データに精巧に類似したデータを生成することができる．DCGAN では 2 つのネットワークに畳み込みニューラルネットワーク (CNN) を適用している．CNN は通常のニューラルネットワークに畳み込み層とプーリング層を追加することで識別精度の向上を図っている．GAN においても CNN を適用することで，従来より高解像度で画像を生成することに成功している．

3.3 異常検出

3.2 節で学習した DCGAN を用いて，文献 [11] を参考にして異常検出を行う．正常データにより DCGAN で学習を行なった際，学習後の Generator は正常データの分布 p に従って画像を生成するモデルとなっている．このモデルに新たなサンプル $\boldsymbol{x}$ を入力したとき，p に従ってサンプルした $\boldsymbol{x}$ であれば潜在空間内に $G(\boldsymbol{z}) \approx \boldsymbol{x}$ となる $\boldsymbol{z}$ が存在し，p とは異なった分布からサンプルした $\boldsymbol{z}$ であれば $\boldsymbol{x}$ を生成する $\boldsymbol{z}$ は存在しないと考えられる．したがって $\boldsymbol{z}$ が存在しない場合におけるサンプル $\boldsymbol{x}$ は異常と判断する．しかし Generator においてサンプル $\boldsymbol{x}$ から潜在変数 $\boldsymbol{z}$ を逆写像することはできないという問題点がある．その問題を解決するため以下の手順にしたがって潜在変数 $\boldsymbol{z}$ を探索する．

(1) $G(\boldsymbol{z}) \approx \boldsymbol{x}$ となる $\boldsymbol{z}$ を探すため，$\boldsymbol{z_0}$ をランダムにサンプリング

(2) $\boldsymbol{x}$ と $G(\boldsymbol{z_0})$ の誤差を次式で算出

$$L(\boldsymbol{z_0}) = \sum |\boldsymbol{x} - G(\boldsymbol{z_0})| \tag{2}$$

(3) 誤差を元に $\boldsymbol{z_0}$ を更新し，$\boldsymbol{z_1}$ とする

(4) 2, 3 を n 回繰り返し，最適な $\boldsymbol{z_n}$ を探索

探索した $\boldsymbol{z_n}$ を基に次式で異常度 $A(\boldsymbol{x})$ を算出する．式中の λ はハイパーパラメータとし，ここでは 0.1 とする．

$$A(\boldsymbol{x}) = (1-\lambda) \cdot L_R + \lambda \cdot L_D \tag{3}$$

ここで，

$$L_R = \sum |\boldsymbol{x} - G(\boldsymbol{z_n})| \tag{4}$$

$$L_D = \sum |D(\boldsymbol{x}) - D(G(\boldsymbol{z_n}))| \tag{5}$$

である．

4. 実験

本手法の異常検出精度を確認する実験を行った．評価方法は ROC(Receiver Operating Characteristic) 曲線および AUC 値を用いた．ROC 曲線とは正常と異常を分類する際の閾値を変化させたときの真陽性 (True Positive Rate) と偽陽性 (False Positive Rate) を計算し，プロットしたときの軌跡である．ここで，真陽性とは異常データに対して正しく異常とした割合を示し，偽陽性とは正常データを誤って異常と検出した割合を示す．すなわち ROC 曲線が左上方に近付くほど検出性能が高いことを示している．また ROC 曲線が左上方に近づいていることを表す指標として AUC(Area Under the Curve) 値を用いる．AUC 値は ROC 曲線の下側の面積を算出したものであり，この値が 1 に近いほど完全に分類できていることを示す．

表 1　データセット A の内訳

データセット	ラベル	アプリケーション・攻撃	枚数
学習	正常	Mail	1000
		Line	1000
		PostgreSQL	1000
テスト	正常	Mail	200
		Line	200
		PostgreSQL	200
	異常	f5_attack	87
		port_scan	282
		host_scan	664
		syn_flood	196

4.1　実験データ

以下の 2 種類のデータセットを用いて実験を行った．

4.1.1　データセット A

実験に使用するトラフィックデータは Mac mini(Late 2012)，macOS バージョン 10.12.6 を用いて取得した．まず，正常なトラフィックデータの例として 3 つのアプリケーション (Mail，Line, PostgreSQL) を稼働させ，トラフィックデータを取得した．取得したトラフィックデータをアプリケーション毎に 2 分割し，それぞれを学習用とテスト用データとして用いる．ここで取得したトラフィックデータには使用したアプリケーションとは別のバックグラウンドで実行されているプロセスの通信も混入している．次に異常なトラフィックデータの例として，4 種類の攻撃 (f5_attack, port_scan, host_scan, syn_flood) を行ったときのトラフィックデータを取得する．異常なトラフィックデータはテスト用データとして用いる．各アプリケーションの稼働時に取得した正常なトラフィックデータと 4 種類の攻撃を行った異常なトラフィックデータを画像に変換した時の枚数を表 1 に示す．学習データは 3000 枚，テストデータは 1829 枚使用した．

4.1.2　データセット B

MWS2018 データセットの BOS データセット [12] を使用した．BOS データセットは，組織内ネットワークへの侵害活動を想定した研究用データセットであり，本研究の目的である実ネットワークにおける異常検出に適していると考えられる．学習データとして，このデータセットの中でもマルウェアの進行度の低いトラフィックデータである 2017 年 8 月 17 日に観測されたデータを使用した．このデータセットは進行度が 2 であり，マルウェアによるパケットは含まれていない．テストデータとしてはマルウェアの進行度の高い 2018 年 1 月 23 日に観測されたデータセットを使用した．このテストデータは進行度が 8 であり，マルウェアを実行後にマルウェアの通信が継続的に観測されている．そして本研究では，マルウェアの通信相手 (特定の IP アドレス) との通信を攻撃，それ以外の通信を正常と定義している．

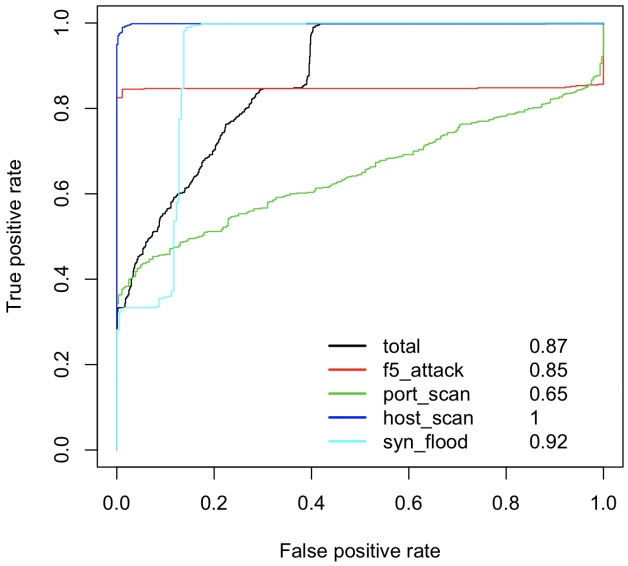

図 7　データセット A を用いた時の実験結果

4.2　実験結果および考察

4.2.1　実験 1

データセット A を用いて実験を行った．実験の結果を ROC 曲線および AUC 値によって評価した結果を図 7 に示す．図中の黒線で表されるグラフはデータセット全体を用いて ROC 曲線を描画した結果であり，その他のグラフは各攻撃ごとに描画した場合の結果である．全体としては AUC 値が 0.87 と高い結果となった．攻撃ごとの結果を見ると，F5 アタック，ホストスキャン，SYN フラッド攻撃については 0.85 以上の AUC 値となったが，ポートスキャンについては AUC 値が 0.65 と低い結果となった．正常なトラフィック画像と攻撃ごとのトラフィック画像の比較を図 8 に示す．図より AUC 値が高くなった 3 種類の攻撃 (f5_attack, host_scan, syn_flood) については正常なトラフィック画像と明確な差異があることがわかる．しかしポートスキャンについては，ポートスキャン以外の正常な通信が多数含まれていたため，正常なトラフィック画像と類似していることがわかる．そのため，異常値が低くなり，AUC 値が低くなったと考えられる．

これらの結果より，学習データと異なる特徴を持ったトラフィック画像については分類が可能であることがわかった．またポートスキャンのように攻撃通信を含んでいるものの，正しく分類できないものもあった．攻撃通信の特徴を反映した画像への変換方法の検討が今後の課題として挙げられる．

4.2.2　実験 2

データセット B を用いて実験を行った．テストデータへのラベル付けとしてマルウェアの通信相手 (以下，C&C

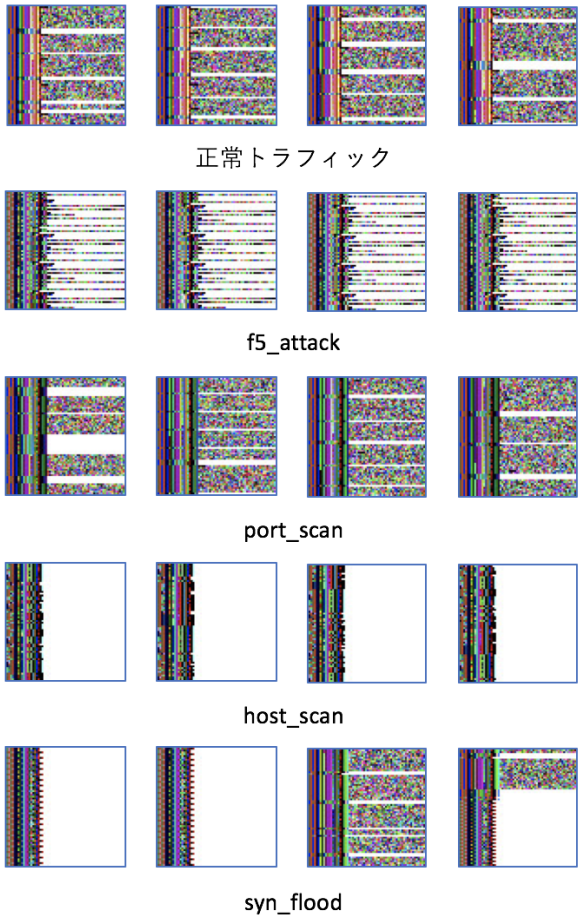

図 8 正常なトラフィックと異常なトラフィックの画像比較

サーバ) との通信を行っているパケットの数が N 以上の場合，異常とした．N の値を 1, 15, 20, 25, 30 と変化させたときの ROC 曲線および AUC 値を図 9 に示す．またテストデータにおける C&C サーバとの通信を行ったパケット数毎の比較を図 10 に示す．図 9 より，一枚のトラフィック画像 64 パケット分の中に C&C サーバとの通信が 25 パケット以上含まれる画像を異常としたとき，最も検出性能が高く，30 以上の場合は検出性能が下がることがわかった．図 10 においても C&C サーバとの通信が多いトラフィック画像は特徴的であることがわかる．

また全体として AUC 値が低くなった原因としては，データセット B ではデータセット A ほど攻撃の特徴が明確に現れていなかったため，データセット B におけるマルウェア特有の特徴を DCGAN によって捉えることができなかったためであると考えられる．これを解決するためには，まず DCGAN のパラメータを調整することでより特徴を捉えるように改善することが挙げられる．また攻撃の特徴が現れやすい様な画像への変換手法の検討が必要であると考えられる．

ここで，データセット B のデータについて，通信の相手先 IP アドレスごとにトラフィックデータを抽出し，トラフィック画像に変換した例を図 11 に示す．ここで，ホスト A, B, C はマルウェアを実行したホストと同一のサブネット内のホストである．図より相手先ホストごとにトラフィック画像の特徴が異なっていることがわかる．このように，画像への変換方法を変更することによって，通信の特徴を顕著に表すことができる様になるため，本手法による異常の識別性能が向上すると考えられる．しかしながら，ホスト毎にトラフィック画像を生成する場合，ホストスキャンのような不特定多数のホストと通信を行う攻撃の検出が難しくなることや，トラフィック画像中の特定の IP アドレス等の情報が埋め込まれているため，画像中に含まれている情報の学習・検出への影響が懸念される．今後の課題として，DCGAN のパラメータ調整と共に，攻撃の特徴が現れやすいトラフィック画像の生成方法並びに学習・検出方法の検討などが挙げられる．

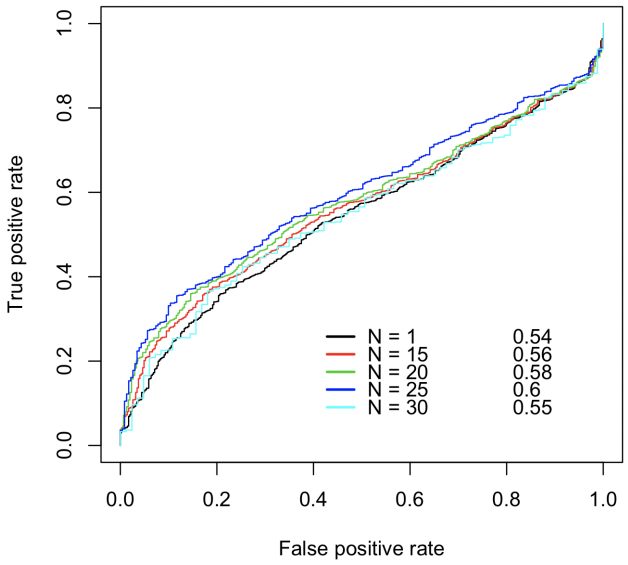

図 9 データセット B を用いた時の実験結果

5. まとめ

本論文では，Deep Learning の手法の一つである DCGAN を用いて，ネットワークの異常を検出する手法を提案した．実験では 2 種類のデータセットを用いて本手法の有効性を確認した．実験の結果，正常なトラフィックと異常なトラフィックに明確に差異があった場合は分類可能であるが，差異が小さい場合は分類することができなかった．今後の課題としては，DCGAN のパラメータ調整や画像への変換方法の検討が挙げられる．

参考文献

[1] Snort, <https://www.snort.org/>(参照 2018-08-09).
[2] Suricata, <http://suricata-ids.org/>(参照 2018-08-09).
[3] The Bro, <http://www.bro.org/>(参照 2018-08-09).

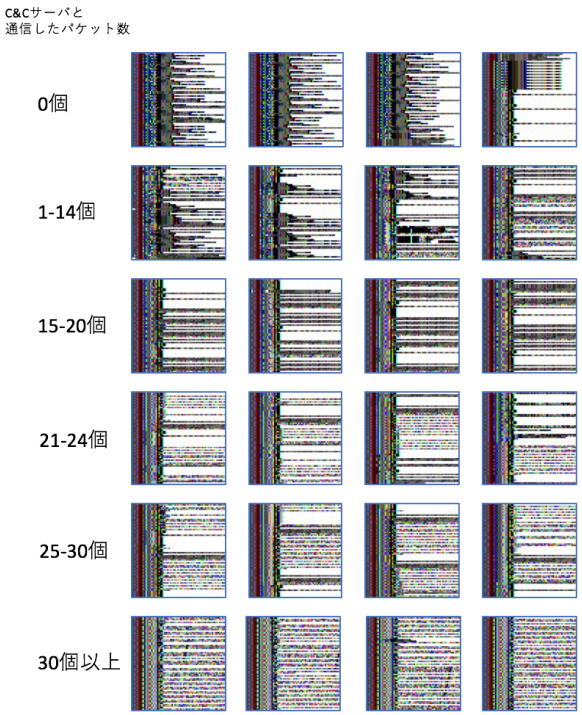

図 10 C&C サーバとの通信を行ったパケット数毎のトラフィック画像の例

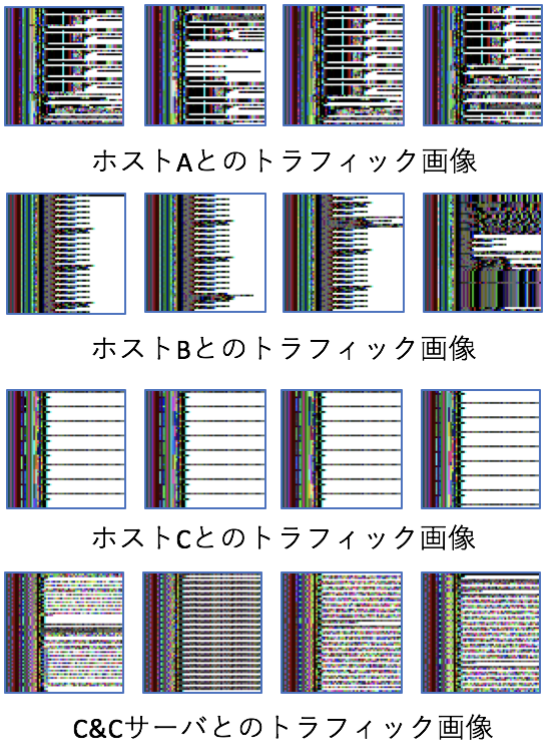

図 11 画像化方法を変更したときのトラフィック画像の例

[4] 山田明, 三宅優, 田中俊昭：亜種攻撃を検知できる侵入検知システム，信学技報，ISEC2004-31，pp.119-126(2004).

[5] 小島俊輔, 中嶋卓雄, 末吉敏則：エントロピーベースのマハラノビス距離による高速な異常検知手法，情報処理学会論文誌, Vol.52, No.2, pp.656-668(2011).

[6] 佐藤陽平, 和泉勇治, 根元義章：複数の検出モジュールの組み合わせによるネットワーク異常検出の高精度化, 信学技報, NS2004-144, pp.45-48(2004).

[7] 平松尚利, 和泉勇治, 角田裕：複数の通常状態を用いたネットワーク異常検出，信学技報, CS2006-32, pp.61-66(2006).

[8] Hatada, Mitsuhiro, and Tatsuya Mori : Finding New Varieties of Malware with the Classification of Network Behavior, IEICE TRANSACTIONS on Information and Systems, vol.E100.D, no.8, pp.1691-1702(2017)

[9] Lan Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio : Generative adversarial nets. In Advances in Neural Information Processing Systems, pp 2672-2680(2014).

[10] Alec Radford, Luke Metz, and Soumith Chintala : Unsupervised representation learning with deep convolutional generative adversarial networks, arXiv preprint arXiv:1511.06434(2015).

[11] Thomas Schlegl, Philipp Seebock, Sebastian M Waldstein, Ursula Schmidt-Erfurth, and Georg Langs : Unsupervised anomaly detection with generative adversarial networks to guide marker discovery. In International Conference on Information Processing in Medical Imaging, pp 146-157(2017).

[12] Large-scale CelebFaces Attributes (CelebA) Dataset, <http://mmlab.ie.cuhk.edu.hk/projects/CelebA.html/>(参照 2018-08-16).

[13] 高田雄太，寺田真敏，松木隆宏，笠間貴弘，荒木粧子，畑田充弘：マルウェア対策のための研究用データセット～MWS Datasets 2018～，情報処理学会論文誌, Vol.2018-CSEC-82，No.38，pp.1-8(2018).