

卒業論文概要書

Summary of Bachelor's Thesis

Date of submission: 02/06/2019 (MM/DD/YYYY)

学科名 Department	情報通信	氏名 Name	丸寄佳奈子	指導 教員 Advisor	渡辺 裕 ㊞
研究指導名 Research guidance	オーディオビジュアル 情報処理研究	学籍番号 Student ID number	1w152355-8 ^{CD}		
研究題目 Title	Capsule Network を用いた敵対的生成ネットワークによる画像生成 Image Generation by Generative Adversarial Network Using Capsule Network				

1. まえがき

近年, Convolutional Neural Network (CNN) [1] を用いた画像処理が数多く提案されている. 新しいデータを生成する敵対的生成ネットワークである Generative Adversarial Network (GAN) [2] もその一つであり, CNN を利用した DCGAN[3] などが存在する. CNN を用いた画像処理には, データセットとして大量の画像データが必要となるが, データを大量に収集することは困難である. そこで, GAN により画像データを大量に生成することができれば, データセットの量を補うことができると考えられる.

しかしながら, CNN には画像の特徴間の空間的情報が失われるという欠点がある. 2017 年に CNN の欠点を補うことのできる Capsule Network[4] が発表された. そこで, 本研究では CNN の欠点を補い, CNN よりも画像の特徴を捉えることに優れる Capsule Network を GAN の構造に組み込んだ Capsule GAN を提案する.

2. Capsule Network

Capsule Network とは, CNN をベースとしたニューラルネットワークである. 各ニューロンの入力をスカラーでなくベクトルとすることで, 画像の特徴間の空間情報の保持が可能となる. 画像を特徴間の空間的情報を保持したまま処理し, 出力層として DigitCaps 層を出力する. DigitCaps 層は, 画像の特徴をもった 16 次元のベクトルの集合となっている. ベクトルの数は識別クラス分あり, ベクトルの大きさが識別クラスの確率となる.

3. 提案手法

本研究で提案する Capsule GAN の構造を図 1 に示す. Discriminator には Capsule Network を CNN のように組み込んでいる. Generator には, Discriminator から取り出した画像の特徴を保持している DigitCaps

層を潜在変数と掛け合わせ, 入力としている.

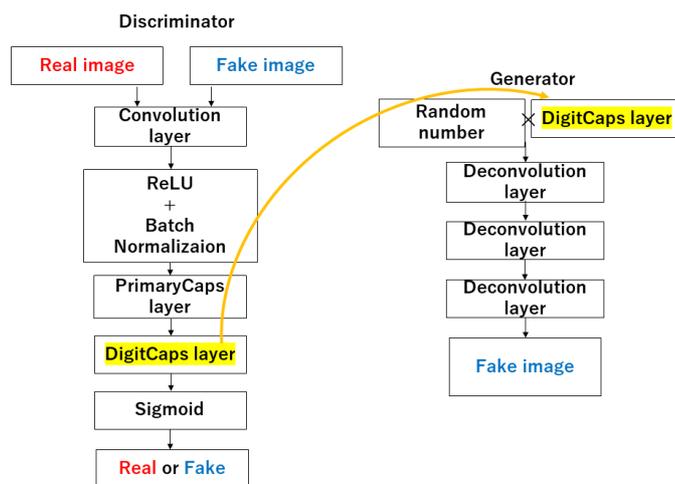


図 1 Capsule GAN の構造

4. 実験

MNIST[5] とカラー画像である猫画像をデータセットとして, それぞれ Capsule GAN を用いて画像を生成した. 実験結果を図 2, 図 3 に示す. また, 生成画像を品質の良い画像, 悪い画像と分けた. 実験結果を図 4 に示す.

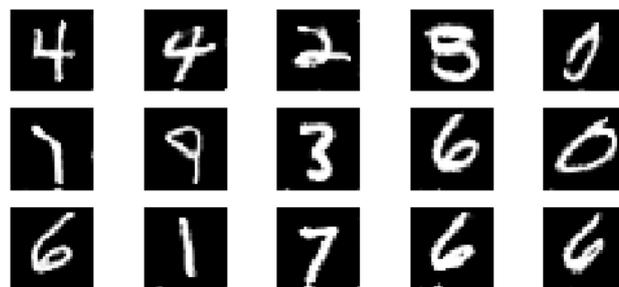


図 2 生成画像 (MNIST)



図3 生成画像 (猫画像)



図4 品質の良い画像と悪い画像の比較

図4よりMNIST, 猫画像の両方において生成画像の品質にばらつきがあることがわかる. MNISTでは, 画像の外周が欠ける画像が生成された. これは, 画像を生成する過程で値のオーバーフローが生じたと考えられる. また, 猫画像はMNISTよりも複雑な画像であるため, GANによる学習が難しく品質の悪い画像が生成されたと考えられる.

MNISTを用いてDCGANでも同様に画像を生成し, Capsule GANとの比較を行った. 比較にはInception Score (IS) [6]とFréchet Inception Distance (FID) [7]を使用した. 画像はそれぞれ10000枚用いた. 比較を行う際, 外周の欠けていない画像のみを100枚使用した比較も同様に行った. 結果を表1に示す.

表1 DCGANとの比較結果

モデル	IS	FID
Capsule GAN (10000 枚)	2.37	14.14.
DCGAN (10000 枚)	2.32	7.65
Capsule GAN (100 枚)	2.58	57.56
DCGAN (100 枚)	2.39	57.87

表1より, ISは画像を10000枚, 100枚使用した場合のどちらともCapsule GANの方がそれぞれ0.05, 0.19の値の向上を確認できた. FIDでは10000枚使用した場合はCapsule GANの方が, 6.49値が悪くなった. 一方, 画像を100枚使用した場合にはCapsule GANの方が0.31の値の向上が確認できた. したがって, 画像の外周を除いた数字のみの評価ではCapsule GANの方がDCGANよりも性能が良いことがわかる. このことから, Capsule GANはDCGANよりも画像の特徴を捉え, 画像を生成することができる事が確認できた.

5. まとめ

本研究では, Capsule Networkを用いたGANであるCapsule GANを提案するとともに, 画像を生成し, DCGANとの比較を行った. また, 外周が欠けている画像を除いた場合にはCapsule GANがDCGANよりも優れていることを示すことができた. 今後の課題として, より複雑な画像への対応および画像の外周が欠ける原因を明らかにする必要がある.

参考文献

- [1] A. Krizhevsky, I. Sutskever and G. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks", Neural Information Processing Systems (NIPS), pp. 1106-1114, Dec. 2012.
- [2] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville and Y. Bengio, "Generative Adversarial Networks", Neural Information Processing Systems (NIPS), Dec. 2014.
- [3] A. Radford, L. Metz and S. Chintala, "Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks", International Conference on Learning Representations (ICLR), Jan. 2016.
- [4] S. Sabour, N. Frosst and G. E. Hinton, "Dynamic Routing Between Capsules", Neural Information Processing Systems (NIPS), Dec. 2017.
- [5] Y. LeCun, C. Cortes and C. J. C. Burges, "The MNIST Database of Handwritten Digits", <http://yann.lecun.com/exdb/mnist/>, 1998.
- [6] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford and X. Chen, "Improved Techniques for Training GANs", Neural Information Processing Systems (NIPS), Dec. 2016.
- [7] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler and S. Hochreiter, "GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium", Neural Information Processing Systems (NIPS), Dec. 2017.

2018年度 卒業論文

Capsule Network を用いた敵対的生成ネットワーク
による画像生成

Image Generation by Generative Adversarial Network
Using Capsule Network

指導教員 渡辺 裕 教授

早稲田大学 基幹理工学部

情報通信学科

1W152355-8

丸寄 佳奈子

目次

第1章 序論.....	1
1.1 研究の背景.....	1
1.2 本研究の目的.....	1
1.3 関連研究.....	1
1.4 本論文の構成.....	2
第2章 ディープラーニング.....	3
2.1 まえがき.....	3
2.2 ディープラーニングの概要.....	3
2.3 CNN.....	3
2.3.1 畳み込み層 (Convolution 層).....	4
2.3.2 プーリング層 (Pooling 層).....	5
2.3.3 CNN の欠点.....	5
2.4 Capsule Network.....	6
2.5 GAN.....	7
2.5.1 GAN の概要.....	7
2.5.2 DCGAN.....	8
2.5.3 評価方法.....	8
2.5.3.1 Inception Score (IS).....	8
2.5.3.2 Fréchet Inception Distance (FID).....	8
2.6 むすび.....	9
第3章 提案手法.....	10
3.1 まえがき.....	10
3.2 Capsule GAN の構造.....	10
3.3 むすび.....	11
第4章 実験.....	12
4.1 まえがき.....	12
4.2 データセット.....	12
4.2.1 MNIST.....	12
4.2.2 猫画像.....	12
4.3 提案手法に基づいた実験.....	12
4.3.1 MNIST を用いた結果.....	12
4.3.2 猫画像を用いた結果.....	13
4.3.3 考察.....	14

4.3.3.1 MNIST を用いた結果の考察	14
4.3.3.2 猫画像を用いた結果の考察	15
4.3.3.3 MNIST と猫画像を用いた結果の比較	17
4.3.4 Capsule GAN の修正	17
4.4 提案手法を修正した実験	18
4.4.1 MNIST を用いた結果	18
4.4.2 猫画像を用いた結果	19
4.4.3 DCGAN との比較	20
4.4.4 考察	22
4.4.4.1 MNIST を用いた結果の考察	22
4.4.4.2 猫画像を用いた結果の考察	22
4.4.4.3 DCGAN との比較結果の考察	22
4.5 むすび	22
第 5 章 結論と今後の課題	23
5.1 結論	23
5.2 今後の課題	23
謝辞	24
参考文献	25
図一覧	27
表一覧	28
研究業績	29

第1章 序論

1.1 研究の背景

近年, Convolutional Neural Network (CNN) [1]を用いた画像処理が数多く提案されている. 顔認識[2], 白黒画像のカラー化[3]などが例に挙げられる. 新しいデータを生成する敵対的生成ネットワークである Generative Adversarial Network (GAN) [4]もその一つであり, CNN を用いた Deep Convolutional GAN (DCGAN) [5]などが存在する. しかし, CNN には画像の特徴間の空間的情報が失われるという欠点がある. 2017 年に CNN の欠点を補った Capsule Network[6]が発表された. CNN に比べ画像の特徴を捉えることに優れる Capsule Network を GAN に取り入れれば, より品質の良いデータを生成することができると考えられる. そこで, 本研究では Capsule Network を GAN の Discriminator, Generator の両方の構造に組み込んだ Capsule GAN を提案する.

1.2 本研究の目的

CNN を用いた画像処理には, 大量の画像データが必要となる. しかしながら, 大量の画像データを収集することは難しく, 公開されているデータセットなどが無い場合は画像データを全て自分で収集する必要がある, 自作の CNN を構築することは困難となる. そのため, GAN により画像データを大量に生成することができれば, 大量のデータセットを作成することが可能となり, 自作の CNN を構築することができる. しかし, GAN で生成された画像を CNN のデータセットとして使用するため, 生成画像は本物の画像と区別がつかないレベルの品質が求められる. したがって, 本研究では Capsule GAN により, 品質の良い画像を生成することを目的とする.

1.3 関連研究

Capsule Network を用いた GAN の研究として, Huseyn Gadirov らの研究[7]が挙げられる. Huseyn Gadirov らは, GAN の Discriminator のみに Capsule Network を組み込んでいる. しかしながら, Generator の構造は DCGAN のように CNN を用いており, Capsule Network を組み込んでいない. また, MNIST[8]では画像の生成に成功しているが, カラー画像の CIFAR10[9]では失敗している.

1.4 本論文の構成

本論文の構成を以下に示す.

第1章は本章であり, 本研究の背景, 目的及び関連研究について述べる.

第2章では本研究で用いるディープラーニングの関連用語について述べる.

第3章では本研究で提案する手法について述べる.

第4章では提案手法の実験, 結果及び考察について述べる.

第5章では本論文の結論と今後の課題について述べる.

第2章 ディープラーニング

2.1 まえがき

本章では、本研究で用いるディープラーニングの技術である CNN, Capsule Network, GAN, 及び GAN の評価方法について述べる.

2.2 ディープラーニングの概要

ディープラーニングとは、階層の深いニューラルネットワークを用いる機械学習の一つである. ニューラルネットワークとは、人間の脳の神経細胞であるニューロンの構造を模したコンピュータ処理の仕組みのことである. ディープラーニングは、ニューラルネットワークを利用することにより画像認識や音声認識の分野で従来の手法よりも高い性能を達成している[10].

2.3 CNN

CNN とは、画像認識や音声認識などで多く用いられているニューラルネットワークの一種である[11]. CNN の基本構造を図 2.1 に示す. 各層は複数のノードからなる. CNN では、畳み込み層、プーリング層、全結合層を隠れ層に持つ. 畳み込み層とプーリング層は交互に組み込まれるが、それぞれの層の数は構造により異なる. これら二つの層により、データの特徴を得ることができる. 全結合層では、得られた特徴を一つのノードに結合し、活性化関数により変換された値を出力する. 全結合層の出力を出力層で確率に変換することで、入力データを識別、分類することが可能となる.

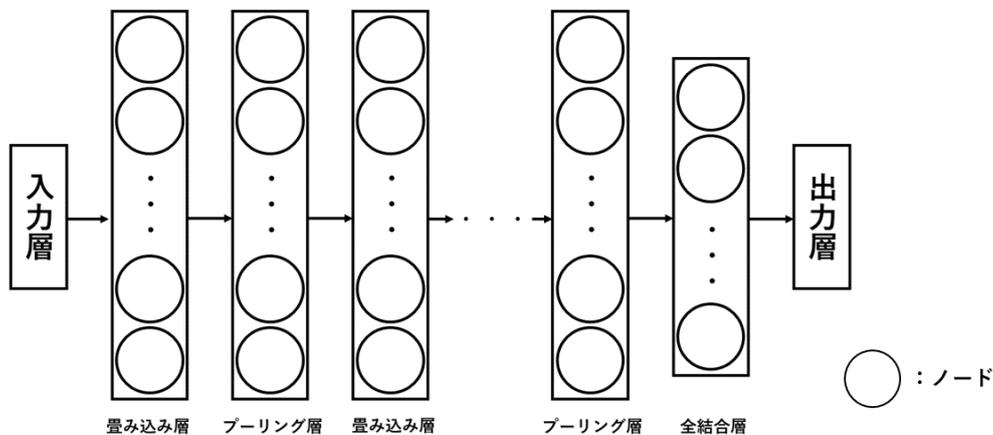


図 2.1 CNN の基本構造

2.3.1 畳み込み層 (Convolution 層)

畳み込み層では、入力データに対してフィルタを畳み込む演算を行う[12]。フィルタとは、入力データの特徴を抽出する検出器であり、格子状に数値を格納している。畳み込みの概略を図 2.2 に示す。フィルタを畳み込むことにより、入力データの特徴をもつ特徴マップが出力される。複数のフィルタを用いると、使用したフィルタの数だけ特徴マップが出力される。図 2.2 では、 3×3 のフィルタをスライドさせながら画像を畳み込んでいる。フィルタと画像を掛け合わせた値が特徴マップの値となる。出力された特徴マップは、各フィルタにより入力データのそれぞれ異なる特徴を捉えている。したがって、畳み込み層での処理により、入力データの特徴を得ることができる。

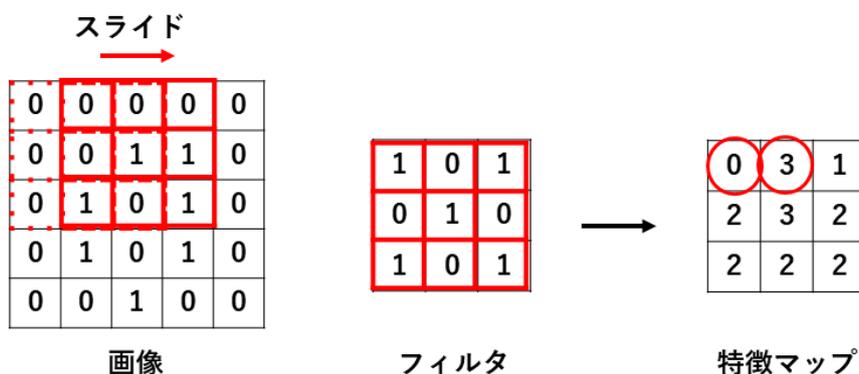


図 2.2 畳み込みの概略図

2.3.2 プーリング層 (Pooling 層)

プーリング層では、入力された特徴マップを縮小し、データを扱いやすくする[13]。プーリングの概略を図 2.3 に示す。図 2.3 では、特徴マップを 2×2 の領域に分け、領域ごとに最大値をとる最大プーリングを行っている。最大プーリングの他に、領域ごとの平均をとる平均プーリングなどがある。プーリングを行うことで、特徴の位置感度を低下させることができる。入力データが画像の場合、プーリング層での処理により、画像の位置変化に強くなる。したがって、画像が数ピクセル移動、回転していてもそれらの違いを吸収することができる。

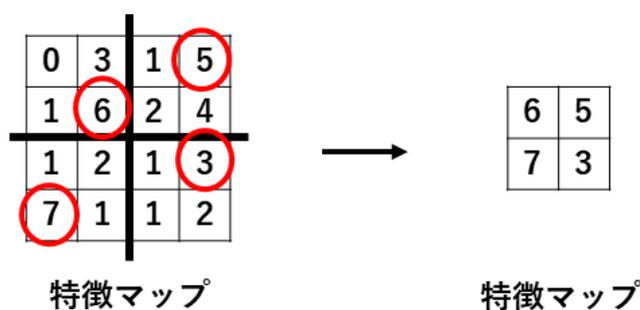


図 2.3 プーリングの概略図

2.3.3 CNN の欠点

CNN は、プーリング層により画像の位置変化に強くなるが、一方で画像の空間的情報を損失してしまうという欠点がある。画像の空間的情報の損失について、図 2.4 より人の顔を例にして説明する。CNN では、人の顔のパーツである目、鼻などの特徴を畳み込み層で捉え、プーリング層で特徴マップを縮小する。畳み込み層が顔のパーツの特徴をそれぞれ捉えると、画像内でパーツの配置がバラバラになっていたとしても、プーリング層が特徴マップを縮小することでその画像を人の顔だと判断し、特徴間の位置関係を無視してしまう[14]。したがって、プーリング層での処理により CNN は画像の特徴間の位置関係である空間的情報を損失する。

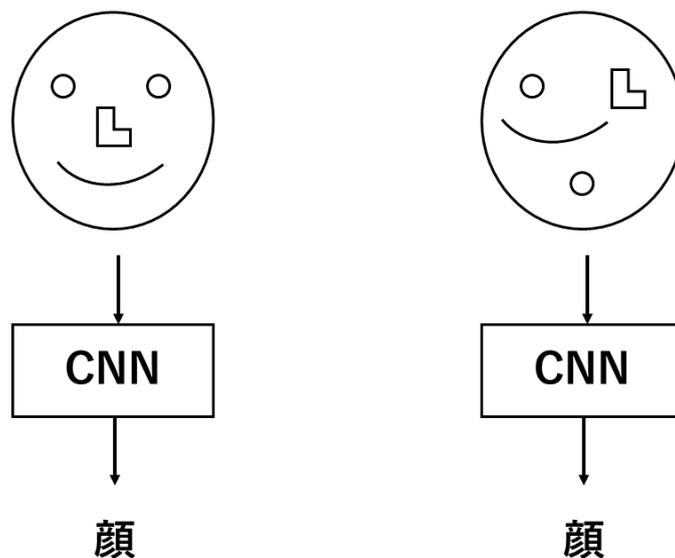


図 2.4 CNN の誤認識

2.4 Capsule Network

Capsule Network とは, CNN をベースとしたニューラルネットワークである. CNN は, プーリング層により特徴間の空間的情報が失われるという欠点がある. Capsule Network ではプーリング層をなくし, 各ニューロンへの入力をスカラーではなくベクトルとすることで, その欠点を補うことを可能とした. 図 2.5 に Capsule Network の基本構造を示す. まず, 画像に対して畳み込み処理を 2 回行う. 1 回目と 2 回目の処理後の特徴マップの数は変わらないが, 2 回目の処理後では特徴マップを 8 個ずつにグループ化し, 8 次元のベクトルとしている. この 8 次元のベクトルをカプセル, 層を PrimaryCaps 層と呼ぶ. PrimaryCaps 層のベクトルに対して重みをかける dynamic routing と呼ばれる処理を繰り返す[15]. Dynamic routing により, 16 次元のベクトルが識別クラス分出力される. それぞれのベクトルの大きさを求め, 最も値が大きいクラスが識別結果となる. 16 次元のベクトルは, 画像の特徴を含んでいる.

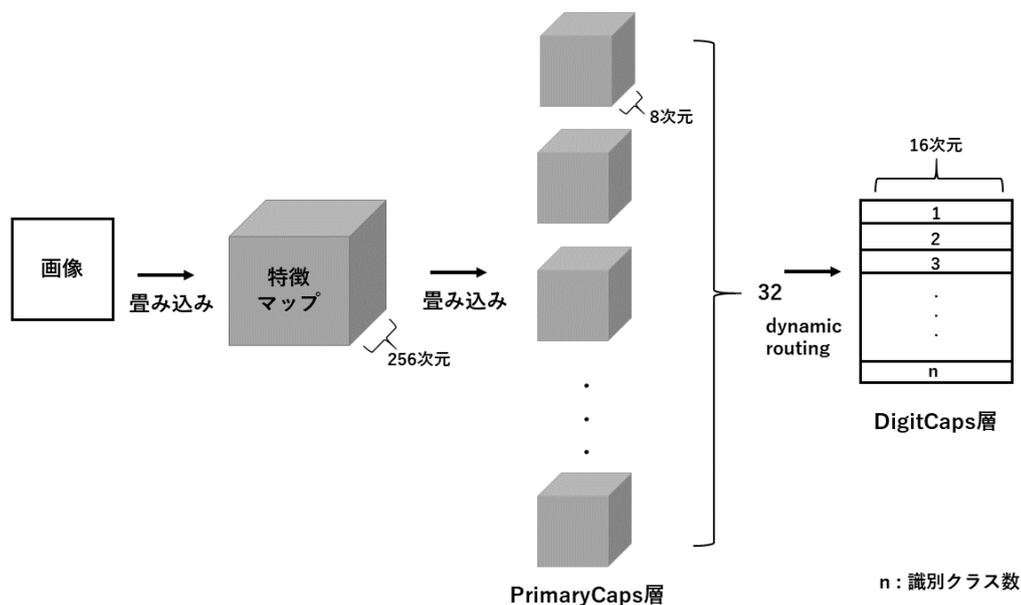


図 2.5 Capsule Network の構造

2.5 GAN

2.5.1 GAN の概要

GAN とは, Discriminator, Generator と呼ばれる二つのネットワークを用いて, 入力データと似たようなデータを生成するアルゴリズムである[16]. 図 2.6 に GAN の基本構造を示す. Generator は乱数を入力とし, データセットと似たようなデータを生成し, 出力する. Discriminator は, Generator が生成したデータと訓練データセットを入力とし, 入力データがデータセット (本物) か生成データ (偽物) かを識別する. Generator は Discriminator を騙せるように学習し, Discriminator は Generator に騙されないように学習していく.

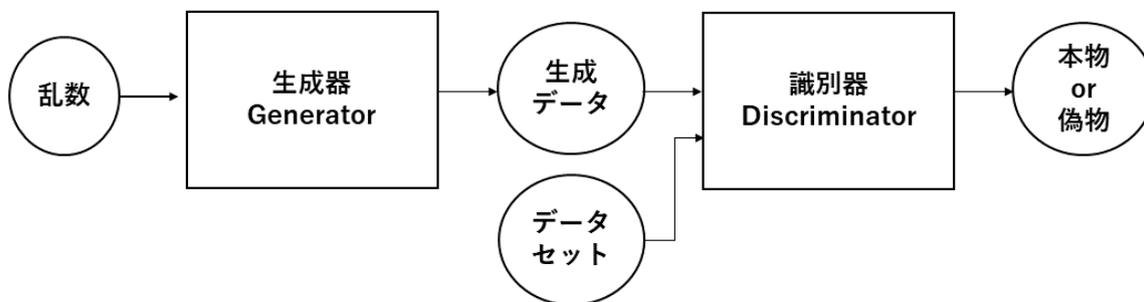


図 2.6 GAN の基本構造

2.5.2 DCGAN

DCGAN とは, GAN の Discriminator, Generator の構造に CNN を組み込んだ GAN である. GAN は学習を進めていくことが難しいが, DCGAN ではプーリング層, 全結合層の廃止などといった手法[17]を提案することにより学習の安定に成功している.

2.5.3 評価方法

2.5.3.1 Inception Score (IS)

Inception Score (IS) [18]とは, GAN を評価する際に使用される評価指標である. 画像が Inception モデル[19]で識別しやすく, かつ識別されるラベルの種類が多いほど IS が高くなる. IS は, 値が大きいくほど生成画像が良い画像であると評価する[20]. x_i を*i*番目の画像データ, y をラベル, *i*番目の画像を Inception モデルに入力して得られるラベル y の確率を $p(y|x_i)$, 画像データ全体のラベル y の確率を $p(y)$, 使用する画像の集合を X とする. IS は, $p(y|x_i)$ と $p(y)$ の確率分布の KL ダイバージェンスを求めたものである. 式を以下に示す.

$$\text{IS} = \exp\left(\frac{1}{X} \sum_{x_i \in X} p(y|x_i) \log \frac{p(y|x_i)}{p(y)}\right) \quad (2.1)$$

$p(y|x_i)$ と $p(y)$ の確率分布の差が大きいくほど, IS は大きくなる.

2.5.3.2 Fréchet Inception Distance (FID)

Fréchet Inception Distance (FID) [21]とは, IS と同様に GAN を評価する際に使用される評価指標である. GAN で生成した画像の集合 A_1 と, GAN で生成したい画像の集合 A_2 の距離を計算した値が FID となる. FID は値が小さいほど生成画像が良い画像であると評価する[22]. 画像の集合 A_1, A_2 の平均ベクトルを μ_1, μ_2 , 共分散行列を Σ_1, Σ_2 とすると, FID は以下の様に求められる.

$$\text{FID} = |\mu_1 - \mu_2|^2 + \text{tr}\left(\Sigma_1 + \Sigma_2 - 2(\Sigma_1 \Sigma_2)^{\frac{1}{2}}\right) \quad (2.2)$$

2.6 むすび

本章では、本研究で用いるディープラーニングの技術である CNN, Capsule Network, GAN, 及び GAN の評価方法について述べた.

第3章 提案手法

3.1 まえがき

本章では，本研究で提案する手法の概要について述べる．本研究では，GAN の Discriminator, Generator の両方に Capsule Network を組み込んだ Capsule GAN を提案する．

3.2 Capsule GAN の構造

図 3.1 に本研究で提案する Capsule GAN の構造を示す．

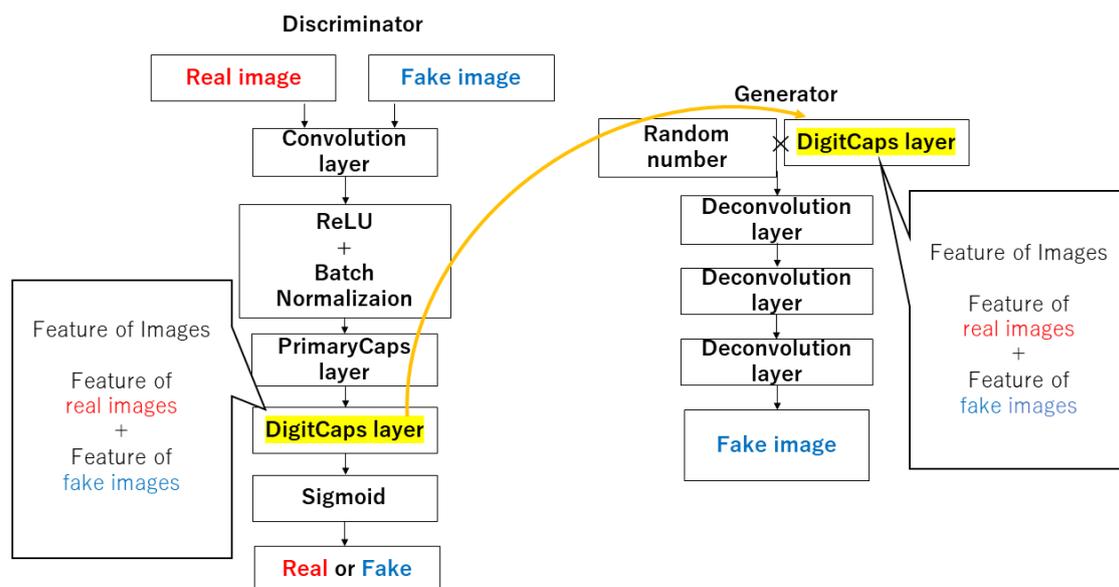


図 3.1 Capsule GAN の構造

まず，Discriminator の構造について説明する．Discriminator は訓練データセットの画像，Generator が生成した画像を入力とする．画像を Capsule Network に通し，出力として入力された画像が訓練画像（本物）か生成画像（偽物）かを出力する．また，画像の特徴を含んでいる DigitCaps 層を Generator に使用するため取り出す．Discriminator に入力する画像は，訓練画像，生成画像の順に入力される．そのため，DigitCaps 層の出力も訓練画像の特徴，生成画像の特徴の順番で出力される．次に，Generator の構造について説明す

る。Generator は Discriminator から取り出した DigitCaps 層と乱数を掛け合わせた値を入力とする。その後、入力値を DCGAN と同様に Deconvolution 層と呼ばれる Transposed Convolution 層に通し画像を生成する。したがって、本研究で提案する GAN の構造は、Generator の入力値に画像の特徴をもった DigitCaps 層を使用することにより Discriminator, Generator の両方に Capsule Network を用いた構造となっている。

3.3 むすび

本章では、本研究で提案する Capsule GAN の構造について述べた。

第4章 実験

4.1 まえがき

本章では, Capsule GAN について 2 種類のデータセットを用いた実験を行う. 実験に使用したデータセット, 実験結果及び考察について述べる.

4.2 データセット

本研究では, 0~9 の手書き数字の画像のデータセットである MNIST とカラーの動物画像データセットである猫画像を用いて実験を行った.

4.2.1 MNIST

MNIST とは, 0~9 までの手書き数字の画像データと, その数字のラベルデータがセットとなったデータセットである. 学習用に 60000 枚, 検証用に 10000 枚用意されている. 画像のサイズは 28×28 [pixel] であり, 白黒画像である. 本実験では, 学習用の画像 60000 枚を使用した.

4.2.2 猫画像

本実験では, 白黒画像である MNIST の他にカラー画像でも実験を行うため, 猫の画像を使用した. オックスフォード大学が公開している動物画像データセット, "The Oxford-IIIT Pet Dataset"[23] から猫の画像 4978 枚を使用した. また, 画像数を増やすためクローリングを行い, 総画像数を 7,836 枚とした. 画像のサイズは, 64×64 [pixel] に正規化して使用する.

4.3 提案手法に基づいた実験

3.2 節で述べた提案手法に基づいて実験を行った. 結果を 4.3.1 節, 4.3.2 節に示す.

4.3.1 MNIST を用いた結果

学習回数を 50000 回とし実験を行った. また, Discriminator (D) と Generator (G) の勾配更新回数比を, $D:G = 1:1, 3:1, 5:1$ とした. 実験で使用したパラメータを表 4.1, 結果を図 4.1 に示す.

表 4.1 パラメータ (MNIST)

パラメータ	値
学習回数 [回]	50000
勾配更新回数比 (D:G)	1:1, 3:1, 5:1
バッチサイズ	32
学習率	0.0002

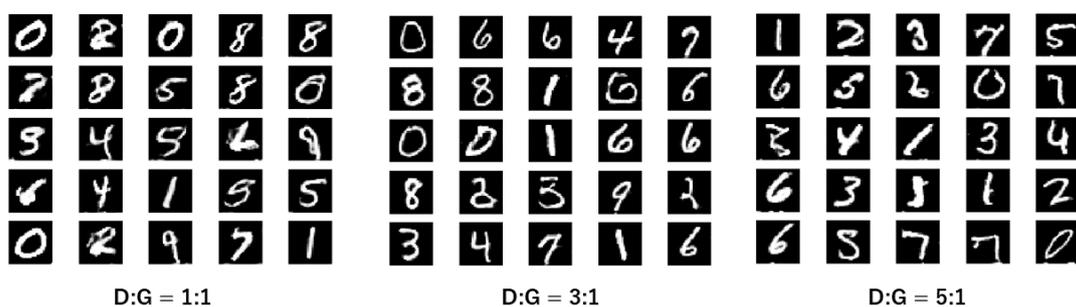


図 4.1 生成画像 (MNIST)

また、生成画像から IS, FID を計算し、評価を行った。画像は 10000 枚を使用した。結果を表 4.2 に示す。

表 4.2 評価結果

勾配更新回数比 (D:G)	IS	FID
1:1	2.50	32.68
3:1	2.68	15.25
5:1	2.35	8.40

4.3.2 猫画像を用いた結果

学習回数を 30000 回, D:G = 5:1 として実験を行った。学習回数を 30000 回としたのは、それ以上学習回数を行った結果、モード崩壊を起こしたためである。実験で使用したパラメータを表 4.3, 結果を図 4.2 に示す。

表 4.3 パラメータ (猫画像)

パラメータ	値
学習回数 [回]	30000
勾配更新回数比 (D:G)	5:1
バッチサイズ	32
学習率	0.0002



図 4.2 生成画像 (猫画像)

4.3.3 考察

4.3.3.1 MNIST を用いた結果の考察

4.3.1 節より考察する. 表 4.2 より, FID は Discriminator と Generator の勾配更新回数比を 5:1 としたものが, 3:1, 1:1 よりもそれぞれ 6.85, 24.28 の値の向上を確認できた. したがって, FID の結果では, Discriminator の勾配更新数を多くした方が品質の良い画像を生成することがわかる. 一方, IS の結果を比較すると勾配更新回数比が 3:1 としたものが, 1:1, 5:1 よりもそれぞれ 0.18, 0.33 値が大きいことがわかる. ここで, それぞれランダムに 100 枚ずつ生成画像を選び, 生成された数字を分類した. 結果を表 4.4, 図 4.3 に示す.

表 4.4 生成された数字の分布

勾配更新回数比	0	1	2	3	4	5	6	7	8	9	分散
1:1	19	21	6	11	6	9	10	3	10	5	34.4
3:1	11	37	1	3	4	9	7	14	5	9	105.3
5:1	8	9	3	10	16	7	17	11	9	10	16.7

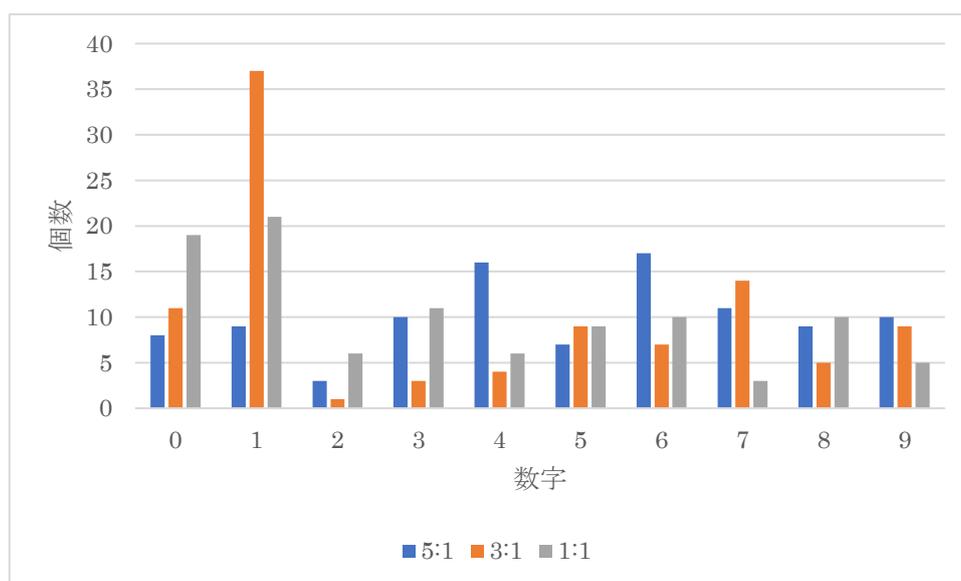


図 4.3 生成された数字の分布

表 4.4 より、分散から生成画像の数字別の数のばらつきが、勾配更新回数比が 3:1, 1:1, 5:1 の順で大きいことがわかる。したがって、5:1 のときが他の二つの場合と比べて生成される数字の種類に偏りがなく、様々な種類の数字を生成できていることがわかる。また、図 4.3 より勾配更新回数比が 3:1 のとき、1 を多く生成していることがわかる。他の数字よりも比較的識別が容易な 1 を多く生成しているため、Inception モデルも識別しやすく、IS が大きくなったと考えられる。また、IS に比べて FID は勾配更新回数比のそれぞれの値の差が大きく、結果に差がでている。したがって、今回の実験の結果からは勾配更新回数比が 5:1 のときが、一番生成画像の品質が良いと考えられる。

4.3.3.2 猫画像を用いた結果の考察

4.3.2 節より考察する。猫画像を用いた場合、生成画像の一部の品質が極端に落ち、モザイク画像のようにになっていることがわかる。今回の実験では、生成画像を 25 枚ずつ出力している。バッチサイズを 32 としているため、Discriminator の入力画像は、3.2 節より訓練

画像 16 枚，生成画像 16 枚の順番となっている．したがって，Generator に使用される画像の特徴を持っている DigitCaps 層の順番も，訓練画像の特徴 16 個，生成画像の特徴 16 個の順番である．生成画像の出力される順番は，図 4.4 のようになっている．また，複数回出力した結果を図 4.5 に示す．図 4.4，図 4.5 より比較すると，どの出力においても 17 番目付近から生成画像の品質が落ちていることがわかる．17 番目以降の画像は，生成画像の特徴をもった DigitCaps 層を Generator の入力に用いている．したがって，生成画像の特徴を用いた画像の品質が落ちていると考えられる．

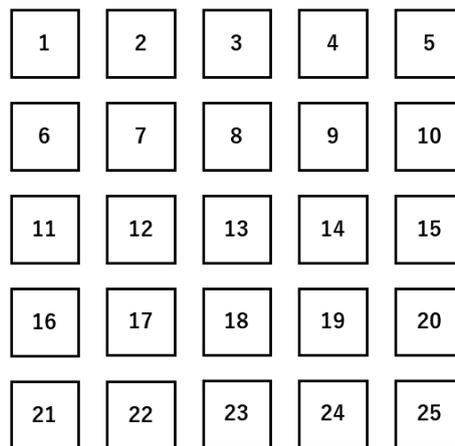


図 4.4 図の出力の順番

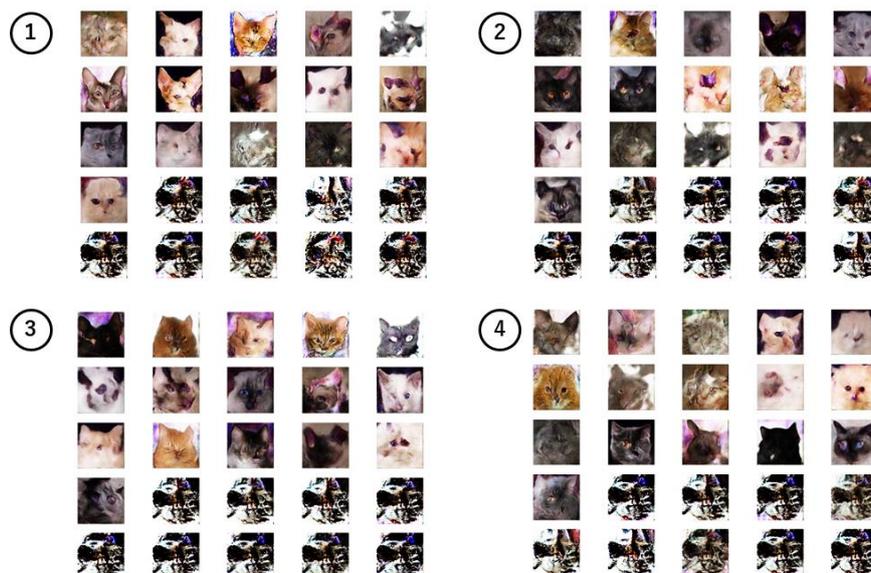


図 4.5 複数回出力した結果

4.3.3.3 MNIST と猫画像を用いた結果の比較

4.3.1 節, 4.3.2 節より, MNIST を用いた結果と猫画像を用いた結果を比較する. MNIST を用いた結果では, 生成画像の特徴を用いた出力画像に品質の低下はみられないことがわかる. MNIST を用いた場合と猫画像を用いた場合において結果に違いがでた理由は, Generator の学習速度によるものだと考えられる. GAN の学習では, Generator の学習より先に Discriminator の学習が収束する傾向がある[24]. 猫画像を用いた場合, Generator が上手く画像を生成することができないまま学習が進んでしまったと考えられる. そのため, 生成画像の特徴を用いた出力画像の品質が落ちたと考えられる. MNIST を用いた場合では, Generator も早い段階で学習が収束していったため, 生成画像からでも上手く特徴を捉えることができ, 出力画像の品質が一定であったと考えられる.

4.3.4 Capsule GAN の修正

4.3.3.2 節より, 生成画像の特徴をもった DigitCaps 層を使用すると学習が上手くいかないことがわかった. そのため, 3.2 節で述べた Capsule GAN の構造を修正する. 修正した Capsule GAN の構造を図 4.6 に示す.

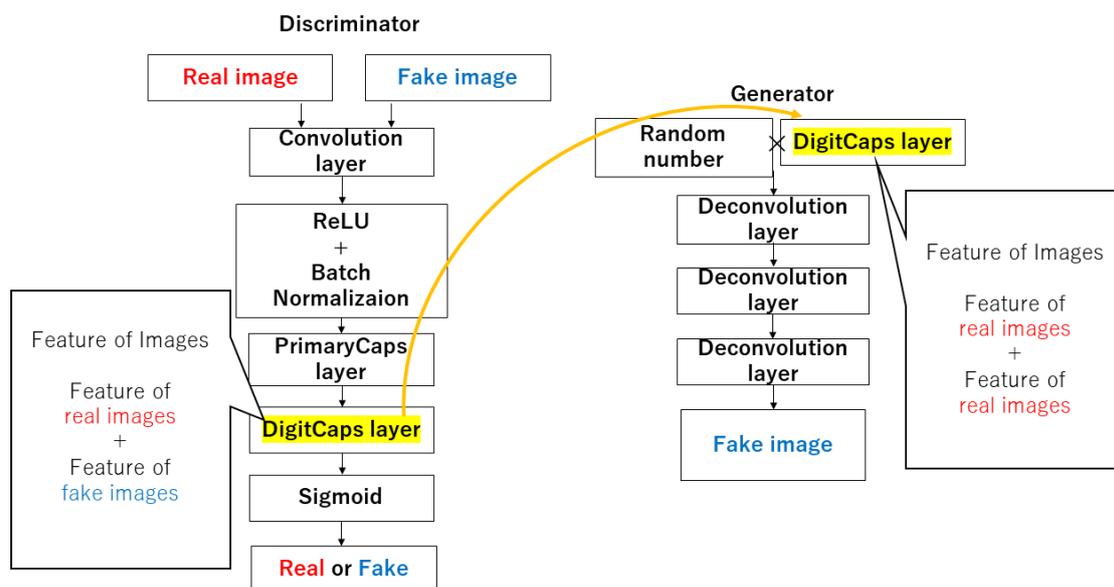


図 4.6 修正した Capsule GAN の構造

Generator に使用する DigitCaps 層を変更した. 生成画像の特徴を使用せず, 訓練画像の特徴のみを使用する構造とした.

4.4 提案手法を修正した実験

4.3.4 節で述べた，修正した Capsule GAN の構造を用いて実験を行った．結果を 4.4.1 節，4.4.2 節に示す．また，DCGAN との比較を行った．結果を 4.4.3 節に示す．

4.4.1 MNIST を用いた結果

表 4.1 と同様のパラメータで実験を行った．ただし，Discriminator と Generator の勾配更新回数比は $D:G = 5:1$ のみとした．結果を図 4.7 に示す．

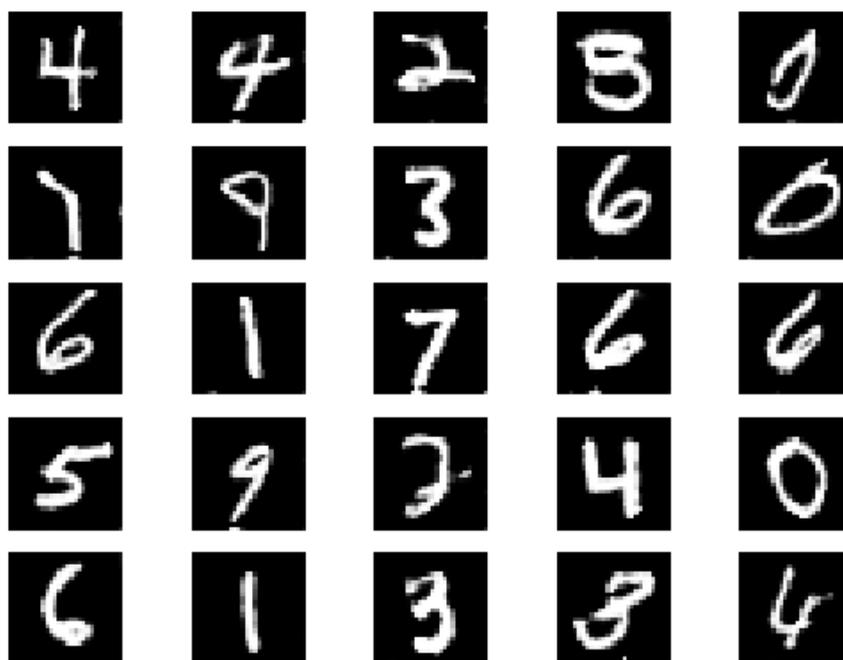


図 4.7 生成画像 (MNIST)

図より，一部の画像の外周が欠けていることが確認できる．図 4.8 に外周が欠けている画像と欠けていない画像の比較を示す．



図 4.8 外周が欠けている画像と欠けていない画像の比較

4.4.2 猫画像を用いた結果

表 4.3 と同様のパラメータで実験を行った。結果を図 4.9 に示す。

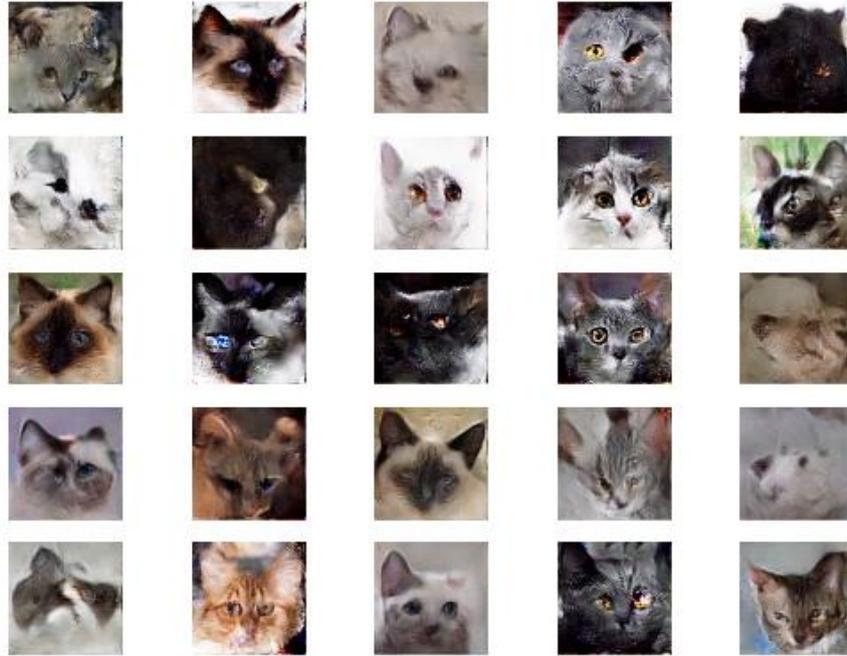


図 4.9 生成画像（猫画像）

また，生成画像を品質の良い画像，悪い画像に分けた結果を図 4.10 に示す。



図 4.10 品質の良い画像と悪い画像の比較

4.4.3 DCGAN との比較

MNIST を用いて DCGAN でも同様に実験を行い，Capsule GAN と比較した．DCGAN のパラメータは，Capsule GAN と同じ値を用いた．DCGAN による生成画像を図 4.11 に示す．

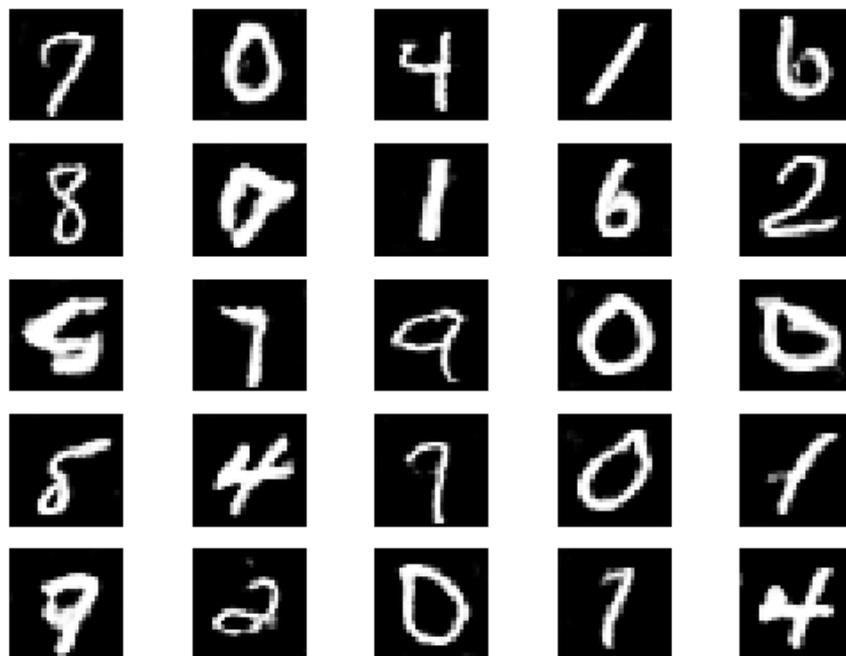


図 4.11 DCGAN による生成画像

また，Capsule GAN と DCGAN の生成画像を比較したものを図 4.12 に示す．

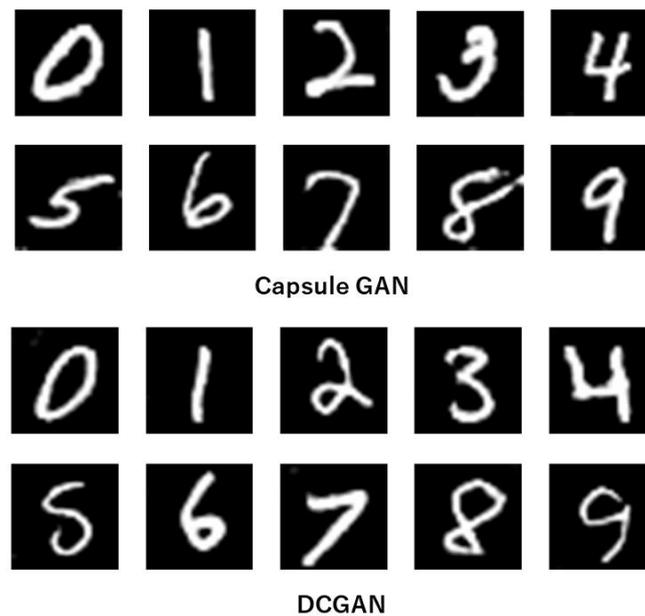


図 4.12 生成画像の比較

DCGAN, Capsule GAN により生成された画像から IS, FID を求めた. 画像は 10000 枚を使用した. 結果を表 4.5 に示す.

表 4.5 Capsule GAN と DCGAN の評価結果

モデル	IS	FID
Capsule GAN	2.37	14.14
DCGAN	2.32	7.65

図 4.8 より, Capsule GAN には外周が欠けている画像がある. 外周が欠けていない画像 100 枚と DCGAN による生成画像 100 枚から IS と FID を計算し, 評価を行った. 結果を表 4.6 に示す.

表 4.6 Capsule GAN と DCGAN の評価結果 (100 枚)

モデル	IS	FID
Capsule GAN	2.58	57.56
DCGAN	2.39	57.87

4.4.4 考察

4.4.4.1 MNIST を用いた結果の考察

4.4.1 節より考察する。図 4.8 より、外周の欠けている画像と欠けていない画像があることが確認できる。これは、画像を生成していく途中で値のオーバーフローが生じたと考えられる。また、表 4.2, 表 4.5 より Capsule GAN の構造の修正前, 修正後の IS と FID の値を比較すると、修正前の方が IS は 0.01, FID は 5.74 良い結果となった。MNIST を用いた場合、Generator は学習の速い段階である程度の品質の画像が生成できていた。また、修正前では、1 回の学習で Generator に使用する 32 個の DigitCaps 層はそれぞれ違う値であった。一方、修正後では 16 個の DigitCaps 層をそれぞれ 2 回ずつ使用し 32 個としていた。そのため、修正前の方が、修正後よりも学習の際に多くの種類のパラメータを使用できていたことがわかる。したがって、構造を修正する前の方が、IS と FID の評価値が良い結果となったと考えられる。

4.4.4.2 猫画像を用いた結果の考察

4.4.2 節より考察する。図 4.9 より、Capsule GAN の構造の修正前のようなモザイク画像が生成されていないことがわかる。また、図 4.10 より MNIST を用いた結果とは違い、生成画像の品質にばらつきがあることが確認できる。猫画像は MNIST に比べて複雑なため、GAN による学習が難しく、品質の悪い画像が生成されると考えられる。したがって、猫画像のような複雑な画像にも対応できるように Capsule GAN のパラメータ及び構造を修正する必要がある。

4.4.4.3 DCGAN との比較結果の考察

4.4.3 節より考察する。表 4.5 より、IS では Capsule GAN の方が DCGAN よりも 0.05 の値の向上を確認できた。一方、FID では DCGAN よりも 6.49 値が悪くなった。表 4.6 より、Capsule GAN の外周の欠けていない生成画像と DCGAN の生成画像を比較すると、FID においても Capsule GAN の方が DCGAN よりも 0.31 の値の向上を確認できた。このことから、生成画像の評価に生成画像の外周が欠けている点が影響しているといえる。また、表 4.6 の結果は画像の外周ではなく画像の特徴である数字を評価していると考えられる。そのため、Capsule GAN の方が DCGAN よりも画像の特徴を捉え、数字を品質良く生成できているといえる。

4.5 むすび

本章では、本研究で提案した手法に基づく実験の概要、結果及び考察について示した。

第5章 結論と今後の課題

5.1 結論

本研究では, Capsule Network を用いた GAN である Capsule GAN を提案し, 生成画像を評価した. GAN の Discriminator には, Capsule Network の構造を CNN のように組み込んだ. Generator には, Discriminator から取り出した Capsule Network の層である DigitCaps 層を入力に使用した. データセットには MNIST, 猫画像を使用し, それぞれの場合で画像を生成した. MNIST では, DCGAN と比較を行い IS, FID を求めた. IS では DCGAN よりも良い結果を出すことに成功した. FID では, 外周が欠けている画像を除いた場合では, DCGAN よりも良い結果となった. 猫画像を用いた場合では, 品質にばらつきはあるが, 猫画像を生成することに成功した.

5.2 今後の課題

本研究では, 生成画像の外周が欠ける問題がある. また, 猫画像のような複雑な画像の場合, MNIST のように生成画像の品質が一定ではない. したがって, Capsule GAN の構造及びパラメータを見直し, 様々な画像に対応できるように修正していく必要がある.

謝辞

本研究の実験器具の準備, 環境を整えてくださり, 研究テーマに対して熱心かつ丁寧に指導してくださった渡辺教授に深く感謝申し上げます.

また, 日頃から研究の問題点に対して相談に乗っていただき, アドバイスをくださった研究室の皆様に心より感謝申し上げます.

最後に, 私をここまで育ててくださり, 学費を工面してくださった家族に対して感謝致します.

参考文献

- [1] A. Krizhevsky, I. Sutskever and G. Hinton, “ImageNet Classification with Deep Convolutional Neural Networks”, Neural Information Processing Systems (NIPS), pp. 1106-1114, Dec. 2012.
- [2] Q. Le, M. Ranzato, R. Monga, M. Devin, G. Corrado, K. Chen, J. Dean and A. Ng, “Building High-level Features Using Large Scale Unsupervised Learning”, International Conference on Machine Learning (ICML), June 2012.
- [3] S. Iizuka, E. Simo-Serra and H. Ishikawa, “Let there be Color! : Joint End-to end Learning of Global and Local Image Priors for Automatic Image Colorization with Simultaneous Classification”, ACM Transactions on Graphics (TOG), July 2016.
- [4] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville and Y. Bengio, “Generative Adversarial Networks”, Neural Information Processing Systems (NIPS), Dec. 2014.
- [5] A. Radford, L. Metz and S. Chintala, “Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks”, International Conference on Learning Representations (ICLR), Jan. 2016.
- [6] S. Sabour, N. Frosst and G. E. Hinton, “Dynamic Routing Between Capsules”, Neural Information Processing Systems (NIPS), Dec. 2017.
- [7] H. Gadirov, M. Tamošiūnaitė and D. Vitkute-Adzgauskiene, “Capsule Architecture as a Discriminator in Generative Adversarial Networks”, Vytautas Magnus University, Feb. 2018, M. D. thesis.
- [8] Y. LeCun, C. Cortes and C. J. C. Burges, “The MNIST Database of Handwritten Digits”, <http://yann.lecun.com/exdb/mnist/>, 1998.
- [9] A. Krizhevsky, “Convolutional deep belief networks on CIFAR-10”, Aug. 2010.
- [10] 武井宏将, 初めてのディープラーニング—オープンソース “Caffe” による演習付き—, リックテレコム, p.19, 2016.
- [11] 斎藤康毅, ゼロから作る Deep Learning—Python で学ぶディープラーニングの理論と実装—, オライリー・ジャパン, p.205, 2016.
- [12] 株式会社 Spot, “定番の Convolutional Neural Network をゼロから理解する”, https://deepage.net/deep_learning/2016/11/07/convolutional_neural_network.html (2019年1月現在) .
- [13] 株式会社システムインテグレータ, “畳み込みニューラルネットワーク_CNN (Vol.16)”, <https://products.sint.co.jp/aisia/blog/vol1-16> (2019年1月現在) .

- [14] 清水亮, “深層学習を根底から覆すカプセルネットワークの衝撃”, <https://wirelesswire.jp/2018/03/64233/> (2019年1月現在) .
- [15] S. Sabour, N. Frosst and G. E. Hinton, “Dynamic Routing Between Capsules”, *Neural Information Processing Systems (NIPS)*, pp.2-4, Dec. 2017.
- [16] 株式会社システムインテグレータ, “敵対的生成ネットワーク_GAN (Vol.19) ”, <https://products.sint.co.jp/aisia/blog/vol1-19> (2019年1月現在) .
- [17] Y. Shinya, “はじめての GAN”, <https://elix-tech.github.io/ja/2017/02/06/gan.html> (2019年1月現在) .
- [18] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford and X. Chen, “Improved Techniques for Training GANs”, *Neural Information Processing Systems (NIPS)*, Dec. 2016.
- [19] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke and A. Rabinovich, “Going deeper with convolutions”, *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.
- [20] “Inception score”, <http://bluewidz.blogspot.com/2017/12/inception-score.html> <https://elix-tech.github.io/ja/2017/02/06/gan.html> (2019年1月現在) .
- [21] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler and S. Hochreiter, “GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium”, *Neural Information Processing Systems (NIPS)*, Dec. 2017.
- [22] “Fréchet Inception Distance”, <http://bluewidz.blogspot.com/2017/12/frechet-inception-distance.html> (2019年1月現在) .
- [23] O. M. Parkhi, A. Vedaldi, A. Zisserman and C. V. Jawahar, “Cats and Dogs”, *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2012.
- [24] “Generating Faces with Torch”, <http://torch.ch/blog/2015/11/13/gan.html> (2019年1月現在) .

図一覧

図 2.1	CNN の基本構造	4
図 2.2	畳み込みの概略図.....	4
図 2.3	プーリングの概略図	5
図 2.4	CNN の誤認識	6
図 2.5	Capsule Network の構造	7
図 2.6	GAN の基本構造	7
図 3.1	Capsule GAN の構造	10
図 4.1	生成画像 (MNIST)	13
図 4.2	生成画像 (猫画像).....	14
図 4.3	生成された数字の分布.....	15
図 4.4	図の出力の順番	16
図 4.5	複数回出力した結果	16
図 4.6	修正した Capsule GAN の構造	17
図 4.7	生成画像 (MNIST)	18
図 4.8	外周が欠けている画像と欠けていない画像の比較.....	18
図 4.9	生成画像 (猫画像).....	19
図 4.10	品質の良い画像と悪い画像の比較	19
図 4.11	DCGAN による生成画像	20
図 4.12	生成画像の比較	21

表一覧

表 4.1	パラメータ (MNIST).....	13
表 4.2	評価結果.....	13
表 4.3	パラメータ (猫画像).....	14
表 4.4	生成された数字の分布.....	15
表 4.5	Capsule GAN と DCGAN の評価結果.....	21
表 4.6	Capsule GAN と DCGAN の評価結果 (100 枚).....	21

研究業績

- [1] 松田, 丸寄, 渡辺, ”生成画像品質を考慮した CapsGAN によるデータ拡張”, 映像情報メディア学会冬季大会, 12D-4, Dec. 2018.
- [2] 丸寄, 渡辺, ”A Study on GAN Using Capsule Network”, 電子情報通信学会総合大会, Mar. 2019 (発表予定) .