2022 年度
中部大学大学院工学研究科ロボット理工学専攻

博士学位論文

敵対的方策による **mixup** を用いた
データ増幅と学習法に関する研究

足立 浩規

論　文　要　旨

Deep Convolutional Neural Networks (CNN) は 2012 年以降，目覚ましい発展によって画像分類や認識だけでなく，さまざまなタスクにおいて人間よりも優れた性能を発揮している．CNN は運転支援システムや医療診断システムなどに活用することで，我々の生活を豊かにできる可能性を秘めているが，CNN は致命的な問題点を 2 つ抱えている．

まず，CNN によって優れた分類を実現するためには大規模な学習データが必要となることが，1 つ目の問題点である．データに偏りがある場合や限られたデータ数では，CNN は学習データに過適合して著しく性能が劣化する．この問題点の直感的な解決策は人手による大規模データセットの作成であるが，多方面で高コストであるため既存のデータに幾何変化等を施すデータ増幅が一般に使用される．しかし，従来のデータ増幅は物体の見え方が異なるものの，画像中に映る物体は一貫して同じである．

そこで本研究では，画像生成モデルから得た生成画像を用いてデータ増幅を行う．画像生成モデルは学習データセットを内挿するような画像が得られるため，画像中に映る物体のバリエーションを増強できる．まず，顔画像認識に着目して，重み付けした顔属性を用いて顔画像生成する．提案手法は高精細な顔画像が得られるため，顔画像認識精度を向上させることができる．次に，CNN が特徴的な領域に強く注視して識別することに着目して，識別対象領域に絞って生成した画像を増幅データとして活用する．提案手法によって生成した画像は，学習データが限られているときに増幅データとして使用することで飛躍的に性能を向上させることができる．

2 つ目の問題点は，悪意のある摂動を付与した画像 (adversarial examples) によって，CNN の分類結果がいとも簡単に変えられてしまうことである．この摂動は人間にとって知覚困難なほど微小な変化であり，最先端な分類器であっても生じる致命的な問題であるため，CNN をベースとしたアプリケーションのセキュリティの脅威となる．Adversarial Training (AT) は優れた頑健性を獲得できる最も有名な方法である一方，ロバスト過適合や通常のサンプルに対する分類精度を劣化させることが問題視されている．

この問題に対処する手法として，まず本研究では学習する adversarial examples のバリエーションを増幅させる方法を提案する．AT において，通常の分類精度を保ちつつ，優れた頑健性を得るためには一般的な学習よりも大規模で複雑なデータが必要であることが理論的に証明されている．そこで，提案手法はデータ増幅手法と組み合わせて adversarial examples を作成することで多様な adversarial examples を学習する．提案手法は通常の分類精度を維持しつつ，著しい頑健性の向上ができる．次に，多クラス分類に適した Instance-Reweighted Adversarial Training (IRAT) を提案する．従来の IRAT は，正解クラス確率と最も迷ったクラス確率をベースに攻撃リスクが高いサンプルを特定して，損失へ重み付けする．しかし，この計算方法は多クラス分類にも関わらず，暗黙的に 2 クラス分類が想定されている．そこで，本研究では従来法の弱点を証明し，従来法で計算された重要度を適切に変換できる新たな指標を提案する．提案手法は，いくつかの攻撃に対する頑健性を向上させることができる．

さらに，データ増幅の一種である mixup と敵対的方策を組み合わせた新たな学習法について取り

組む．mixup は 2 つのデータを任意の内挿比で合成して学習する強力なデータ増幅である．mixup は様々な派生手法が提案されているが，その多くがデータの合成方法に着目されており，内挿比に着目した研究が圧倒的に少ない．直感的に，CNN が最も苦手なデータを学習することで優れた分類性能が獲得できるが，学習に有効な内挿比はデータペアごとや学習の進捗と共に異なると考えられる．そこで本研究では，損失が最大となる内挿比で合成したデータを CNN に学習させる新たなデータ増幅と学習法を提案する．提案手法は，画像分類のあらゆるベンチマークデータセットとベースモデルにおいて最高性能が実現できる．

目 次

第 1 章　序論 **1**

1.1　研究の背景 2

1.2　研究目的 2

1.3　本論文の構成 4

第 2 章　画像処理分野における敵対的深層学習の関連研究 **6**

2.1　Generative Adversarial Networks: 画像生成モデルとしての敵対的方策 7

2.1.1　安定した学習のための GAN 9

2.1.2　高精細な画像生成をする GAN 10

2.1.3　意図した画像生成が可能な GAN 13

2.1.4　生成画像の評価指標 14

2.2　モデルの頑健性のための敵対的方策 16

2.2.1　代表的な敵対的攻撃 17

2.2.2　Adversarial Training 18

2.2.3　Adversarial Training の派生手法 19

2.3　まとめ 29

第 3 章　重み付き条件を入力とした conditional GAN による顔画像生成とデータ増幅 **30**

3.1　関連研究 32

3.2　提案手法 32

3.2.1　問題設定 33

3.2.2　Weighted conditional layer の導入 33

3.2.3　マルチタスク Discriminator 35

3.3　評価実験 36

3.3.1　実験概要 36

3.3.2　生成画像の主観評価 37

3.3.3　実験結果 37

3.3.4　Ablation study 38

3.3.5　生成画像を用いた顔属性認識 40

3.3.6　考察 42

3.4 まとめ 43

第 4 章 注視領域を考慮した GAN による画像生成 45

4.1 関連研究 46
4.1.1 データ増幅 46
4.1.2 視覚的説明 46
4.2 提案手法 47
4.2.1 問題設定 47
4.2.2 注視領域を考慮した Discriminator 48
4.3 評価実験 51
4.3.1 実験の詳細 52
4.3.2 生成画像の画質評価 52
4.3.3 先行研究との識別精度の比較 54
4.3.4 Ablation study 55
4.3.5 Soft label に関する考察 56
4.4 まとめ 57

第 5 章 敵対的サンプルの多様性を増強した敵対的学習 58

5.1 予測分布の分析 60
5.2 提案手法 61
5.2.1 Overview 61
5.2.2 Masking phase 62
5.2.3 Mixing phase 64
5.3 評価実験 64
5.3.1 実験の詳細な設定 65
5.3.2 精度比較結果 65
5.3.3 Ablation study 67
5.3.4 Additional discussion 68
5.4 まとめ 69

第 6 章 多クラス間のマージンを考慮した敵対的学習 70

6.1 従来の IRAT における弱点 72
6.2 提案手法：Margin Reweighting 74
6.2.1 マージンに対する重要度の定量化 74
6.2.2 不正解率に対する重要度表現 76
6.2.3 従来の IRAT への提案手法の導入 77
6.3 Experiments 79
6.3.1 Experimental details 79

6.3.2 従来法との比較 80
6.3.3 その他の IRAT との比較 82
6.3.4 Ablation study 83
6.4 Discussion and Limitations 85
6.5 まとめ 85

第 7 章 敵対的方策による mixup を用いたデータ増幅と学習法 **87**
7.1 予備知識と関連研究 89
7.1.1 データ増幅 89
7.2 Adversarial Interpolating Policy 90
7.2.1 提案手法の概要 90
7.2.2 三分探索を用いた極大値探索 91
7.2.3 敵対的方策を適用する位置 93
7.3 評価実験 93
7.3.1 実験の詳細な設定 93
7.3.2 精度比較結果 95
7.3.3 内挿比探索回数の違いによる分類精度 96
7.3.4 入力層における提案手法の効果と誤差曲面の可視化 97
7.4 Discussion and Limitations 97
7.5 まとめ 98

第 8 章 結論と展望 **100**
8.1 結論 100
8.2 展望 102

謝 辞 **103**

参考文献 **105**

研究業績一覧 **120**

図 目 次

1.1 本論文の構成. 5

2.1 GAN のネットワーク構造. 7
2.2 MNIST データセットを用いたモード崩壊の例. 9
2.3 GAN の派生手法と評価指標の推移. 10
2.4 Self-attention 機構の処理. 11
2.5 PGGAN の学習過程. 12
2.6 StyleGAN によって生成した画像に生じる問題.(文献 [1] より引用.) 12
2.7 各手法の条件入力方法の違い. 13
2.8 adversarial examples の例.(文献 [2] より引用.) 16
2.9 Adversarial Training の流れ. 18
2.10 Adversarial Training の遷移. 19
2.11 AVmixup の摂動作成と，教師信号作成の例. 20
2.12 TLA training と AGKD-BML の triplet loss の違い. 22
2.13 異なる地点でパラメータ更新した時のマージンの関係.(文献 [3] より引用.) 23
2.14 損失関数と特徴空間の関係. 24
2.15 ロバスト過適合と誤差曲面の関係. 25
2.16 GAIRAT と WMMR/MAIL の重要度定義の違い．図形の大きさは重要度の高さを表している. 27

3.1 提案手法を導入した Generator の構造. 33
3.2 提案手法の Discriminator の構造. 34
3.3 各手法の顔画像生成例 [128 × 128 pixels]. 38
3.4 WcPGGAN で 192 × 256 [pixels] の顔画像生成例. 39
3.5 WcDCGAN における層ごとの顔属性の寄与率. 42
3.6 WcPGGAN における顔属性の寄与率と中間生成画像. 43

4.1 学習時の提案手法の損失の推移 47
4.2 提案手法の構造. 48
4.3 ABN と提案手法の Attention branch の設計の違い. 49

4.4 実画像および各手法の生成画像の情報量エントロピー． 51
4.5 各手法の生成画像と提案手法によって獲得した注視領域． 53
4.6 各手法で生成した画像を用いて学習したモデルの識別精度比較．(a) および (b) 共に左が CIFAR-10，右が SVHN の認識精度を表している．点線はデータ増幅を用いずに学習したモデルの認識精度を表している． 54
4.7 D2A2GAN と ACGAN，それぞれの生成画像に関する Discriminator の事後分布． . . 56

5.1 通常のサンプル，FGSM と PGD で求めた adversarial examples，それぞれをモデルに入力して得られる予測分布と予測結果． 60
5.2 M^2AT を用いた adversarial examples の作成のコンセプト図． 62
5.3 通常サンプルと adversarial examples に対する分類精度の推移． 67
5.4 各摂動許容範囲 ϵ を用いた adverarial examples に対する精度の推移．FGSM は全てのグラフで同じ結果である． 68

6.1 従来法と提案手法を組み込んだ表現の違いを表したコンセプト図．図形の大きさは重みの大きさ，実線と破線はそれぞれパラメータ更新前後の識別境界を表している．また，図形の種類は異なるクラスを表している． 71
6.2 CIFAR-10 を用いて通常の Adversarial Training した ResNet-18 おける，正解クラス確率 $p_{y_i}(\boldsymbol{x}_i)$ と式 (2.37) によって計算されたマージンの相関．オレンジとブルーの点は，それぞれ正解サンプルと不正解サンプルを表している． 72
6.3 (a) 低い信頼度および (b) 高い信頼度のサンプル，それぞれの予測分布に関して式 (6.11) で計算したコサイン類似度計算の関係．赤色は各サンプルにおける正解クラスを表している． 76
6.4 CIFAR-10 における，信頼度と式 (6.11) によって求めた類似度の関係． 77

7.1 通常の mixup と提案手法の内挿比のサンプリング方法の違い． 88
7.2 三分探索を用いた内挿比探索． 92
7.3 CIFAR-10 を学習した PreActResNet18 の誤差曲面． 97

表 目 次

3.1 定量的画質評価結果. 39
3.2 異なる条件入力方法の定量的評価. 40
3.3 Recognition branch の有無の定量的評価. 40
3.4 顔属性認識の結果（%）．w/o AL は active learning 無し，w/ AL は active learning 有りを指している．太字は，Baseline から精度が向上したことを表している. 41

4.1 Inception score と FID の比較. 52
4.2 Ablation study. 55

5.1 通常サンプルに対する分類精度と敵対的攻撃に対する頑健性の比較．太字は最高性能の手法，$*$ は論文値の精度を引用していることを表している. 66
5.2 CIFAR-10 における TRADES との性能比較．太字は最高性能を表している. 67
5.3 提案手法の Ablation study．太字は最高性能を表している. 67
5.4 PGD-20 を用いた Transfer ベースのブラックボックスアタックの結果．横の手法は攻撃モデル，縦の手法は防御モデルを表している. 69

6.1 CIFAR-10 における各手法の Last model の性能 (%). 80
6.2 CIFAR-100 における各手法の Last model の性能 (%). 81
6.3 CIFAR-10 における各手法の Best model の性能 (%). 82
6.4 CIFAR-100 における各手法の Best model の性能 (%). 83
6.5 CIFAR-10 における，GAIRAT と EWAT との性能比較 (%). 83
6.6 CIFAR-100 における，GAIRAT と EWAT との性能比較 (%). 84
6.7 ハイパーパラメータ τ に関する Ablation study. 84
6.8 式 (6.11) の類似度計算に関する Ablation study. 85

7.1 PreActResNet18 における分類性能比較．† は論文から引用した値である. 95
7.2 PreActResNet34 と WideResNet28-10 における分類性能比較．† は論文から引用した値である. 96
7.3 各データセットにおける内挿比探索回数の違いによる分類精度. 96
7.4 入力層における提案手法の有無による分類精度比較. 97
7.5 従来法と提案手法の処理速度．($K = 8$) 98

第1章

序論

本章では，本研究の背景及び目的，本論文の構成について述べる．

1.1 研究の背景

Deep Convolutional Neural Networks (CNN) は Krizhevsky ら [4] 以降，飛躍的な成長を遂げ，人間に匹敵するほど高精度な画像分類を実現している [5]．CNN は畳み込み処理を積層してネットワークが構築されており，学習用データセット全体の分類誤差を最小化するように重みパラメータを更新する．これによって，CNN はこれまで研究者の経験則で設計した特徴量とは比べ物にならないほど柔軟な特徴量を捉えられるため，画像分類以外の様々なタスクにおいても活躍している [6, 7, 8, 1]．2021 年には，CNN の性能を凌駕する，もはや畳み込み処理を必要としない Vision Transformer (ViT) [9] が脚光を浴びている．コンピュータビジョン分野における CNN の発達は，我々人類の生活を豊かにする上で重要な役割を担っている．例えば，運転支援システムでは，車載カメラから道路標識や歩行者を上手く認識することで，ドライバーのアシストや交通事故の防止するための技術として活用ができる．また，様々な病理画像から人間の医師では見落としてしまうほどの疾患を発見することで，医師のアシスタントとしての活用ができる．このように，CNN は我々の生活に関わる多岐に渡る分野において活躍が期待されており，活発に研究や開発が進められている．

様々な期待が込められて日々発展している CNN であるが，実は致命的な問題を 2 つ抱えている．1 つ目は，高精度な画像分類を実現するために膨大な学習用データセットが必要となる点である．CNN は学習データから上手く特徴量を学習することで高性能な推論が可能となる一方，データに偏りがある場合や学習データ自体が極端に少ないとき，著しい性能劣化を招く原因となる．この性能劣化を防ぐために，人手による学習に有効なデータの慎重な選定と教師信号のアノテーションが重要となるが，人的コストが非常に高いことが問題となる．そのため，既存のデータを有効活用して，学習用データを水増しすることが必須となる．

2 つ目は，Adversarial Examples によって容易に誤分類の誘発が可能な点である．これまで述べたように CNN は優れた画像分類を実現している反面，Adversarial Examples と呼ばれる悪意のある摂動を付与した画像に脆弱であることが知られている [10, 2]．この画像に対する摂動は人間には知覚困難であることから，CNN をベースとするアプリケーション (例えば，自動運転車両や医療診断システム) のセキュリティの脅威となる可能性が高い．Adversarial Examples による攻撃を緩和するための技術として，敵対的学習 (Adversarial Training) [11] が広く用いられている．Adversarial Training は通常の学習用データを学習する代わりに，それらの Adversarial Examples を学習することで攻撃に頑健なモデルを獲得することができる．しかしながら，Adversarial Training を適用したモデルは，摂動のないデータに対する分類性能が著しく劣化するだけでなく，ロバスト過適合 (robust overfitting) を招くと言われている．従って，これらの問題を緩和しつつ頑健なモデルを獲得可能な学習方法が必要であるため，様々なアプローチが提案されている．

1.2 研究目的

本研究では，大きく分けて以下に示す 3 つの項目について取り組む．

1. 生成画像を用いた学習データの増幅.
2. 敵対的学習を適用したモデルの頑健性向上と通常サンプルの分類精度の維持.
3. 敵対的方策と mixup を組み合わせた新たなデータ増幅と学習法.

1 つ目は，顔属性認識と一般物体認識を対象として，画像生成モデルによって生成した画像で物理的な画像の水増しを行い，認識精度向上を目指す．2 つ目は，学習する敵対サンプルのバリエーションを意図的に増幅させることによって，通常の分類精度を保ちつつ敵対的攻撃に対する頑健性向上を目指す．また，従来の Instance-Reweighted Adversarial Training が多クラス分類において不十分な表現であることを明らかにして，更なる頑健性向上を目指す．3 つ目は，mixup において最も損失が高くなる内挿比を意図的に求めて，損失を最小化することで従来法よりも高精度な画像分類を目指す．以下に各項目に関して詳細に述べる．

生成画像を用いた学習データの増幅

学習用データセットを作成するための方法として，人手によるデータ収集とそれらのデータに対する教師信号のアノテーションが考えられるが，明らかに人的コストが高い．従って，CNN を学習する際には既存データに幾何変化を施して，仮想的にデータを水増しするデータ増幅 (data augmentation) が広く活用されている．しかし，データ増幅によって変化が加わった画像は，元の画像と見え方が異なるものの，画像中に移る物体は一貫して同じである．本研究では，敵対的学習を用いた画像生成モデルである Generative Adversarial Networks (GAN) [12] を用いたデータ増幅を目指す．その中でも，意図した画像生成が可能な conditional GAN [13] に着目して，重み付き条件を用いた顔画像生成のための cGAN と注視領域を考慮した cGAN の 2 つのアプローチを提案する．重み付き条件を用いた顔画像生成のための cGAN で生成した顔画像は，定量的評価において優れているだけでなく学習データとしても有効である．注視領域を考慮した cGAN で生成した画像は，定量的な画質が低いにも関わらず，極端に学習用データが少ない場合に著しい性能向上を実現することができる．

敵対的学習を行ったモデルの頑健性を向上させつつ，性能劣化を予防

Adversarial Training は攻撃に対する頑健性を高める反面，摂動なし画像である通常データに対する分類精度を著しく劣化させることが知られている．この問題を解消するためには，通常の学習よりも膨大で複雑なデータを学習させる必要があることが理論的に示されている [14]．Adversarial Vertex mixup (AVmixup) [15] は，仮想的に定義した Adversarial Examples と摂動なし画像を mixup [16] しながら学習することで，通常データの分類性能を維持しつつ飛躍的な頑健性向上を実現した．しかし，AVmixup で学習される Adversarial Examples は，まだ限定的であると捉えて，本研究では更なる，バリエーション豊富な Adversarial Exmaples を学習することができる Masking and Mixing Adversarial Training を提案する．提案手法は，AVmixup より著しい頑健性向上を実現しつつ，通常データの性能を維持することができる．さらに，提案手法は強い摂動に対して AVmixup よりも優れた頑健性を実現できる．

多クラス分類に適した Instance-Reweighted Adversarial Training の実現

Instance-Reweighted Adversarial Training (IRAT) は，各サンプルに対する攻撃されやすさを識別境界とのマージンとして定量化し，攻撃されやすいサンプルほど損失に大きな重み付けして学習する Adversarial Training である．特に，クラス確率に基づいて定量化する手法は，著しい頑健性向上を実現している．しかし，これらの手法は，正解クラス確率と最も迷ったクラス確率のみからマージンを算出するため，暗黙的に 2 クラス分類が想定されている．従って，求めたマージンは多クラス分類において不十分な表現である可能性が高い．本研究では，まず，従来のマージンが多クラス分類を想定した場合，表現が不十分であることを明らかにする．具体的には，異なる確率を持ついくつかのサンプルから同じマージンを求められる事例を示す．そして，従来のマージンを多クラス分類に適した表現に変換するための手法を提案する．提案手法を従来法に組み込み，適切な表現に変換して学習することで，いくつかの敵対的攻撃に対する頑健性の向上だけでなく，摂動なし画像の分類精度も僅かに向上させることができる．

敵対的方策による mixup を用いたデータ増幅と学習法

mixup [16] は 2 つのデータを任意の確率で合成して新たなデータを作成する強力なデータ増幅である．mixup は数多くの派生手法が提案されているが，その多くがデータの合成方法に着目しており，内挿比に対するアプローチは非常に少ない．直感的に，損失が最大となる内挿比を用いて合成したデータを正確に分類できるように鍛えたモデルは，推論データに対する分類精度が向上すると考えられる．本研究では，敵対的方策を利用して内挿比を求め，それらのデータに対する損失を最小化するように学習する手法を提案する．内挿比は各データに対して $[0, 1]$ のスカラー値で定義するため，敵対的攻撃のように求めるのではなく，古典的な最小値探索を逆向きに使用して損失が最大となる内挿比を求める．そのため，Adversarial Training よりも少ない計算コストで内挿比を求めることが可能となる．提案手法はあらゆるデータセットとベースモデルにおいて最高性能を達成することができる．

1.3 本論文の構成

本論文は図 1.1 に示すように，8 つの章で構成されている．1 章では，本研究の背景と目的を具体的に述べた．2 章では，画像生成モデルと画像分類モデルの 2 つの観点で敵対的方策を詳しく述べた後，それぞれの関連研究をまとめる．3 章では，画像生成モデルに与える顔属性に重み付けすることで，高品質な顔画像生成が可能な手法を提案する．4 章では，識別時に注目する領域を画像生成モデルに組み込むことで，学習用データセットとして有効な画像生成ができる手法を提案する．5 章と 6 章では，CNN の敵対的攻撃に対する頑健性を向上させるための手法をそれぞれ提案する．まず，5 章では，一般的なデータ増幅方法を，多様な Adversarial Examples を学習するために有効活用した Masking-Mixing Adversarial Training を提案して，6 章で従来の Instance-Reweighted Adversarial Training の弱点を理論的および経験的な実験により明らかにした後に，その弱点を解消するための Margin Reweighting を提案する．7 章では，敵対的方策を使用した新たな mixup を提案する．最後に，8 章で本研究の全体を総括し，今後の展望を述べる．

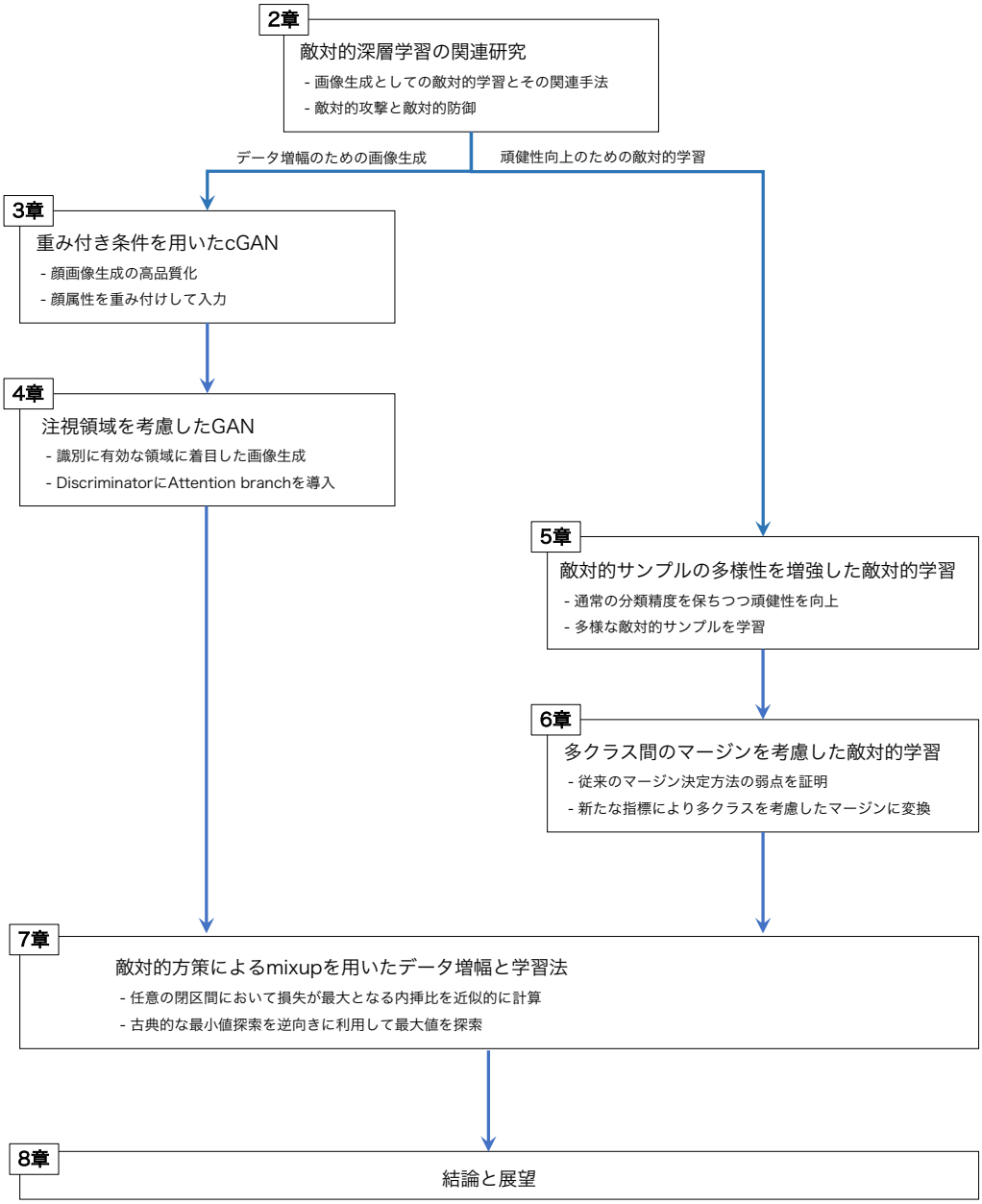

2章
敵対的深層学習の関連研究
- 画像生成としての敵対的学習とその関連手法
- 敵対的攻撃と敵対的防御
データ増幅のための画像生成
頑健性向上のための敵対的学習
3章
重み付き条件を用いたcGAN
- 顔画像生成の高品質化
- 顔属性を重み付けして入力
4章
注視領域を考慮したGAN
- 識別に有効な領域に着目した画像生成
- DiscriminatorにAttention branchを導入
5章
敵対的サンプルの多様性を増強した敵対的学習
- 通常の分類精度を保ちつつ頑健性を向上
- 多様な敵対的サンプルを学習
6章
多クラス間のマージンを考慮した敵対的学習
- 従来のマージン決定方法の弱点を証明
- 新たな指標により多クラスを考慮したマージンに変換
7章
敵対的方策によるmixupを用いたデータ増幅と学習法
- 任意の閉区間において損失が最大となる内挿比を近似的に計算
- 古典的な最小値探索を逆向きに利用して最大値を探索
8章
結論と展望

図 1.1: 本論文の構成.

第2章

画像処理分野における敵対的深層学習の関連研究

画像処理分野において，敵対的方策は画像生成モデルとモデルの頑健性向上を目的として利用されている．敵対的方策を用いた画像生成モデルは Generative Adversarial Networks (GAN) [12] と呼ばれる．一方，モデルの頑健性向上を目的とした敵対的方策は，敵対的学習 (Adversarial Training) と呼ばれている．

本章では，まず 3 章と 4 章に関連する GAN について概説して，GAN の派生手法について調査してまとめる．次に，5 章と 6 章に関連する敵対的攻撃と敵対的防御，特に Adversarial Training の関連研究を調査してまとめる．

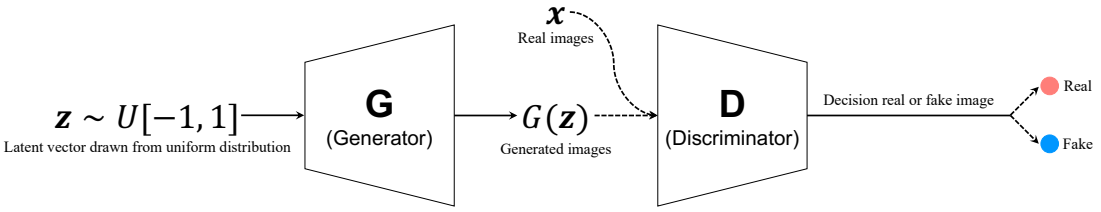


図 2.1: GAN のネットワーク構造.

2.1 Generative Adversarial Networks: 画像生成モデルとしての敵対的方策

敵対的生成ネットワーク (Generative Adversarial Networks: GAN) [12] は図 2.1 に示すように，画像を生成する Generator と，入力された画像が実画像か否かを判定する Discriminator の 2 つのネットワークからなる画像生成モデルである．Generator は $U[-1, 1]$ または $N(0, 1)$ のどちらかの事前分布 $p_z(\boldsymbol{z})$ からサンプリングした潜在ベクトル $\boldsymbol{z} \in \mathbb{R}^d$ を入力として画像を生成する．Discriminator は生成画像 $G(\boldsymbol{x})$ または実画像 $\boldsymbol{x}$ が入力されて，どちらが入力されたか判定する．Generator は Discriminator を惑わすような画像生成が目的である一方，Discriminator は入力画像を正確に判定することが目的である．このように，2 つのネットワークを敵対させながら学習することで，Variational Autoencoder (VAE) [17] のような関数近似なしで画像生成が可能となる．

GAN の目的関数は，Discriminator を $D(\cdot)$，Generator を $G(\cdot)$ とすると以下の式で表すことができる．

$$V(D, G) = \mathbb{E}_{\boldsymbol{x} \sim p_{data}(\boldsymbol{x})}[\log D(\boldsymbol{x})] + \mathbb{E}_{\boldsymbol{z} \sim p_z(\boldsymbol{z})}[\log (1 - D(G(\boldsymbol{z})))] \tag{2.1}$$

ここで，$p_{data}(\boldsymbol{x})$ は訓練データが従う真の分布である．式 (2.1) の目的関数は連続値を扱う期待値計算であるため，以下のように式変形することが可能である．

$$V(D, G) = \int_{\boldsymbol{x}} p_{data}(\boldsymbol{x}) \log D(\boldsymbol{x}) d\boldsymbol{x} + \int_{\boldsymbol{z}} p_z(\boldsymbol{z}) \log (1 - D(G(\boldsymbol{z}))) d\boldsymbol{z} \tag{2.2}$$

さらに，生成データ $G(\boldsymbol{z})$ を訓練データの一部として考えた際，無意識の統計学者の法則 (law of the unconscious statistician; LoTUS) を用いて，

$$\begin{aligned} V(D, G) &= \int_{\boldsymbol{x}} p_{data}(\boldsymbol{x}) \log D(\boldsymbol{x}) d\boldsymbol{x} + \int_{\boldsymbol{x}} p_g(\boldsymbol{x}) \log (1 - D(G(\boldsymbol{z}))) d\boldsymbol{x} \\ &= \int_{\boldsymbol{x}} p_{data}(\boldsymbol{x}) \log D(\boldsymbol{x}) + p_g(\boldsymbol{x}) \log (1 - D(G(\boldsymbol{z}))) d\boldsymbol{x} \end{aligned} \tag{2.3}$$

と表現することができる．

Discriminator は生成データと訓練データを正確に判別することが目的であるため，目的関数全体を最大化する問題に帰着する．LoTUS を用いて変形した式 (2.3) は，$[0, 1]$ で $D(\boldsymbol{x}) = \frac{p_{data}(\boldsymbol{x})}{p_{data}(\boldsymbol{x}) + p_g(\boldsymbol{x})}$

を最大値にとるため，最適化された Discriminator の目的関数は以下の式で表現できる．

$$\begin{aligned}\max_D V(D,G) &= \mathbb{E}_{\boldsymbol{x}\sim p_{data}(\boldsymbol{x})}[\log D(\boldsymbol{x})] + \mathbb{E}_{\boldsymbol{z}\sim p_z(\boldsymbol{z})}[\log(1-D(G(\boldsymbol{z})))] \\ &= \mathbb{E}_{\boldsymbol{x}\sim p_{data}(\boldsymbol{x})}[\log D(\boldsymbol{x})] + \mathbb{E}_{\boldsymbol{x}\sim p_g(\boldsymbol{x})}[\log(1-D(\boldsymbol{x}))] \\ &= \mathbb{E}_{\boldsymbol{x}\sim p_{data}(\boldsymbol{x})}\left[\log\frac{p_{data}(\boldsymbol{x})}{p_{data}(\boldsymbol{x})+p_g(\boldsymbol{x})}\right] + \mathbb{E}_{\boldsymbol{x}\sim p_g(\boldsymbol{x})}\left[\log\frac{p_g(\boldsymbol{x})}{p_{data}(\boldsymbol{x})+p_g(\boldsymbol{x})}\right] \quad (2.4)\end{aligned}$$

一方，Generator は Discriminator が生成データを訓練データと間違うようなデータを生成することが目的である．言い換えると，Generator は生成データの分布を訓練データの分布に一致させること目的とするため，確率分布の類似度計算方法の 1 つである Jensen-Shannon ダイバージェンス (JS ダイバージェンス) を使用した分布の近似を考えると，

$$\begin{aligned}D_{JS}[p_{data}\|p_g] &= \frac{1}{2}D_{KL}\left[p_{data}\|\frac{p_{data}+p_g}{2}\right] + \frac{1}{2}D_{KL}\left[p_g\|\frac{p_{data}+p_g}{2}\right] \\ &= \frac{1}{2}\left(\int_{\boldsymbol{x}} p_{data}(\boldsymbol{x})\log\frac{2\times p_{data}(\boldsymbol{x})}{p_{data}(\boldsymbol{x})+p_g(\boldsymbol{x})}d\boldsymbol{x} + \int_{\boldsymbol{x}} p_g(\boldsymbol{x})\log\frac{2\times p_g(\boldsymbol{x})}{p_{data}(\boldsymbol{x})+p_g(\boldsymbol{x})}d\boldsymbol{x}\right) \\ &= \frac{1}{2}\left(2\log 2 + \int_{\boldsymbol{x}} p_{data}(\boldsymbol{x})\log\frac{p_{data}(\boldsymbol{x})}{p_{data}(\boldsymbol{x})+p_g(\boldsymbol{x})} + p_g(\boldsymbol{x})\log\frac{p_g(\boldsymbol{x})}{p_{data}(\boldsymbol{x})+p_g(\boldsymbol{x})}d\boldsymbol{x}\right) \\ &= \frac{1}{2}(\log 4 + V(D,C)) \quad (2.5)\end{aligned}$$

となり，Generator は JS ダイバージェンスを最小化することから，式 (2.5) は，

$$\min_G V(D,G) = 2D_{JS}[p_{data}\|p_g] - 2\log 2 \quad (2.6)$$

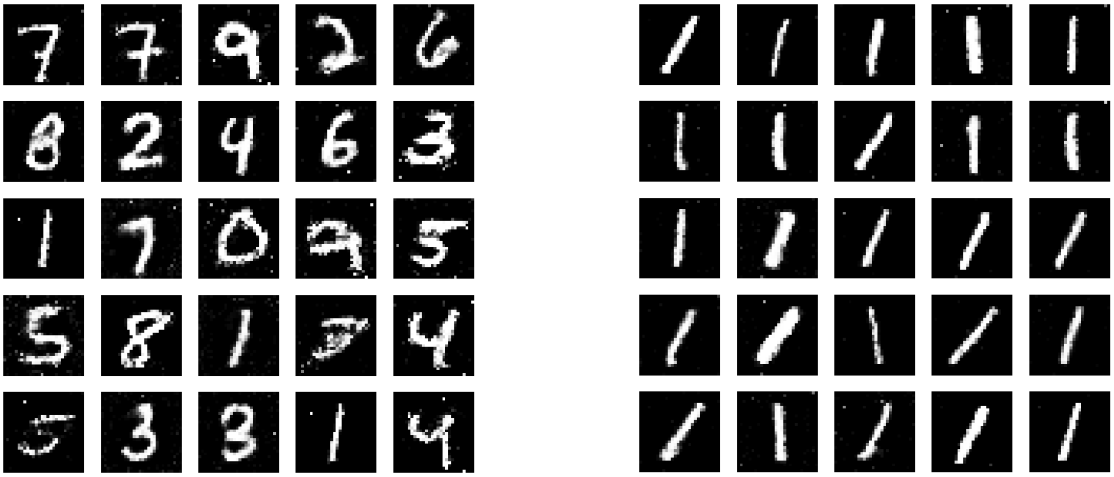
と書き換えることができる．目的関数は JS ダイバージェンスが 0 のとき，$\min_G V(D,G) = -2\log 2$ を最小値に取る．従って，GAN が最適化する目的関数は以下の式で表すことができる．

$$\min_G \max_D V(D,G) = \mathbb{E}_{\boldsymbol{x}\sim p_{data}(\boldsymbol{x})}[\log D(\boldsymbol{x})] + \mathbb{E}_{\boldsymbol{z}\sim p_z(\boldsymbol{z})}[\log(1-D(G(\boldsymbol{z})))] \quad (2.7)$$

Generator と Discriminator の重みパラメータは，学習中，交互に更新することで優れた画像生成を可能としている．

GAN の学習において，Generator と Discriminator の間のナッシュ均衡を保つことが非常に重要である．ナッシュ均衡が崩れた場合，2 つのネットワークが不安定になり，Generator がモード崩壊を起こすことが知られている．モード崩壊とは，図 2.2 (b) に示すように，Discriminator を上手く騙せた生成画像に依存して 1 種類の画像のみ生成することや，Discriminator の学習が早く進み過ぎて画像生成が困難になることを指す．

GAN は 2014 年に Goodfellow ら [12] によって提案されて以降，図 2.3 に示すように，数多くの派生手法として進化を遂げている．本章では，2.1.1 で「学習の安定化を扱った GAN」，2.1.2 で「高解像度な画像生成を行う GAN」，2.1.3 で「意図した画像生成が可能な GAN」，2.1.4 で「生成画像の評価指標」について述べる．



(a) モード崩壊が生じてない画像生成例　　(b) モード崩壊が生じた画像生成例

図 2.2: MNIST データセットを用いたモード崩壊の例.

2.1.1 安定した学習のための GAN

vanilla GAN では実画像か否かをバイナリクロスエントロピー (binary cross entropy; BCE) 誤差を用いた学習がされていた．しかし，BCE 誤差は対数を含むため Discriminator の出力値が 0 に近くなった時に値が発散し不安定になる．Least Squares GAN (LSGAN) [18] は，損失関数に対数表現を用いない二乗誤差のみで設計することで安定した学習を実現した．Wasserstein GAN (WGAN) [19] は従来の GAN で暗黙的に解いていた JS ダイバージェンスの最小化を，Wasserstein 距離の最小化で置き換えることによって不安定な学習を打破している．WGAN の損失は次式で表すことができる．

$$L_{WGAN} = \mathbb{E}_{\boldsymbol{z}\sim p_z(\boldsymbol{z})}[D(G(\boldsymbol{z}))] - \mathbb{E}_{\boldsymbol{x}\sim p_{data}(\boldsymbol{x})}[D(\boldsymbol{x})] \tag{2.8}$$

WGAN では Discriminator の出力を Wasserstein 距離と考えるため，Discriminator の重みパラメータの値を一定範囲内に収めることでリプシッツ連続を満たす関数を仮定している．この強い制約が学習を困難にするため，WGAN Gradient-Penalty (WGAN-GP) [20] は式 (2.9) のように損失関数に勾配の L2 ノルムを計算する正則化項を追加することで，重みのクリッピングと同様の効果を実現した．

$$L_{WGAN} = \mathbb{E}_{\boldsymbol{z}\sim p_z(\boldsymbol{z})}[D(G(\boldsymbol{z}))] - \mathbb{E}_{\boldsymbol{x}\sim p_{data}(\boldsymbol{x})}[D(\boldsymbol{x})] + \lambda\mathbb{E}_{\boldsymbol{z}\sim p_z(\boldsymbol{z})}[(\|\nabla_{G(\boldsymbol{z})}D(G(\boldsymbol{z}))\|_2 - 1)^2] \tag{2.9}$$

WGAN や WGAN-GP に従うと，Discriminator がリプシッツ連続を常に満たすことが重要であることは明らかである．WGAN-GP はモデルパラメータの更新以外で微分が必要となるため，WGAN に比べて多くの学習時間が必要となり，ネットワークの規模が大きくなったときに計算コストが弱点となる．そのため Miyato ら [21] は，畳み込み処理の後に置かれるバッチ正規化 [22] を Spectral Normalization に置き換えることによって，式 (2.8) のまま Discriminator 全体に制約を課すことに成功した．

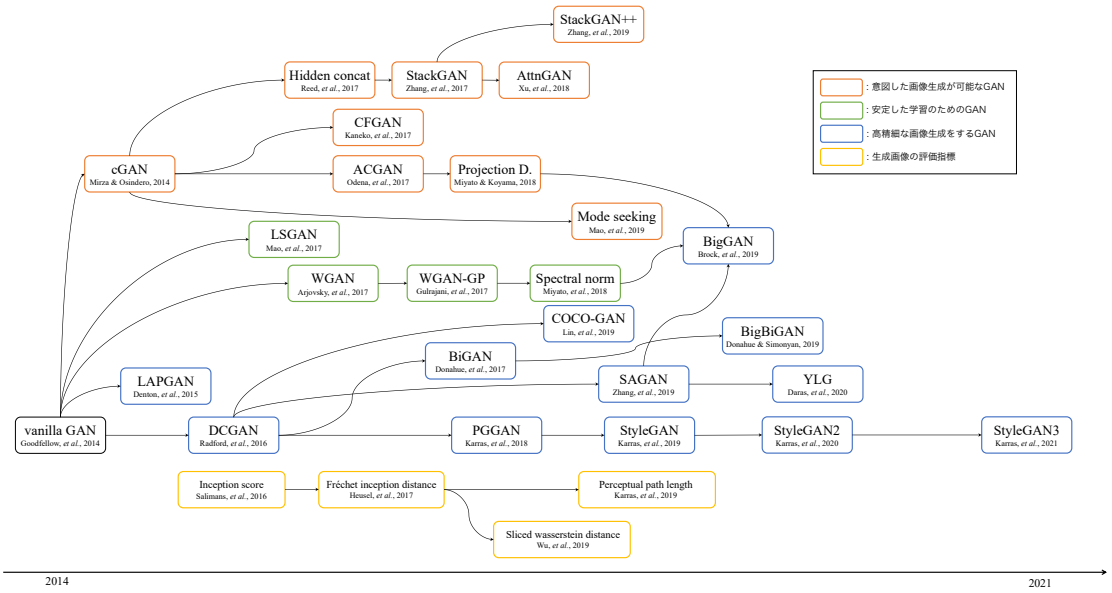


図 2.3: GAN の派生手法と評価指標の推移.

異なる観点では，Discriminator と Generator の間に能力ギャップに関しても研究がされている．Salimans ら [23] は，Generator と Discriminator をそれぞれ異なる回数パラメータ更新することで，モード崩壊が生じない安定した学習を目指した．Heusel ら [24] は，2 つのネットワークに対して異なる学習率を使用して学習する，つまり Discriminator の学習速度を遅らせることで Salimans ら [23] と同じ効果を実現した．

2.1.2 高精細な画像生成をする GAN

vanilla GAN は，全結合層を主体とした多層パーセプトロン (multi layer perceptron; MLP) で構成されているため，空間的な情報を捉えることが苦手であり複雑な画像生成が困難となる．この困難を乗り越えるために，数多くのアプローチが提案されている [25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 1, 36, 37]. 本章では代表的な手法に限定して概説する．

Deep Convolutional GAN (DCGAN) Radford らによって提案された DCGAN [26] は，畳み込み処理を主体として Generator と Discriminator を設計して学習することで，vanilla GAN よりも複雑な画像生成を実現した．DCGAN では，Generator に畳み込み処理の逆，つまり特徴マップを徐々に大きくする逆畳み込み処理 (deconvolution) や，バッチ正規化 (batch normalization) [22] を駆使して安定した学習を実現している．さらに，最適化関数に Adam [38] が使用されており，DCGAN で使用されたハイパーパラメータが GAN の学習に適切であるため，多くの手法で流用されている．DCGAN は vanilla GAN よりも複雑な画像生成を可能としているが，128×128 ピクセル以上の高解像度な画像生成を安定して行うことが困難である．

Self-Attention GAN (SAGAN) DCGAN の提案によって複雑な画像生成 (例えば，顔画像生成) が

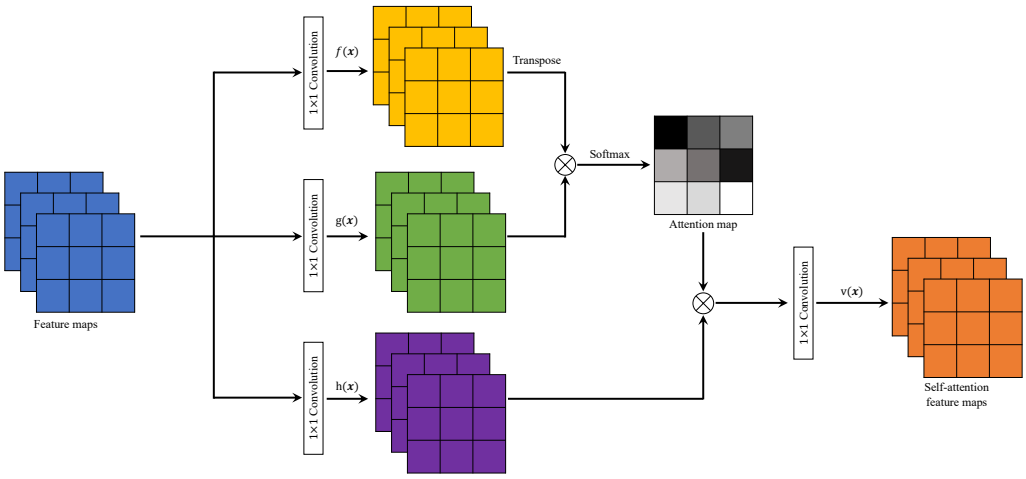


図 2.4: Self-attention 機構の処理.

可能になったものの，データセット全体で姿勢が統一されていないような画像生成は困難であった．これは，畳み込み処理や逆畳み込み処理の特性上，大域的な位置関係を捉えること苦手であることが原因とされている．顔画像のような姿勢が統一されたデータは限られており，一般物体をいかにうまく生成できるかが挑戦的な課題であった．SAGAN [29] では，図 2.4 に示すような，自己注視 (self-attention) 機構を Generator の出力付近と Discriminator の入力付近に組み込むことで，大域的な関係性を捉えた画像生成ができる．非常にシンプルな Self-attention 機構にも関わらず，SAGAN は姿勢が統一されていない一般物体の画像生成精度の飛躍的な向上を実現した．SAGAN の self-attention を Sparse Transformers [39] に倣って改良した Your Local GAN (YLG) [40] が提案されている．YLG は生成画像の品質を劇的に向上させただけでなく，self-attention の計算効率を改善することに成功した．

Progressive Growing GAN (PGGAN)　DCGAN をはじめとした多くの GAN では，潜在ベクトル $\boldsymbol{z}$ から所望の解像度まで画像をアップサンプリングしていた．しかし，Karras ら [28] はこの学習方法が複雑かつ高解像度な画像生成を困難にする原因と捉えて，Generator と Discriminator を共に成長させながら学習する PGGAN を提案した．PGGAN の具体的な学習方法は，図 2.5 に示すように，4×4 ピクセル程度の解像度をうまく生成できるようになったタイミングで，新たな層を追加して 8×8 ピクセルの画像を生成できるように学習する．この処理を所望の解像度まで繰り返すことによって，1024×1024 ピクセルの高解像度な顔画像生成を実現した．PGGAN では，画像生成に適した正規化である Pixelwise normalization を設計して，Generator の各層のバッチ正規化と置き換えて学習している．Pixelwise normalization は，正規化前後の特徴マップを，それぞれ $a_{x,y}$，$b_{x,y}$，特徴マップの総数を N，$\epsilon = 1.0 \times 10^{-8}$ とすると，次式で表される．

$$b_{x,y} = \frac{a_{x,y}}{\sqrt{\frac{1}{N}\sum_{i=0}^{N-1}(a_{x,y}^{i})^2 + \epsilon}} \tag{2.10}$$

StyleGAN　これまでの GAN では，潜在ベクトルを Generator に入力して画像を生成することが

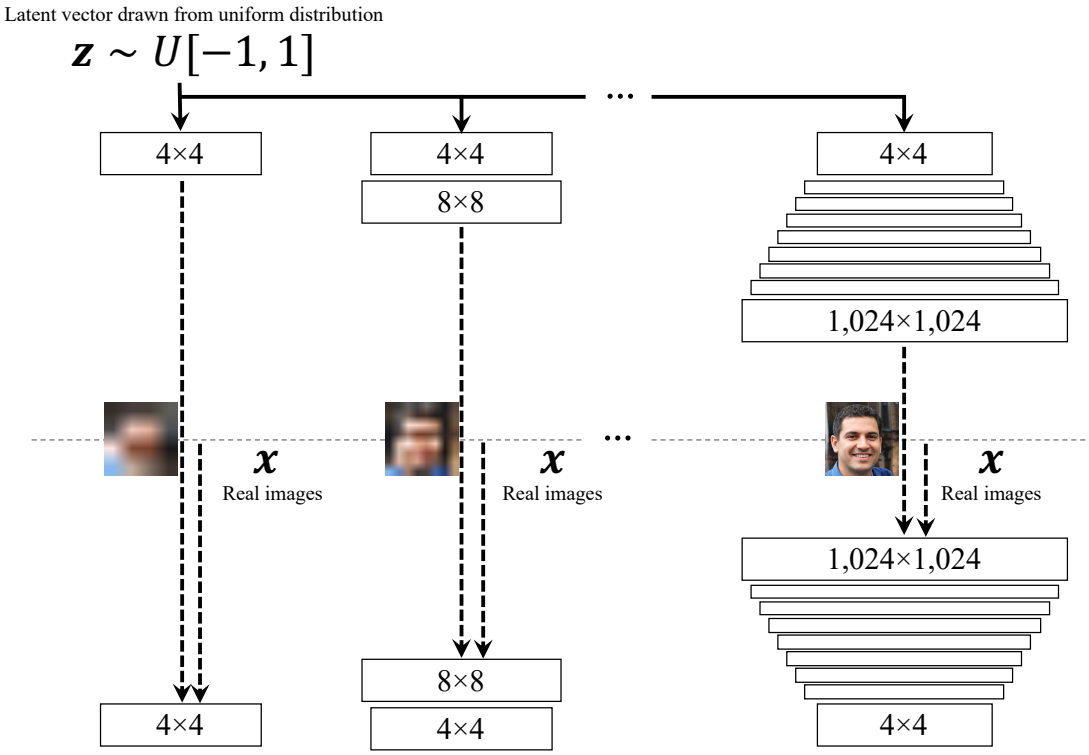


図 2.5: PGGAN の学習過程.



図 2.6: StyleGAN によって生成した画像に生じる問題. (文献 [1] より引用.)

当たり前とされていた. StyleGAN [31] は固定値から画像生成を行い，潜在ベクトル z を Generator の各層に配置された Adaptive Instance Normalization (AdaIN) [41] のパラメータとして注入する. 潜在ベクトル z は，そのまま使用するのではなく，マッピングネットワーク $f : \mathbb{R}^d \to \mathbb{R}^d$ によって別の空間に写像したものを AdaIN のパラメータとして使用する. この学習方法により，各層で適切なスタイルを表現するように学習が可能となる. 顔画像を例とすると，低解像度の時に性別など大域的な属性が生成され，高解像度の時に背景色や肌の色など詳細な情報が変化するようになる. しかし，StyleGAN で生成した画像は図 2.6 に示すように，ウォータドロップのようなノイズが生じる. この問題はスタイル変換で使用される AdaIN が画像生成に適していないため，StyleGAN2 [1] では AdaIN の処理を分解して，畳み込み処理に組み込むことで解消した. StyleGAN2 では PGGAN のようなネットワークを成長させる方法を排除して学習している. さらに，離散的な値から画像生成する StyleGAN2 を，信号処理と組み合わせて連続的な表現へ発展させた StyleGAN3 [37] も提案されい

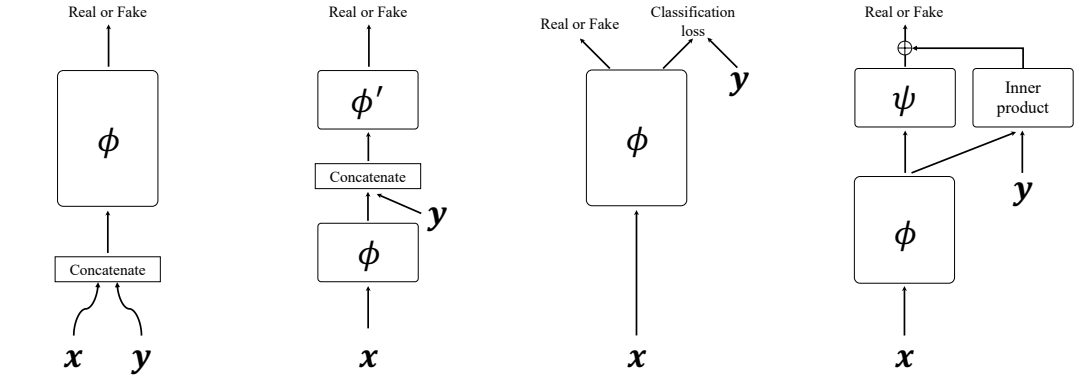


図 2.7: 各手法の条件入力方法の違い．

てる．

BigGAN　PGGAN や StyleGAN は，ネットワークを成長させる学習方法 (progressive growing) を活用することで，高解像な画像生成を実現していた．BigGAN [32] は，self-attention 機構や Spectral normalization などを全ての層に適用し，潜在ベクトル $\boldsymbol{z}$ の入力方法を工夫することで progressive growing な学習なしで高解像度な画像生成を実現した．具体的には，事前分布 $p_z(\boldsymbol{z})$ サンプリングした潜在ベクトル $\boldsymbol{z}$ を等しく分割し，全結合層によって別空間へ写像した特徴ベクトルをバッチ正規化のアフィンパラメータとして使用する．BigGAN は 2.1.3 で述べる内容にも関わりがあるため，分割した潜在ベクトルと条件を合わせて全結合層で写像する．さらに，Brock ら [32] はサンプリングした潜在ベクトルが生成に適してない場合に再度サンプリングする方法や，潜在ベクトルの値の範囲を頭打ちにする処理などを提案している．

2.1.3　意図した画像生成が可能な GAN

条件を入力に用いない vanilla GAN では，式 (2.7) の目的関数を解くことで画像生成を実現している．一方，図 2.7(a) に示す，条件を入力する conditional GAN (cGAN) [13] の場合は次式を解くように拡張される．

$$\min_G \max_D V(D,G) = \mathbb{E}_{\boldsymbol{x}\sim p_{data}(\boldsymbol{x})}[\log D(\boldsymbol{x},y)] + \mathbb{E}_{\boldsymbol{z}\sim p_z(\boldsymbol{z})}[\log(1-D(G(\boldsymbol{z},y),y))] \tag{2.11}$$

ここで，y は $\boldsymbol{x}$ に対する教師信号であり，クラスラベルや文章を使用することが多い．式 (2.11) を用いて学習することで，Generator が教師信号 y に一致するような画像を意図的に生成することが可能となる．cGAN を発展させた手法 [42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52] は数多く提案されているものの，vanilla GAN の進化に比べるとまだ発展途上だといえる．

潜在ベクトルと文章を入力とする cGAN　Reed ら [42] は文章に合った画像生成の先駆けとなる GAN を提案した．この GAN では，特徴ベクトル化した文章を潜在ベクトルと結合して Generator に入力し，

画像を生成する．Discriminator は，図 2.7(b) に示すように，中間層の特徴ベクトルに結合して真贋判定する．これにより，文章の内容を反映した画像生成が可能となったが，高解像度な画像生成が困難である．StackGAN [43] は二段階の画像生成によって高解像度かつ，与えた文章を満たすような画像生成に成功した．StackGAN では，潜在ベクトルと文章を埋め込んだ特徴ベクトルを入力して Generator により画像生成ができるように学習し，次に Encoder-Decoder 構造の Generator によって高解像度な画像生成を学習するため，ネットワーク全体の首尾一貫した学習が困難である．StackGAN++ [50] では，Generator をマルチスケールの画像生成ができるように改良し，各スケールにおいて真贋判定することで 1 段階の学習で StackGAN と同程度の品質を達成した．さらに，Attentional GAN (AttnGAN) [49] では，画像生成時に鍵となる単語を文章中から特定し，そのキーワードの attention を画像に組み込むことで StackGAN よりも優れた画像生成を可能としている．

条件入力方法の改善　従来の cGAN では，Generator と Discirminator の入力に条件を結合することで学習していた．Auxiliary Classifier GAN (ACGAN) [44] では，Discriminator の条件入力を排除する代わりに，図 2.7(c) のように出力層にクラス分類を追加することで条件入力と同様の学習を実現した．ACGAN の Generator は上手くクラス分類されるような画像を生成するように学習されるため，Discriminator 対する条件入力がなくても条件を満たす画像生成が可能となる．Projection discriminator [46] では，図 2.7(d) のように，Discriminator の中間層の特徴量と，任意の次元数の特徴ベクトルへ埋め込んだ条件を内積計算し，Discriminator 本来の出力値へ加算する．このような処理によって，離散的に表現された条件を Discriminator に適した表現へ変換することができ，cGAN による高解像度な画像生成につながる．また，Sage ら [47] は Generator の各層の特徴マップに，onehot 表現した条件を結合することで出力層付近における条件の消失を防止している．服の仮想試着 (virtual try on) [53] を扱う Poly-GAN [52] では，参照画像と骨格情報に畳み込み処理を適用して Generator の各層へ結合することで，優れた画像変換を可能とした．

2.1.4　生成画像の評価指標

生成した画像に対する定量的評価指標として，最も直感的な方法は人間が主体となって行う主観的評価である．しかし，比較手法が増加した場合や適切な被験者数の選定が困難であるため，客観的評価を行うための指標が提案されている [23, 24, 54, 31].

Inception Score (IS)　IS [23] は GAN によって生成した画像に対する評価指標の先駆けとなった計算方法である．IS は，ImageNet データセット [55] を用いて事前学習した Inception v3 [56] を使用して計算する．具体的には，まず Generator によって生成した画像 $\hat{\boldsymbol{x}}$ を n サンプル含む，空でない集合を $\mathcal{G} := \{\hat{\boldsymbol{x}}_i\}_{i=1}^{n}$ とする．そして，各生成画像 $\hat{\boldsymbol{x}}_i$ を Inception network へ入力して出力されるクラス確率 $p(y|\hat{\boldsymbol{x}}_i)$ と $\hat{\boldsymbol{x}}_i$ に関して周辺化した分布 $p(y)$ を用いて，次式の KL ダイバージェンスを用いた類似度でスコアを計算する．

$$\mathrm{IS}(G) = \exp\left(\frac{1}{|\mathcal{G}|}\sum_{\hat{\boldsymbol{x}}_i \in \mathcal{G}} D_{KL}[p(y|\hat{\boldsymbol{x}}_i)\|p(y)]\right) \tag{2.12}$$

IS は「画像の識別しやすさ」と「生成画像全体のバリエーション」の 2 つの観点で評価しており，スコアが高いほど優れた画像生成モデルであることを表している．しかし，IS は一般物体認識を学習したモデルの出力値，つまり予測確率に基づいてスコアを計算するため，顔画像や数字などのカテゴリが限定された生成画像を正確に評価することができない．そのため，一般物体以外を生成する場合は Inception network を所望のデータセットで再学習する必要があり，公平な評価をするためには不向きである．

Fréchet Inception Distance (FID) IS は生成画像群のみを用いてネットワークの出力値により評価していた．一方，FID [24] は実画像群と生成画像群を使用し，それぞれの分布の近さを評価する指標である．FID は，IS とは異なり，実画像と生成画像それぞれを Inception network に入力した時の中間層の特徴マップ $\boldsymbol{h}$ によって距離計算する．しかし，特徴マップ $\boldsymbol{h}$ を直接用いた正確な計算が困難であるため，$\boldsymbol{h}$ が多変量正規分布に従うと仮定して，平均 μ と共分散 Σ を用いた計算を行う．平均 μ と共分散 Σ は画像の集合を A，中間層の特徴マップ $\boldsymbol{h}$ の集合を H とした時，次式で計算することができる．

$$\mu = \frac{1}{|A|}\sum_{\boldsymbol{h}\in H}\boldsymbol{h} \tag{2.13}$$

$$\Sigma = \frac{1}{|A|-1}\sum_{\boldsymbol{h}\in H}(\boldsymbol{h}-\mu)(\boldsymbol{h}-\mu)^{\top} \tag{2.14}$$

そして，最終的な FID は実画像群に対する平均と共分散を μ_1 と Σ_1，生成画像群に対する平均と共分散を μ_2 と Σ_2 として次式で求められる．

$$\mathrm{FID}^2 = |\mu_1-\mu_2|^2 + \mathrm{tr}(\Sigma_1+\Sigma_2-2(\Sigma_1\Sigma_2)^{\frac{1}{2}}) \tag{2.15}$$

FID は Inception network に 2 つのデータを入力した時の特徴マップから距離計算するため，ネットワークの再学習なしで様々なデータセットの評価が可能である．

Perceptual Path Length (PPL) PPL [31] は Generator に入力する潜在ベクトルを任意の地点まで移動させた時に最短距離で移動するかどうか，つまり生成画像が我々人間の知覚に合った遷移をするかどうかを数値化することで，空間の滑らかさを評価する指標である．具体的には，白色の車と黒色の車を生成する潜在ベクトル $\boldsymbol{z}_1$ と $\boldsymbol{z}_2$ を考えたときに，滑らかで綺麗な空間であれば $\boldsymbol{z}_2$ に向かう途中で灰色の車が生成される可能性が高い．これは，色という観点で我々の知覚にあった変化をしているため，優れたモデルといえる．一方，途中で車以外の物体が生成された場合は最短経路を通ることができておらず，その Generator の空間が乱れていると考えることができる．これを数値化するための式は次のとおりである．

$$l_{\mathcal{Z}} = \mathbb{E}\left[\frac{1}{\epsilon^2}d\left(g(\mathrm{slerp}(\boldsymbol{z}_1,\boldsymbol{z}_2;t)), g(\mathrm{slerp}(\boldsymbol{z}_1,\boldsymbol{z}_2;t+\epsilon))\right)\right] \tag{2.16}$$

$$l_{\mathcal{W}} = \mathbb{E}\left[\frac{1}{\epsilon^2}d\left(g(\mathrm{lerp}(f(\boldsymbol{z}_1),f(\boldsymbol{z}_2);t)), g(\mathrm{lerp}(f(\boldsymbol{z}_1),f(\boldsymbol{z}_2);t+\epsilon))\right)\right] \tag{2.17}$$

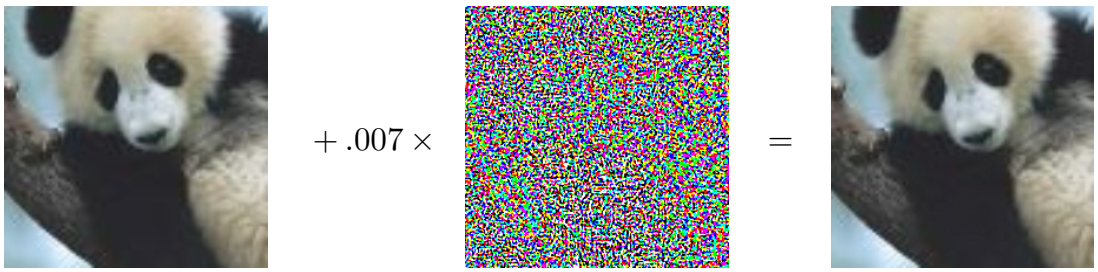


図 2.8: adversarial examples の例．(文献 [2] より引用．)

ここで，$d(\cdot, \cdot)$ は生成画像の知覚的距離 (perceptual distance)，$\text{slerp}(\cdot, \cdot; t)$ と $\text{lerp}(\cdot, \cdot; t)$ はそれぞれ，任意のパラメータ t を用いた 2 つのベクトルの球面線形補間と線形補間を表している．PPL は StyleGAN の文献 [31] 内で提案されているため，式 (2.17) のように，マッピングネットワーク $f(\cdot)$ を通した後の潜在空間に対する評価が主体で設計されている．

2.2 モデルの頑健性のための敵対的方策

CNN は優れた画像分類ができる一方，図 2.8 に示すような，悪意のある微小摂動を付与した画像 (adversarial examples) [10, 2] をいとも簡単に誤分類することが知られている．この微小摂動は，$(\boldsymbol{x}_i, y_i) \sim D$ を入力画像 $\boldsymbol{x}_i \in \mathbb{R}^{c \times h \times w}$ と教師信号 y_i のペアとした時，以下の最適化式を解くことによって求めることができる．

$$\max_{\|\boldsymbol{\delta}_i\|_p \leq \epsilon} L(f(\boldsymbol{x}_i + \boldsymbol{\delta}_i; \boldsymbol{\theta}), y_i) \tag{2.18}$$

ここで，$f : \mathbb{R}^{c \times h \times w} \to \mathbb{R}^k$ は $\boldsymbol{\theta}$ をパラメータにもつモデル，$\epsilon \geq 0$ は摂動許容範囲，$p = \{1, 2, \infty\}$ である．adversarial examples に付与される摂動は，基本的に，人間の目では知覚困難であるとされているため，CNN をベースとしたアプリケーションのセキュリティの脅威となる．adversarial examples は画像に微小摂動を加えるものだけでなく，画像の意味的情報を保持して色味やテクスチャを変更することでモデルに誤分類させるもの [57, 58, 59] も存在しているが，本稿では前者の攻撃を扱う．

このような CNN の脆弱性を緩和するために，生成モデルによるノイズ除去を分類器の前段に設ける方法 [60, 61] や，検出器を用いて adversarial examples を事前に検出する方法 [62, 63, 64, 65]，知識蒸留を使用した方法 [66] などが提案されている．その中でも，敵対的学習 (Adversarial Training) [2, 11] はシンプルな方法ながら最も効果的であるとされており，盛んに研究が進められている分野である．これらを総称して，敵対的防御 (adversarial defense) と呼ぶ．本章では，まず，代表的な敵対的攻撃手法を述べた後に，Adversarial Training の具体的な学習方法と派生手法について詳細に述べる．

2.2.1 代表的な敵対的攻撃

微小摂動に対するニューラルネットワークの脆弱性は，2013 年に Szegedy ら [10] によって発見された．Szegedy らは，入力画像 $\boldsymbol{x}_i$ が任意のクラス $l \in \{0, \ldots, k\}$ に誤分類されるような微小摂動 $\boldsymbol{r}$ を以下の最適化式を解くことで求める．

$$\text{minimize } c|\boldsymbol{r}| + L(f(\boldsymbol{x}_i + \boldsymbol{r}), l) \text{ subject to } \boldsymbol{x}_i + \boldsymbol{r} \in [0, 1]^m \tag{2.19}$$

しかし，適切な摂動を求めるためには膨大な時間を必要とするため，この研究以降で高速に強い摂動を求めるアプローチが提案されている [2, 67, 68, 11, 69, 70, 71, 72].

Fast Gradient Sign Method (FGSM) FGSM [2] は入力画像 $\boldsymbol{x}_i$ に対する損失を，入力画像の各画素に関して微分することで得る勾配方向を利用して高速に摂動を求める攻撃である．FGSM による adversarial examples の算出は以下の式で表すことができる．

$$\hat{\boldsymbol{x}_i} := \boldsymbol{x}_i + \epsilon \cdot \text{sign}\left(\nabla_{\boldsymbol{x}_i} L(f(\boldsymbol{x}_i; \boldsymbol{\theta}), y_i)\right) \tag{2.20}$$

ここで，$\text{sign}(\cdot)$ は求めた勾配から符号を抜き出すための符号関数である．式 (2.20) は $\boldsymbol{x}_i$ に対する教師信号 y_i との損失が最大になるように摂動を求めるため，基本的にどこのクラスに誤分類するか未知である．このような攻撃を，一般に非標的攻撃 (untargeted attack) と呼ぶ．一方，式 (2.19) のように，狙ったクラスに惑わす摂動作成方法のことを標的攻撃 (targeted attack) と呼ぶ．

Projected Gradient Descent (PGD) FGSM は勾配方向に摂動許容範囲 ϵ を乗算することで，1 ステップで摂動を導出していた．つまり，任意の空間において摂動の上限値または下限値のみを用いた攻撃をすることになる．PGD [11] は空間内，つまり入力画像付近により強い摂動が存在する可能性があると考えて，ステップサイズ $\alpha \leq \epsilon$ を用いて反復的に摂動を算出する．PGD による adversarial examples の算出は以下の式で表される．

$$\hat{\boldsymbol{x}}_i^{(t+1)} := \Pi_{\mathcal{B}[\boldsymbol{x}_i^{(0)}]}\left(\hat{\boldsymbol{x}}_i^{(t)} + \alpha \cdot \text{sign}\left(\nabla_{\hat{\boldsymbol{x}}_i^{(t)}} L(f(\hat{\boldsymbol{x}}_i^{(t)}; \boldsymbol{\theta}), y_i)\right)\right) \tag{2.21}$$

ここで，$\mathcal{X}$ をデータ集合とすると，$\mathcal{B}[\boldsymbol{x}_i^{(0)}] := \{\hat{\boldsymbol{x}}_i \in \mathcal{X} \mid \|\boldsymbol{x}_i - \hat{\boldsymbol{x}}_i\|_p \leq \epsilon\}$ である．したがって，$\Pi_{\mathcal{B}[\boldsymbol{x}_i^{(0)}]}$ は $\boldsymbol{x}_i^{(0)}$ を中心としたノルム空間から外れた値を空間内に引き戻す関数である．PGD は 1 サンプルに対して任意の回数反復して摂動を求めるため，速度面で FGSM に劣るものの非常に強力な摂動を用いた攻撃を可能とした．

Carlini & Wagner (CW) FGSM や PGD では高速に摂動を求めるため，モデルの勾配を利用している．CW [68] では，Szegedy ら [10] の摂動算出方法に立ち返って，適切な損失計算の選定と最適化式を改善することで強力な adversarial examples を求める攻撃である．CW は，圧倒的な防御性能を誇っていた Defensive Distillation [66] と呼ばれる知識蒸留を用いた敵対的防御を打ち破った非常に強い攻撃である．しかし，CW は Szegedy ら [10] と同様，摂動算出に膨大な時間を費やすため，CW で使用される損失関数を用いた PGD による攻撃で評価されることが一般的である．

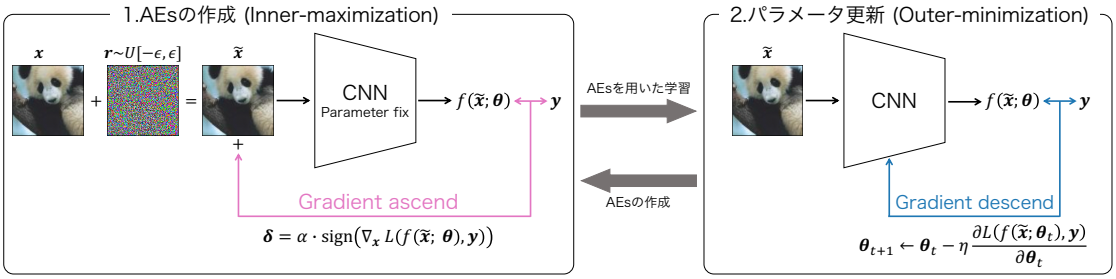


図 2.9: Adversarial Training の流れ．

2.2.2 Adversarial Training

Adversarial Training [2, 11] は学習中に adversarial examples を求めながら，その adversarial examples を誤分類しないようにモデルの重みパラメータを更新することで敵対的攻撃に対する頑健性を向上させる学習方法である．この時の注意点として，以下の 4 点が挙げられる．

1. 基本的に，adversarial examples のみ用いて重みパラメータを更新する．
2. 学習初期から adversarial examples に対する損失を最小化する．
3. 学習中のモデルにおいて adversarial examples を定義して学習する．
4. adversarial examples の算出時は，勾配が更新されないように重みパラメータを固定する．

Adversarial Training の学習プロセスは，図 2.9 に示すように，adversarial examples の作成とモデルのパラメータ更新の 2 つに分けることができる．adversarial examples の作成は，式 (2.18) の最適化式を解くことで摂動 $\boldsymbol{\delta}_i$ を求めることから，内側の最大化 (inner maximization) と呼ばれる．一方，パラメータ更新は求めた adversarial examples に対する損失が最小となるように，モデルの重みパラメータを更新することから外側の最小化 (outer minimization) と呼ばれる．これらの処理は次式によって表すことができる．

$$\min_{\boldsymbol{\theta}} \mathbb{E}_{(\boldsymbol{x}_i, y_i) \sim \mathcal{D}} \left[\max_{\|\boldsymbol{\delta}_i\|_p \leq \epsilon} L(f(\boldsymbol{x}_i + \boldsymbol{\delta}_i; \boldsymbol{\theta}), y_i) \right] \tag{2.22}$$

Adversarial Training は 1 つのネットワークで最大化問題と最小化問題を扱う点が，2.1 で述べた GAN と異なる．

Goodfellow ら [2] は inner maximization に FGSM を使用して，全結合層を主体としたニューラルネットワークに対する頑健性を向上させた．この Adversarial Training では adversarial examples と摂動のない通常の学習データの双方に対する分類誤差の最小化問題を解いている．Goodfellow ら [2] の後続研究として，PGD で求めた adversarial examples のみ用いて学習する Adversarial Training [11] が提案されており，初めて深層学習における Adversarial Training の有効性が示された．

Adversarial Training は敵対的防御の中でも非常に優れた頑健性を得られる反面，adversarial examples に対する過適合 (ロバスト過適合; robust overfitting) や，通常のサンプルに対する分類精度を著しく劣化させることが広く知られている．そのため，これらの問題点を解消するために，数多くのアプローチが提案されているだけでなく，この現象の理論的な解明も進められている．

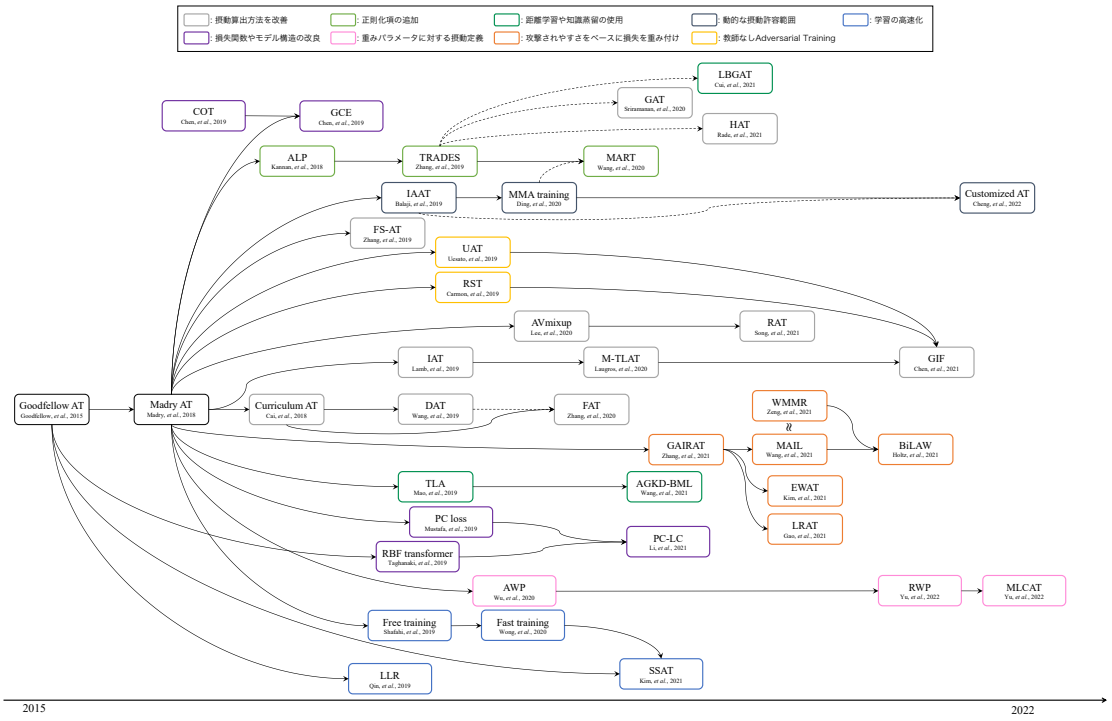


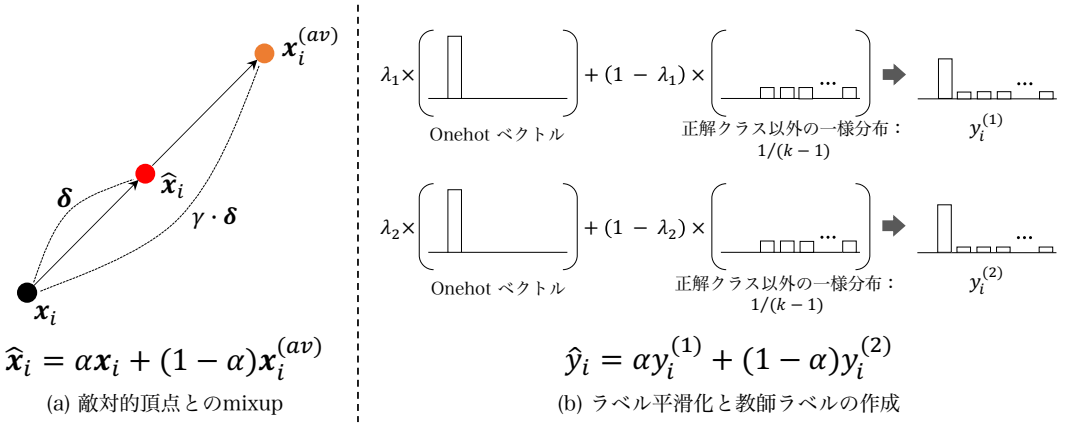
図 2.10: Adversarial Training の遷移.

2.2.3 Adversarial Training の派生手法

2.1.2 の末尾で述べたように，Adversarial Training は優れた頑健性を獲得できる一方，様々な解決すべき問題点が存在する．そのため Adversarial Training は図 2.10 に示すように，Goodfellow らの Adversarial Training (Goodfellow AT) [2] と Madry らの Adversarial Training (Madry AT) [11] を起点に盛んに研究が進められており，数多くの派生手法が提案されている．本章では図 2.10 で分類したカテゴリに沿って，手法の詳細を述べる．

■ 摂動算出方法を改善した手法

Madry AT をはじめとした，多くの Adversarial Training における Inner maximization では，学習初期から PGD で求めた強い adversarial examples を学習させている．Curriculum Adversarial Training (CAT) [73] は，このような強い adversarial examples を学習することが破滅的忘却を引き起こし，頑健性を低下させる原因となると考えて，カリキュラムラーニング [74] を用いた Adversarial Training をする．具体的には，学習初期は弱い摂動をうまく分類できるように学習して，徐々に強い摂動に遷移させる．Dynamic Adversarial Training (DAT) [75] は，inner maximization が収束しているかどうかの基準である First-Order Stationary Condition (FOSC) を用いた adversarial training を行う．FOSC は，



$\hat{x}_i = \alpha x_i + (1-\alpha) x_i^{(av)}$

(a) 敵対的頂点とのmixup

$\hat{y}_i = \alpha y_i^{(1)} + (1-\alpha) y_i^{(2)}$

(b) ラベル平滑化と教師ラベルの作成

図 2.11: AVmixup の摂動作成と，教師信号作成の例.

$\mathcal{X} = \{\boldsymbol{x} \mid \|\boldsymbol{x} - \boldsymbol{x}^0\|_\infty \leq \epsilon\}$ とすると，次式によって算出される．

$$\begin{aligned} c(\boldsymbol{x}^k) &= \max_{\boldsymbol{x} \in \mathcal{X}} \langle \boldsymbol{x} - \boldsymbol{x}^k, \nabla_{\boldsymbol{x}} f(\boldsymbol{\theta}, \boldsymbol{x}^k) \rangle \\ &= \epsilon \|\nabla_{\boldsymbol{x}} f(\boldsymbol{\theta}, \boldsymbol{x}^k)\|_1 - \langle \boldsymbol{x}^k - \boldsymbol{x}^0, \nabla_{\boldsymbol{x}} f(\boldsymbol{\theta}, \boldsymbol{x}^k) \rangle \end{aligned} \tag{2.23}$$

Friendly Adversarial Training (FAT) [76] は誤差を最大にする adversarial examples ではなく，誤分類する adversarial examples の中で最小の誤差のものを学習する．FAT では，そのような adversarial examples を見つけるために，誤分類が生じたタイミングで摂動の探索を停止する early-stopped PGD を使用している．CAT，DAT，FAT はモデルの推論能力に合わせて適切な adversarial examples を求めることによって，ロバスト過適合を緩和し優れた頑健性を獲得することができる．

学習中に mixup [16] したサンプルを学習することで，モデルの正則化を行いロバスト過適合を緩和した手法も提案されている [77, 15, 78, 79]．Interpolated Adversarial Training (IAT) [77] は各サンプルに対する adversarial examples を求めて，2 つの adversarial examples を任意の混合比で mixup [16] する．一方，Mixup with Targeted Labeling Adversarial Training (M-TLAT) [78] は 2 つのサンプルを先に mixup して，mixup したサンプルに対して adversarial examples を求め，学習する．Guided Interpolation Framework (GIF) [79] は IAT や M-TLAT のように，混合比率をランダムに決定するのではなく，識別境界付近のサンプルとなるように固定値 0.5 の比率で mixup する．モデルの頑健性を向上させるためには複雑なサンプル，つまりモデルにとって識別困難なサンプルを大量に集める必要がある [14] ため，GIF は意図的に境界付近のサンプルを生成している．これにより，IAT より優れた性能を達成した．Adversarial Vertex mixup (AVmixup) [15] は，図 2.11(a) のように，通常の摂動を係数倍して仮想的に定義した敵対的頂点 (adversarial vertex) と通常サンプルを mixup することで，通常サンプルの分類精度を劣化させずに頑健性の劇的な向上を実現した手法である．AVmixup の教師信号は，図 2.11(b) のように，ラベルスムージングを用いて定義される．Regional Adversarial Training (RAT) [80] は，2 点から直線的に補間する AVmixup と異なり，3 点から求めた任意の範囲に含まれる adversarial examples を学習させることで更なる頑健性の向上を実現した．

これら以外にも，inner maximization を特徴量の最適輸送距離の最大化にした手法 [81] や，摂動導出の初期方向を適切に求める手法 [72] などが提案されいてる．

■ 損失関数に正則化項を追加した手法

従来の Adversarial Training では，adversarial examples に対するクラス分類誤差を最小化するためにクロスエントロピー誤差が用いられている．この場合，同じクラスであれば付与される教師信号は同じとなるため，どのような adversarial examples も正解率 $p(\hat{\boldsymbol{x}}_i) \approx 1$ となるように学習される．しかし，通常サンプルと adversarial examples の違いは摂動の有無であり，たとえ摂動が付与されていたとしても上手く正解できるときは同じ特徴量を捉えるべきである．この考えに基づいて，Adversarial Logit Pairing (ALP) [82] は次式のように，クロスエントロピー誤差に加えて特徴量の二乗誤差の計算を追加した．

$$L_{\mathrm{ALP}} = L(p(\hat{\boldsymbol{x}}_i;\boldsymbol{\theta}), y_i) + \lambda \cdot \|p(\hat{\boldsymbol{x}}_i;\boldsymbol{\theta}) - p(x_i;\boldsymbol{\theta})\|_2^2 \tag{2.24}$$

TRADES [83] は，ALP とは異なり，通常サンプルに対するクロスエントロピー誤差を計算し，通常サンプルと adversarial examples それぞれの確率分布を Kullback-Leibler ダイバージェンス (KL ダイバージェンス) で近づけることで，頑健性の向上と通常サンプルに対する分類精度低下の緩和を実現した．TRADES の損失関数は以下の式で表される．

$$L_{\mathrm{TRADES}} = L(p(\boldsymbol{x}_i;\boldsymbol{\theta}), y_i) + \lambda \cdot D_{\mathrm{KL}}[p(\boldsymbol{x}_i;\boldsymbol{\theta}) \| p(\hat{\boldsymbol{x}}_i;\boldsymbol{\theta})] \tag{2.25}$$

また，TRADES では inner maximization においてクロスエントロピー誤差の最大化ではなく，通常サンプルと adversarial examples それぞれの確率分布の KL ダイバージェンスを最大化するように PGD で摂動を求める．

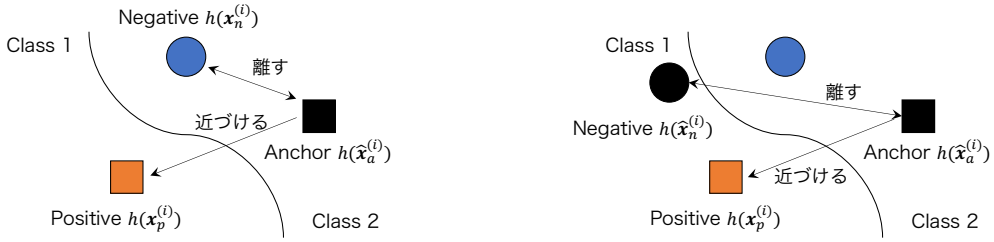
Wang ら [84] は TRADES の正則化を，通常サンプルの状態で誤分類したサンプルに限定して適用した場合，飛躍的に頑健性が向上することを発見した．これに基づいて，Wang ら [84] は次式のように，摂動のない状態のクラス確率 $p(\boldsymbol{x}_i;\boldsymbol{\theta})$ を用いて正則化項へ重み付けする Misclassification Aware Adversarial Training (MART) を提案している．

$$L_{\mathrm{MART}} = \mathrm{BCE}(p(\boldsymbol{x}_i;\boldsymbol{\theta}), y_i) + \lambda \cdot D_{\mathrm{KL}}[p(\boldsymbol{x}_i;\boldsymbol{\theta}) \| p(\hat{\boldsymbol{x}}_i;\boldsymbol{\theta})] \cdot (1 - p(\boldsymbol{x}_i;\boldsymbol{\theta})) \tag{2.26}$$

ここで，$\mathrm{BCE}(p(\boldsymbol{x}_i;\boldsymbol{\theta}), y_i) = -\log(p_{y_i}(\boldsymbol{x}_i;\boldsymbol{\theta})) - \log(1 - \max_{k \neq y_i} p_k(\boldsymbol{x}_i;\boldsymbol{\theta}))$ である．

■ 距離学習や知識蒸留を利用した手法

Triplet Loss Adversarial (TLA) training [85] は，Adversarial Training に初めて距離学習の概念を導入した学習方法である．具体的には，トリプレットロス (triplet loss) [86] を用いて，同じクラスの特



(a) TLA trainingの距離学習　　(b) AGKD-BMLの距離学習

図 2.12: TLA training と AGKD-BML の triplet loss の違い．

徴量を近づけて，異なる特徴量を離すように損失を最小化する．任意の層の特徴量 $h(\hat{\boldsymbol{x}}_i)$ に対する triplet loss は，距離関数 $D(h(\boldsymbol{x}_a^{(i)}), h(\boldsymbol{x}_{p,n}^{(i)})) = 1 - \frac{|h(\boldsymbol{x}_a^{(i)}) \cdot h(\boldsymbol{x}_{p,n}^{(i)})|}{\|h(\boldsymbol{x}_a^{(i)})\|_2 \|h(\boldsymbol{x}_{p,n}^{(i)})\|_2}$ を用いて，

$$L_{triplet}(\boldsymbol{x}_a^{(i)}, \boldsymbol{x}_p^{(i)}, \boldsymbol{x}_n^{(i)}) = [D(h(\boldsymbol{x}_a^{(i)}), h(\boldsymbol{x}_p^{(i)})) - D(h(\boldsymbol{x}_a^{(i)}), h(\boldsymbol{x}_n^{(i)})) + \alpha]_+ \tag{2.27}$$

と計算できるため，TLA training の損失関数は次式で定義される．

$$L_{\mathrm{TLA}} = L(p(\hat{\boldsymbol{x}}_a^{(i)}; \boldsymbol{\theta}), y_i) + \lambda_1 \cdot L_{triplet}(\hat{\boldsymbol{x}}_a^{(i)}, \boldsymbol{x}_p^{(i)}, \boldsymbol{x}_n^{(i)}) + \lambda_2 \cdot L_{norm} \tag{2.28}$$

ここで，$L_{norm} = \|h(\hat{\boldsymbol{x}}_a^{(i)})\|_2 + \|h(\boldsymbol{x}_p^{(i)})\|_2 + \|h(\boldsymbol{x}_n^{(i)})\|_2$ であり，λ_1，λ_2 はそれぞれハイパーパラメータである．TLA training では，adversarial examples をアンカー $\hat{\boldsymbol{x}}_a^{(i)}$，adversarial examples の正解クラスの適当なサンプルをポジティブ $\boldsymbol{x}_a^{(i)}$ として，近づくように学習し，adversarial examples が誤分類しているクラスのサンプルをネガティブ $\boldsymbol{x}_n^{(i)}$ として，離れるように学習する．この時，ネガティブサンプルは，アンカー付近のサンプルを選択するように Deep Adversarial Metric Learning [87] を用いて生成する．TLA training を拡張した手法として，双方向の距離学習を行う Attention Guided Knowledge Distillation with Bi-directional Metric Learning (AGKD-BML) [88] が提案されている．AGKD-BML は誤分類しているクラスの適当なサンプルを正解クラスに向けて targeted attack して求めた adversarial examples をネガティブサンプルとして使用する．それに加えて，CNN の注視領域が adversarial examples によって変化しないように，通常サンプルの注視領域を蒸留している．TLA training と AGKD-BML の triplet loss の違いは図 2.12 に示す通りである．

AGKD-BML では，入力サンプルに対する注視領域の蒸留を扱っていたが，あまり恩恵が得られていない．Learnable Boundary Guided Adversarial Training (LBGAT) [89] では，モデルを 2 つ用意して識別境界を蒸留する学習方法を提案している．具体的には，adversarial examples のみで学習するモデル $\mathcal{M}_{\mathrm{rob}}$ の出力値が，摂動のないサンプルのみで学習するモデル $\mathcal{M}_{\mathrm{nat}}$ の出力値に近づくように平均二乗誤差を用いて蒸留する．この時，$\mathcal{M}_{\mathrm{rob}}$ に対するクラス分類誤差の最小化，つまりクロスエントロピー誤差の最小化は行わない．一般に，KL ダイバージェンスを用いて 2 つの分布を近づけた場合はサンプルの位置が移動し，平均二乗誤差を使用した場合は識別境界が移動すると言われている．

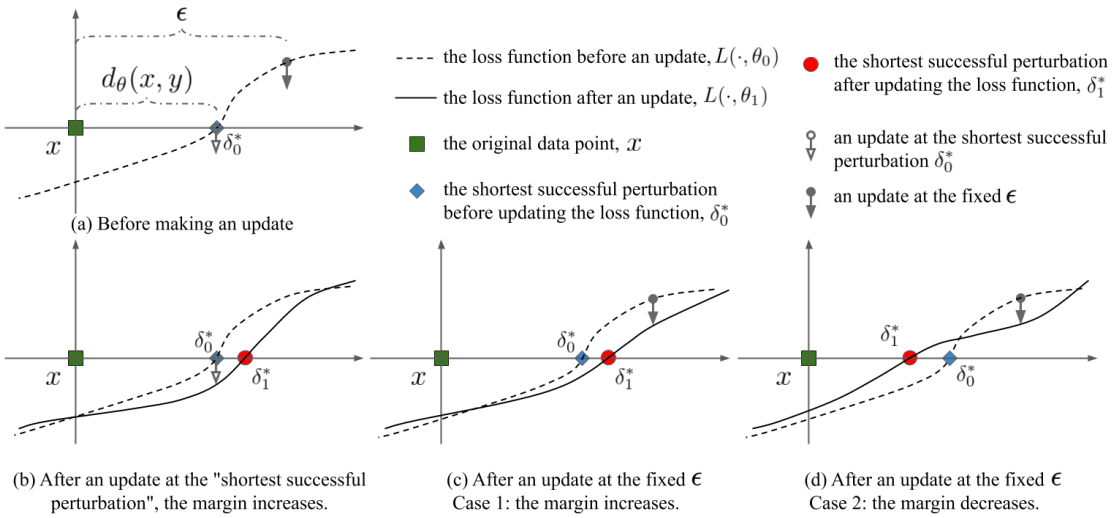


図 2.13: 異なる地点でパラメータ更新した時のマージンの関係．(文献 [3] より引用．)

■ 動的な摂動許容範囲を用いた手法

adversarial examples によって攻撃されづらいモデルを考えた時，あるサンプルと任意の識別境界までのマージンが攻撃に使用される摂動許容範囲よりも大きくなるような特徴空間を獲得することが重要となる．そのため，摂動許容範囲を大きな値で固定して学習することが考えられるが，モデルの性能劣化につながる．Ding ら [3] は従来の Adversarial Training において，学習を通じて固定値の摂動許容範囲を使用しても，図 2.13(c)(d) のようにマージンが広がる保証ができないことを示した．具体的には，摂動許容範囲の地点でパラメータ更新をした場合，識別境界とサンプルのマージンが広がる場合もあれば，狭くなる可能性もあることを示している．この結果に基づいて，Ding ら [3] は，図 2.13(b) のように，最小の摂動を付与した adversarial examples つまり，識別境界上のサンプルによってパラメータ更新する Max Margin Adversarial (MMA) training を提案した．MMA training に類似した手法として，同時期に Instance Adaptive Adversarial Training (IAAT) [90] が提案されている．MMA は二分探索によって厳密に摂動許容範囲を決定する一方，IAAT はサンプルごとに用意した摂動許容範囲を増減させて大まかに決定する．

MMA training や IAAT をさらに改善した手法が，Customized Adversarial Training (CAT) [91] である．CAT では，動的な摂動許容範囲を使用することに加えて，adversarial examples に対する適切な教師信号を作成する方法を提案した．具体的には，2 クラス分類を想定した時，優れた分類器であれば識別境界上のサンプル $\boldsymbol{x}_i$ に対する確率分布は $p(\boldsymbol{x}_i) = [0.5, 0.5]$ となることが理想であるが，onehot ラベルを用いて学習するため，そのような推論ができない学習になっている．つまり，識別境界上のサンプルであっても $p_{y_i}(\boldsymbol{x}_i) \approx 1$ が強要されている．これを解決するために，CAT では次式のように摂動許容範囲 ϵ を考慮したラベルスムージングを行う．

$$\hat{\boldsymbol{y}}_i = (1 - c\epsilon_i)\boldsymbol{y}_i + c\epsilon_i \mathrm{Dirichlet}(1) \tag{2.29}$$

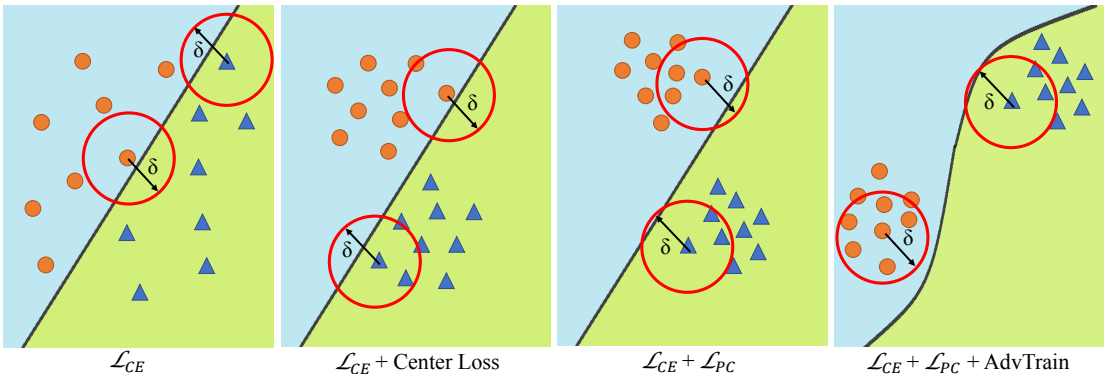


図 2.14: 損失関数と特徴空間の関係.

ここで，$\boldsymbol{y}_i$ は正解クラスが 1 でその他が 0 の onehot ベクトル，c は任意の係数，Dirichlet(1) は全てのパラメータが 1 のディリクレ分布である．CAT は適切に摂動許容範囲とラベルスムージングによって，MMA training や IAAT を上回る性能を達成した．

■ 損失関数やモデル設計を改良した手法

Goodfellow ら [2] は非線形性を高めるように設計したモデルが，敵対的攻撃に頑健となることを示した．この時，Radial Basis Function (RBF) ネットワークを用いた実験をおこなっているが，CNN に RBF カーネルを使用することをモデル設計を複雑にするだけでなく，膨大な計算コストが必要となる．そのため，Taghanaki ら [92] は RBF ユニットのすべてのパラメータを学習可能なパラメータに置き換えてモデル設計することで，CNN への適用を可能とした．さらに，RBF ユニットをマハラノビス距離を用いて設計することで，モデルの柔軟性を向上させた．このネットワークは，3 層程度の CNN において Adversarial Training なしで優れた性能を獲得できるが，多層になるにつれて計算コストの増加とともに，学習も難しくなる．

adversarial examples が容易に求められる例を考えたとき，図 2.14 の左のように，ある特徴空間で識別境界付近のサンプルほど容易に摂動を求めることができると考えることは自然である．従って，特徴量を識別境界から離して同じ特徴量を近くに集めることが重要だと考えられるが，従来の Adversarial Training ではこのような考えが考慮されていない．図 2.14 の左から 2 番目に示すように，Center loss [93] を用いることで同じクラスの特徴量は集めることができるが，境界から離れるような操作が含まれていない．そのため，Prototype Conformity loss (PC loss) [94] は学習可能なクラス重心を用いて正解クラスの特徴量を集めるだけでなく，異なるクラスの特徴量を引き離すように学習する．PC loss の損失関数は以下のように表される．

$$L_{\mathrm{pc}} = \sum_i \left\{ \|f(\boldsymbol{x}_i) - \boldsymbol{w}^c_{y_i}\|_2 - \frac{1}{k-1} \sum_{j \neq y_i} \left(\|f(\boldsymbol{x}_i) - \boldsymbol{w}^c_j\|_2 + \|\boldsymbol{w}^c_{y_i} - \boldsymbol{w}^c_j\|_2 \right) \right\} \tag{2.30}$$

式 (2.30) の第 1 項は Center loss と同様で，正解クラスの特徴量を集めるための損失計算であり，正

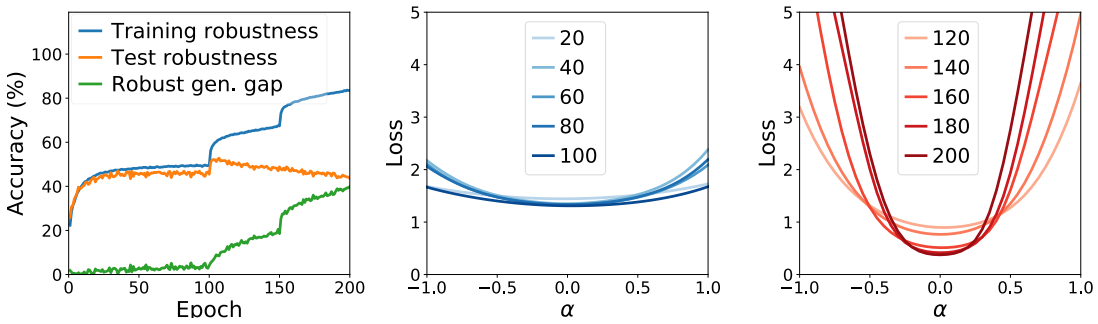


図 2.15: ロバスト過適合と誤差曲面の関係.

解クラス重心とのユークリッド距離を計算し最小化する．一方，第 2 項以降は不正解クラスの重心とのユークリッド距離と，不正解クラス重心と正解クラス重心のユークリッド距離[1]を最大化することで，異なる特徴が引き離れるように学習される．最終的に PC loss は，クロスエントロピー誤差と合わせて学習する．

Taghanaki ら [92] の手法や PC loss [94] は非常に興味深い手法であるが，ネットワーク設計の改良や，更新すべきパラメータがモデル本来のパラメータ以外にが増加するため，拡張性が非常に乏しい．この弱点を解消するためのアプローチとして，Probabilistically Compact Loss with Logit Constraints (PC-LC) [95] が提案されている．PC-LC はネットワーク設計を変更することなく，出力された確率分布のみから PC loss と同様の効果を得ることができる．PC-LC は次式を最小化するように学習される．

$$L_{\mathrm{PC-LC}} = \max\left(0, p_j(\hat{\boldsymbol{x}}_i) + \xi - p_{y_i}(\hat{\boldsymbol{x}}_i)\right) + \lambda\left(\max\left(0, z_{y_i}(\hat{\boldsymbol{x}}_i) - z_j(\hat{\boldsymbol{x}}_i) - C'\right)\right) \tag{2.31}$$

ここで，ξ，C' はそれぞれ，ハイパーパラメータ，$z(\boldsymbol{x}) \in \mathbb{R}^k$ はソフトマックス関数を適用する前のモデル出力である．第 1 項では，クラス境界を正解クラス方向に ξ だけ仮想的に移動させ，ξ 以下のクラス確率はすべて不正解と見做している．これによって，クラス中心に集まるような学習が期待される．第 2 項では，常に正を要請することで隣接クラスを跨がないように，つまり誤分類が生じないように学習を進める．

その他，正解クラス確率を高くするようにクロスエントロピー誤差を最小化するだけでなく，不正解クラスをフラットにするように学習を進める Complement Objective Training (COT) [96] や Guided Complement Entropy (GCE) [97] なども提案されている．厳密には，COT は Adversarial Training として提案されたものではなく，通常の画像分類の文脈で提案されたものである．

■ 重みパラメータに対して摂動定義する手法

Wu ら [98] は，図 2.15 に示すように，ロバスト過適合が生じたモデルを学習させ続けると誤差曲面がシャープになり，過適合が生じてないサンプルはフラットになることを特定した．これは，ロバ

[1] 三角測量の観点で考えると，重心間の計算を含めなくても暗黙的に引き離れる．

スト過適合が生じたモデルが局所的な adversarial examples を学習することを表している．そのため，Wu ら [98] は誤差を最大にする adversarial examples だけでなく，重みパラメータに対する摂動を求めて学習する Adversarial Weight Perturbation (AWP) を提案した．従って，AWP では以下の最適化式を解くことになる．

$$\min_{\boldsymbol{\theta}} \max_{\boldsymbol{v}\in\mathcal{V}} \mathbb{E}_{(\boldsymbol{x}_i,y_i)\sim\mathcal{D}} \left[\max_{\|\boldsymbol{\delta}_i\|_p\leq\epsilon} L(f(\boldsymbol{x}_i+\boldsymbol{\delta}_i;\boldsymbol{\theta}+\boldsymbol{v}),y_i) \right] \tag{2.32}$$

ここで，$\mathcal{V}$ は重みパラメータに対する摂動の集合である．AWP は，通常の Adversarial Training と比較して広範囲の重みパラメータを学習できるため，ロバスト過適合を抑制しつつ，さまざまな敵対的攻撃に対して優れた頑健性を得ることができる．

adversarial examples は各サンプルに対する損失が最大となるように摂動を求める一方，重みに対する摂動はミニバッチ全体の損失を最大にするように重みパラメータの摂動を求める．そのため，AWP はミニバッチ内すべてのサンプルから重みパラメータを変動させる摂動を求めていることになる．Yu ら [99] は損失が低いものに限定して重みパラメータに対する摂動を求める Robust Weight Perturbation (RWP) を提案した．RWP は，各サンプルの損失が c_{min} よりも小さい場合，重みパラメータに対する摂動探索時の計算に含めるように AWP の損失を改良している．RWP が重みパラメータに対する摂動を導出する際に使用する損失関数は以下の式で表すことができる．

$$\boldsymbol{v}^{t+1} \leftarrow \boldsymbol{v}^t + \nabla_{\boldsymbol{v}^t} \sum_{i=1}^{n} \mathbf{1}(\hat{\boldsymbol{x}},y_i) L(f(\hat{\boldsymbol{x}};\boldsymbol{\theta}+\boldsymbol{v}),y_i) \tag{2.33}$$

$$\text{where } \mathbf{1}(\hat{\boldsymbol{x}},y_i) = \begin{cases} 0 & \text{if } L(f(\hat{\boldsymbol{x}};\boldsymbol{\theta}+\boldsymbol{v}),y_i) > c_{min} \\ 1 & \text{if } L(f(\hat{\boldsymbol{x}};\boldsymbol{\theta}+\boldsymbol{v}),y_i) \leq c_{min} \end{cases} \tag{2.34}$$

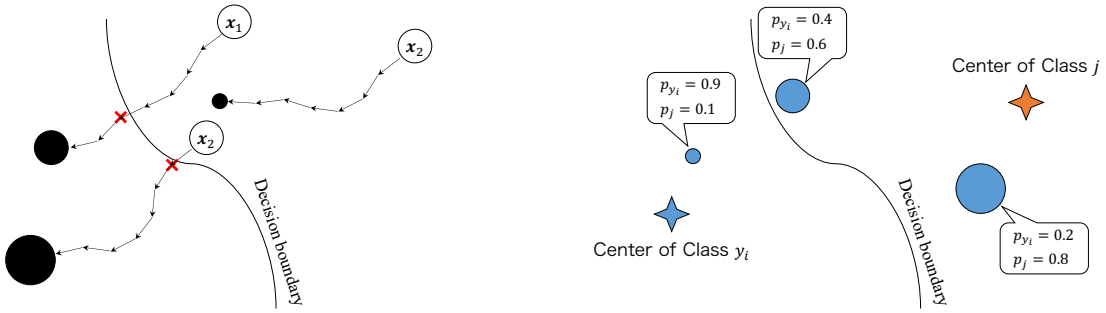
RWP を使用することで，AWP よりも優れたモデルを獲得することができる．また，Minimum Loss Constrained Adversarial Training (MLCAT) [100] は，RWP を上回る優れた頑健性を達成している．

■ 攻撃されやすさをベースとして損失へ重み付けする手法

従来の Adversarial Training は式 (2.22) を解くように学習するが，すべてのサンプルで等しく損失を扱うためロバスト過適合が生じやすい．そのため，次式のように各サンプルの損失に異なる重みを乗算して，分類誤差を最小化する手法がいくつか提案されている [101, 102, 103, 104, 105, 106].

$$\min_{\boldsymbol{\theta}} \mathbb{E}_{(\boldsymbol{x}_i,y_i)\sim\mathcal{D}} \left[\omega(\boldsymbol{x}_i+\boldsymbol{\delta}_i,y_i) \max_{\|\boldsymbol{\delta}_i\|_p\leq\epsilon} L(f(\boldsymbol{x}_i+\boldsymbol{\delta}_i;\boldsymbol{\theta}),y_i) \right] \tag{2.35}$$

Geometry-Aware Instance-Reweighted Adversarial Training (GAIRAT) [102] は，攻撃されやすさを用いて各サンプルの重要度を定義している．具体的には，図 2.16(a) に示すように，PGD において初めに誤分類したステップ数 (the least PGD steps; LPS) を用いて重要度が定義され，少ない回数のサンプルほど攻撃されやすく，多いサンプルまたは最大ステップ数と等しい場合，攻撃される可能性が低い



(a) GAIRAT　　(b) WMMR / MAIL

図 2.16: GAIRAT と WMMR/MAIL の重要度定義の違い．図形の大きさは重要度の高さを表している．

ことを表している．GAIRAT では，PGD の最大ステップ数を K，LPS を $\kappa(\boldsymbol{x}_i, y_i)$ とすると，以下の式を用いて重要度を重みへ変換する．

$$\omega(\boldsymbol{x}_i, y_i) = \frac{(1 - \tanh(\lambda + 5 \times (1 - 2 \times \kappa(\boldsymbol{x}_i, y_i)/K)))}{2} \tag{2.36}$$

ここで，λ はハイパーパラメータであり，$\lambda = \infty$ の時，通常の Adversarial Training と等しくなる．

GAIRAT は LPS に基づいて，各サンプルの重みを定義するため，PGD 以外の攻撃に対する頑健性は乏しいことが知られている．Local Reweighted Adversarial Training (LRAT) [104] は PGD 以外の攻撃 (例えば CW) に対する，脆弱性も考慮することで，PGD 以外の攻撃に対する頑健性を向上させた．また，Entropy Weighted Adversarial Training (EWAT) [105] は，GAIRAT を用いて学習したモデルが識別境界付近が曖昧な表現になることを解決するために，入力サンプルに対するエントロピーを重みとして利用する．これにより，境界付近の曖昧さが解消され，CW やより強い攻撃に対する頑健性を向上させた．

Margin-Aware Instance Reweighting Learning (MAIL) [103] は PGD の攻撃回数に基づく GAIRAT とは異なり，図 2.16(b) に示すように，クラス確率を用いて識別境界とのマージンを算出し，重み付けすることで GAIRAT の弱点を解消した．具体的には，以下の式のように，正解クラスと最も迷ったクラスからマージンを求める．

$$m(\boldsymbol{x}_i, y_i) = \arg\max_{j \neq y_i} p_j(\boldsymbol{x}_i) - p_{y_i}(\boldsymbol{x}_i) \tag{2.37}$$

GAIRAT は LPS が非負であるため，誤分類が生じているか否かを表現することができないが，式 (2.37) は正の値の時と負の値の時で誤分類しているか否かを表現することができる．MAIL は式 (2.37) で求めたマージンを以下の式を用いて重みへ変換する．

$$\omega_{\mathrm{MAIL}}(\boldsymbol{x}_i, y_i) = \mathrm{sigmoid}(\gamma \cdot (m(\boldsymbol{x}_i, y_i) + \beta)) \tag{2.38}$$

ここで，β，γ はハイパーパラメータである．GAIRAT では，重みが離散値で表現されるが，MAIL は連続値で重みを表現することができる．MAIL と同様のアプローチが Weighted MinMax Risk (WMMR) [101]

でも使用されるが，WMMR は以下の式で重みを算出する．

$$\omega_{\mathrm{WMMR}}(\boldsymbol{x}_i, y_i) = \exp(\alpha \cdot m(\boldsymbol{x}_i, y_i)) \tag{2.39}$$

MAIL は sigmoid 関数を使用する一方，WMMR は指数関数を使用する点が異なり，MAIL の方が優れた頑健性を得ることができる．MAIL や WMMR を拡張した方法として，すべてのクラスとのマージンから適切な重みを算出する Bi-level Adverarial Reweighting (BiLAW) [106] が提案されている．BiLAW は，メタ学習 [107] を用いて各サンプルに対する重みを適切に求める．

■ 教師なし Adversarial Training

Schmidt ら [14] は，ロバスト過適合を生じさせずに優れたモデルを獲得するために，通常の画像分類学習よりも複雑かつ膨大なデータを学習する必要があることを理論的に証明した．しかし，膨大なデータを収集し，各サンプルに対する教師信号のアノテーションは高コストである．そのため，教師信号を必要としない Adversarial Training として，Uesato ら [108] は Unsupervised Adversarial Training (UAT)，Carmin ら [109] は Robust Self-Training (RST) を提案した．UAT および RST 共に通常の Adversarial Training よりも頑健性と通常の分類精度を向上させることに成功した．GIF[79] ではデータ収集することもコストがかかると考え，既存のデータから mixup を用いて複雑なデータを作ることで，UAT や RST と同様の効果を得ることを目指した手法である．

■ 学習の高速化

現在主流となりつつある，Madry AT は優れた頑健性を得られる反面，学習中に PGD で adversarial examples を求めるため膨大な計算コストが必要となる．そのため，大規模データセットを用いた学習に途方もない時間を費やすことは明らかである．高速に頑健なモデルを獲得するために，FGSM を用いた Goodfellow AT の使用が考えられるが，PGD のような強い攻撃に対する頑健性を得ることが難しいとされている．さらに，Goodfellow AT によって学習したモデルはラベル漏れ (label leaking) が生じやすいとされている．ここで，label leaking とは，学習済みモデルを評価した際に敵対的攻撃に対する性能が，通常サンプルに対する性能を上回ることである．これらの問題を解消しつつ高速化を目的とした研究がされている [110, 111, 112, 113, 114, 115].

Shafahi ら [110] は，前のミニバッチで求めた摂動を次のミニバッチにおける初期値として摂動を再利用しながら，FGSM で adversarial examples を求める学習方法を提案した．この学習方法では，1 つのミニバッチに対して任意の回数 $m \leq 10$ 回反復して摂動を求めるため，学習の総数を N エポックとした時 N/m エポックの学習回数で Madry AT と同程度の性能を実現している．

Wong ら [111] は，Shafahi ら [110] の学習方法で性能向上する理由を，ランダムな初期値から FGSM を適用することであると特定した．この結果に基づいて，Wong ら [111] は $U[-\epsilon, \epsilon]$ で入力サンプルを初期化して FGSM で摂動を求める学習方法を提案した．この手法によって更なる高速化を実現した．

Fu ら [114] はモデルに内在している潜在的な頑健性を持つ部分ネットワークを宝くじ仮説 [116] によって炙り出すことで Adversarial Training なしで頑健なモデルを獲得した．

2.3 まとめ

本章では，画像処理分野における敵対的深層学習について述べた．2.1 では，まず GAN の学習方法や目的関数の導出を丁寧におこなった後に，関連する GAN の派生手法についてまとめた．2.2 では，2.2.1 で代表的な敵対的攻撃をまとめ，2.2.2 で Adversarial Training のモチベーションと具体的な処理を数式を交えながらまとめた．そして，2.2.3 で Adversarial Training の派生手法について調査し，まとめた．

以降の章では，まず 3 章では，cGAN を用いた顔画像生成における，Generator に対して適切な顔属性入力方法に関する研究に取り組む．この研究では，Weighted conditional GAN (Wc-GAN) を提案し，重み付き条件を Generator に与えることで生成画像を高品質化する．4 章では，GAN による生成画像を画像分類のデータセットとして使用したとき，分類性能を劣化させる問題について取り組む．具体的には，識別時の注視領域を考慮して生成した画像を，ベースとなる学習データが少ない場合に対する増幅データとして使用することで分類性能を向上させる．5 章では，Adversarial Training を用いて学習したモデルが，通常の分類精度を劣化させる問題について取り組む．本研究では，データ増幅手法と Adversarial Training を上手く組み合わせることによって豊富なバリエーションの摂動を学習し，通常の分類精度を保ちつつ著しい頑健性を向上を目指す．6 章では，Instance-Reweighted Adversarial Training の一種である MAIL や WMMR で算出する識別境界とのマージンが多クラス分類で不十分である問題について取り組む．本研究では，まず異なるクラス確率を持つサンプルから同じマージンが求められることを証明し，新たな指標を用いて多クラス分類に適したマージンに変換可能な Margin Reweighting を提案する．提案手法を MAIL や WMMR に組み込むことによって，従来法の頑健性を向上させる．7 章では，データ増幅の一種である mixup [16] の内挿比の算出方法について取り組む．mixup を含む多くの派生手法は，基本的にデータの合成方法に着目して研究が進められており，内挿比は確率分布から取り出したランダムな確率によって合成される．本研究では，損失が最大となる内挿比を意図的に求めて，その内挿比を活用して合成したデータに対する損失を最小化する Adversarial Interpolating Policy を提案する．提案手法はあらゆるデータセットとベースネットワークで優れた分類性能が達成できる．

第3章

重み付き条件を入力としたconditional GANによる顔画像生成とデータ増幅

画像内に映る人物がどのような属性 (性別や髪色など) であるかを認識および理解することは，顔画像から人物を照合するようなシステム開発において重要な問題である．深層学習の急速な発展に伴って，高精度な認識が可能となっている一方，学習に利用するデータに偏りが大きい場合，少量サンプルの属性の認識率は著しく低下することが知られている．人手によってサンプル数を増幅することで性能向上は期待できるが，多方面で高コストであることが挑戦的な課題となる．

深層生成モデル [17, 117, 118, 12] はデータの生成が可能であることから，人手を必要としないデータ増幅への利用が期待できる．特に，2014 年に Goodfellow らによって発表された GAN [12] は，興味深いアプローチとして脚光を浴びており，GAN は潜在変数のみから画像生成するものと，潜在変数と条件 (クラスラベルや文章など) を入力して意図した画像生成を行うものの 2 つに大別できる．前者は先進的に研究が進められている一方，後者は代表的な手法 [13, 44, 46] が提案されているものの，まだ発展途上だといえる．本研究では，顔属性の認識率向上に貢献する顔画像を cGAN [13] によって生成することを目標とする．

cGAN の Discriminator に対する条件の与え方は様々な工夫がされているにも関わらず，Generator は潜在変数と同時に条件を与える方法が主流である．そのため，深いネットワーク構造であるほど条件が考慮されづらくなり，出力層付近で条件消失が生じると考えられる．実際，この問題は [47, 52] で主張されており，Generator の入力層以外にも条件を与えることで問題に対処している．本研究でも，条件を反映した高品質な顔画像生成をするために，Generator を先行研究と同様に設計する．しかしながら，通常，顔画像は 1 つ以上の属性が定義されるマルチラベルであるため，ポジディブになる属性が複数個存在する．マルチラベルを用いて顔画像を生成するときは，各属性に対して適した層があると考える．この考えはヒトが似顔絵を描くときに全てのパーツを同時に描き始めるのではなく，大域的な情報から徐々に詳細な情報を描く感覚に類似している．この疑問を解消するための最も直感的な戦略は，人の感覚によって与える層や強度を区別することであるが，人手による最適な位置の探索は高コストかつ厳密な基準がないため困難である．そこで本研究では，条件を与える前に畳み込み処理を用いて重み付けする Weighted conditional layer (Wc-layer) を導入する．Wc-layer は入力した属性に重み付けができるため，最適な入力位置を Generator 自身が吟味しながら画像生成することができる．Discriminator は ACGAN [44] と同様に Discriminator 内部で入力画像のクラス分類を行うようマルチタスク化する．以降，クラス分類を行うブランチを Recognition branch，従来と同様に実画像か生成画像か判別するブランチを Adversarial branch と呼称する．Wc-layer とマルチタ

スク Discriminator を組み合わせて学習することで，低解像度の時にグローバルな属性，高解像度になるにつれてローカルな属性のような段階的な反映が実現できる．

評価実験では，客観的評価及び主観的評価により提案手法の有効性を示す．基本的に，客観評価では生成画像が条件を満たしているかどうかを含めた評価はできない．そこで，被験者に画像を提示して画質及び条件を考慮した評価が可能な主観評価を利用する．さらに，CNN で顔属性認識学習をする際に，提案手法で認識率の低い属性を生成して追加データとして利用した時の精度を調査する．しかし，生成画像全てが高解像度で条件を満たしている可能性は低いため，生成画像全て学習に用いた実験だけでなく，Active learning によってデータ選別した時の性能も調査する．最後に考察として，Wc-layer の畳み込み処理の重みパラメータから，各層における属性の寄与率の調査をする．これにより，提案手法を用いることで段階的に条件を反映できることを示す．

3.1 関連研究

GAN を用いて意図した画像を生成するための最もシンプルな方法として，Mirza らの提案した cGAN [13] が有名である．cGAN は，Generator と Discriminator の双方にクラスラベルや文章等の条件を与えることによって画像生成する．cGAN の応用手法は，Discriminator への条件の与え方に焦点を置いた手法が数多く提案されている．Reed ら [42] は，文章から画像の生成を前提として，Discriminator の中間層に Embedded した文章を条件として結合する手法を提案した．Odena らが提案した ACGAN [44] は，Discriminator への条件入力を省く代わりに，Discriminator 内部で入力画像をクラス識別することで，意図した画像かつ認識し易い画像の生成を実現した．Wan らの提案した手法 [119] は，マルチラベル（性別，年齢，エスニティシ）を用いて顔画像を生成するために ACGAN と同様のアプローチを使用している．Wang ら [53] は DCGAN を 3 次元に拡張することで，表情変化を捉えた GAN による画像生成法を提案した．また，Miyato らは，任意の次元数へ投影した条件を Discriminator の出力値へ加算することによって，高解像度な画像生成が可能な projection Discriminator [21] を提案した．一方，提案手法は，Generator 及び Discriminator 双方の条件入力方法に焦点をおいて高解像度な画像生成を図る．

Discriminator を改良することで高解像な画像生成を図った手法とは異なり，Generator を改良した手法も提案されている．Fused-GAN [120] は Generator の中間層で条件付きと条件なしに分岐することで，条件の詳細な情報を反映した画像生成を実現した．Yuan ら [121] は $\{0, 1\}$ で表現した顔属性をグローバルとローカルなベクトルに分割して，StackGAN [43] のように多重解像度の画像生成手法を用いることで，入力した顔属性の条件を満たした鮮明な画像生成を達成した．Sage らの提案した手法 [47] は，Generator の各層へ onehot 表現した条件ベクトルを一様に与えている．Poly-GAN [52] は，条件となる画像と骨格情報を畳み込み処理を介して各層の特徴マップのサイズに合わせた後に結合する手法である．これらの手法により，出力層付近での条件消失を予防できる．一方，提案手法は条件を出力層まで均等に反映させつつ，条件の各要素に重み付けをすることで高解像度かつ条件を満たした画像生成を目的とする．

提案手法は，入力条件が出力層付近での消失を予防するという観点で Sage らの手法と同じ設計をしているが，学習を通して最適な条件の入力位置を決定する点が異なる．Poly-GAN も同様に条件の消失を予防するために条件を層ごとに入力するが，条件に画像を用いているため，各層で与える属性の強さや種類などは考慮されていない．また，条件は入力画像の特徴抽出をする Encoder の各層にのみ与えており，画像の生成に直結する Decoder 側に条件の入力はされていない．従って，提案手法は任意の層で反映する属性の種類や強度を考慮している点が新規性である．

3.2 提案手法

本章では，提案手法の詳細を述べる．まず，3.2.1 節で本研究の問題設定を具体的に述べて，以降の節で手法の詳細を述べる．

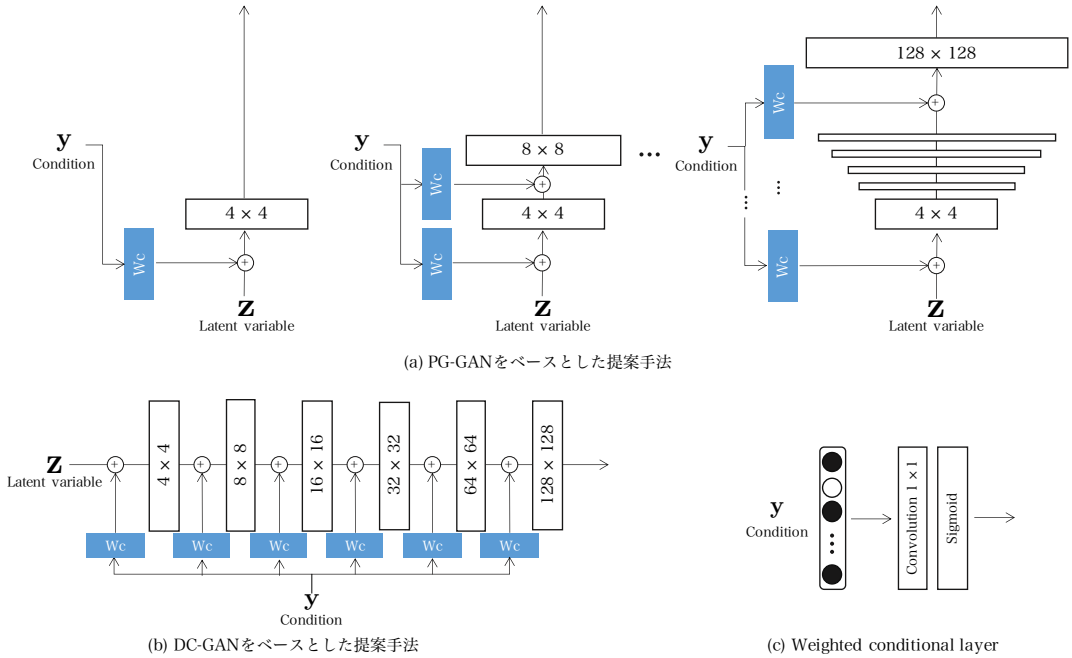


図 3.1: 提案手法を導入した Generator の構造.

3.2.1 問題設定

本研究は cGAN の学習中に条件として与える各顔属性に重み付けすることで，高精細かつ認識に有効な顔画像を生成することが目的である．Generator は $N(0,1)$ からサンプリングした d 次元の潜在変数 $\mathbf{z} \in \mathbb{R}^d$ と，n 種類の顔属性を含んだ条件ベクトル $\mathbf{y} \in \{0,1\}^n$ を用いて顔画像を生成する．$\mathbf{y}$ は重み付けして Generator の全層の特徴マップと結合する．Discriminator は実画像 $\mathbf{x}$，または生成画像 $\hat{\mathbf{x}} := G(\mathbf{z}, \mathbf{y})$ を入力して真贋判定およびクラス識別する．

3.2.2 Weighted conditional layer の導入

本研究の提案である条件へ重み付けする層を Weighted conditional layer (Wc-layer) と呼称する．Wc-layer を介して Generator の各層へ条件を与えるメリットは以下の 2 つである．

1. 出力層付近での条件の消失を予防できる．
2. 重み付けにより各層で異なる強度の条件を与えることができる．

まず，1 つ目のメリットに関して述べる．Pandey ら [52] も述べているように Generator の入力層のみに条件を与えた場合，ネットワークの深い層で特徴が消失することが考えられる．この状況に陥らないために，提案手法も先行研究と同様に Generator の全層へ条件を与えて，特徴マップをチャネル方向に結合する．これにより，ネットワーク全体で一様に等しく条件を反映する事ができるため，条件が消失する問題へ対処できる．

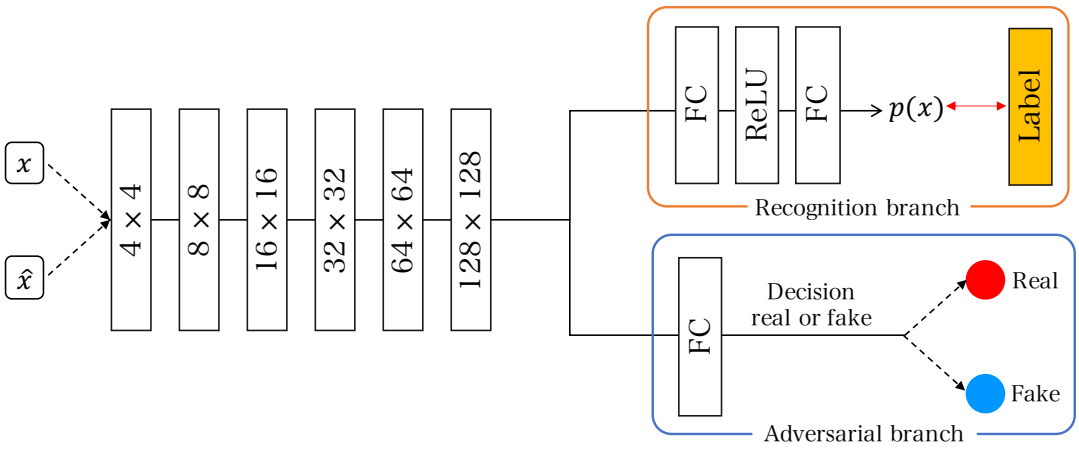


図 3.2: 提案手法の Discriminator の構造.

次に，2 つ目のメリットに関して述べる．顔画像のような 1 サンプルに対して複数の属性が付与されるマルチラベルの場合，各属性で適した入力位置や強度があると考えられる．この問題を紐解くための最も愚直な方法は人手による重み付けである．しかしながら，人手で各要素へ重み付けすることは時間的コストが高いため，非現実的である．また，人間と深層学習の知覚表現は反する可能性も高い．これらを対処するために，提案手法の Wc-layer では図 3.1(c) に示すように，カーネルサイズが 1×1 の畳み込み処理と Sigmoid 関数を用いて重み付けする．カーネルサイズは 1×1 であるため，確実に条件の各要素に重み付けが可能になる．また，重み付けしたことにより，条件が $(0, 1)$ の範囲を超える可能性があるため，入力の値を $(0, 1)$ で修めることが可能な Sigmoid 関数が活躍する．Wc-layer は，$\sigma(\mathbf{y}) := \frac{1}{1+e^{\mathbf{y}}}$ を Sigmoid 関数，重み行列 W_{conv} とバイアス $\mathbf{b} \in \mathbb{R}^m$ をパラメータに持つ畳み込み処理 $\phi(\cdot; W_{conv}, \mathbf{b})$ を用いて，以下の式で表すことができる．

$$\mathbf{h} = \sigma\left(\phi\left(\mathbf{y}; W_{conv}, \mathbf{b}\right)\right) \quad \text{s.t. } \phi : \{0,1\}^n \mapsto (0,1)^m \tag{3.1}$$

ここで，$\mathbf{h}$ は Wc-layer を適用した後の m 次元の条件ベクトルである．一見すると，Poly-GAN と提案手法の処理は等価と捉えることができるが，Poly-GAN の畳み込み処理は条件として与えるデータをダウンサンプルするために利用している点で異なる．

図 3.1(a) に示すように，提案手法のベースネットワークに PGGAN を用いた時は，層を追加すると同時に Wc-layer も追加する．提案手法のベースネットワークを DCGAN とした時は，図 3.1(b) に示すように学習初期から全ての Wc-layer を導入して画像生成を行う．この時，Wc-layer は重みを共有しないため，各層で各要素に対して最適な重みを学習することが可能となる．Wc-layer が出力する重み付けした条件は，与える層の特徴マップのサイズに合わせて拡大することで結合する．

3.2.3 マルチタスク Discriminator

本研究は認識に有効な画像を生成することが目的であるため，ACGAN と同様に Discriminator でクラス分類する．ACGAN は，Discriminator の出力層を $N+1$ ユニットとして，真贋判定とクラス分類を同時に行う．ここで，N はクラス数を表しており，残りの 1 ユニットは敵対的な尤度の出力ユニットである．しかしながら，正確なクラス分類の効果を得るためには，クラス分類と敵対的な尤度の出力を 1 つの全結合から出力することは相応しくない．

従って，本研究では Discriminator の出力層を個別のタスクに分割する．提案手法のベースネットワークを DCGAN としたときの Discriminator を図 3.2 に示す．ベースネットワークが PGGAN の時は，Generator の時同様で層を逐次追加して学習する．入力画像のクラス分類結果は Recognition branch から，実画像か生成画像かの尤度は Adversarial branch から出力する．Recognition branch は，2 つの全結合層で構成されており，確率分布 $p(\mathbf{x})$ を出力して教師信号と誤差計算する．

ポジティブな要素が 1 つでない顔属性は，Softmax 関数とクロスエントロピー誤差による計算ができない．そこで，クラス分類誤差は Sigmoid クロスエントロピー誤差を用いて計算する．Sigmoid クロスエントロピー誤差は，教師ラベルを $\mathbf{y}$，サンプル数を N として以下に示す式で表すことができる．

$$\mathcal{L}_{cls} = -\frac{1}{N}\sum_{i=1}^{N} \mathbf{y}_i^{\top} \log \sigma(\mathbf{x}_i) \tag{3.2}$$

一方，Adversarial branch は，1 つの全結合層で構成されており，真贋判定するために 1 つのスカラー値を出力する．

敵対的誤差は，ベースネットワークが DCGAN の時，式 (2.11) に式 (3.2) を加算したものを用いる．また，ベースネットワークが PGGAN の時は，式 (3.3) に式 (3.2) を加算した誤差関数を用いて学習する．

$$\begin{aligned} \mathcal{L}_{gan} &= E_{\hat{\mathbf{x}}\sim P_{\mathbf{z}}(\mathbf{z})}[D(\hat{\mathbf{x}})] - E_{\mathbf{x}\sim P_{data}(\mathbf{x})}[D(\mathbf{x})] \\ &+ \lambda E_{\tilde{\mathbf{x}}\sim P_{\tilde{\mathbf{x}}}}[(\|\nabla_{\tilde{\mathbf{x}}} D(\tilde{\mathbf{x}})\|_2 - 1)^2] \\ &+ \alpha E_{\mathbf{x}\sim P_{data}(\mathbf{x})}[D(\mathbf{x})^2] \end{aligned} \tag{3.3}$$

ここで，$\tilde{\mathbf{x}} = \epsilon\mathbf{x} + (1-\epsilon)\hat{\mathbf{x}}$，$\epsilon$ は $U[0,1]$ からサンプリングした値，$\lambda = 10$，$\alpha = 0.001$ である．

一般的に GAN の学習は不安定であるため，生成画像のバリエーションが消失するモード崩壊に陥りやすいと言われている．そこで，PGGAN で使用された Minibatch standard deviation (Minibatch stddev) [28] を最後の畳み込み処理前に導入することでモード崩壊を回避する．Minibatch stddev は，特徴マップに関して Minibatch 内の標準偏差を計算した標準偏差マップを特徴マップと合わせて畳み込むことで Minibatch 内の多様性を保証できる手法である．これはベースネットワークが DCGAN と PGGAN のどちらであっても導入する．

3.3 評価実験

提案手法を導入した各 GAN の手法により顔画像を生成し，定量的な画質評価として主観評価及び客観評価を用いて従来法と比較する．また，重み付き条件を Generator に入力する効果とマルチタスク化した Discriminator の効果を Ablation study により示す．最後に，我々の手法で生成した画像を追加して学習した顔属性認識モデルの性能を示す．このとき，生成した画像をそのまま使用する方法と，人が介入してデータを選別する方法の 2 つを使用する．一般に，GAN の生成画像は表情などが読み取りづらい崩れた画像となることがあるため，学習に悪影響を及ぼす可能性が高い．従って，人手によるデータ選別が性能向上に貢献すると考える．

3.3.1 実験概要

本実験では，条件付きに拡張した DCGAN 及び PGGAN を従来手法とする．この 2 つの手法を本実験中は，conditional DCGAN (cDCGAN) と conditional PGGAN (cPGGAN) と呼称する．提案手法を導入した DCGAN 及び PGGAN は，それぞれ WcDCGAN と WcPGGAN と定義する．

学習に使用するデータセットは，CelebA データセット [122] とする．CelebA データセットは，20 万枚を超える顔画像を保有するデータセットであり，約 1 万人の画像が含まれている．各顔画像データに対して，40 属性の顔属性ラベルの付与，5 つの顔器官点 (両目，鼻頭，口先) のアノテーションがされている．顔属性は，ポジティブな属性に 1，ネガティブな属性に-1 が付与されている．本実験では問題を簡単化するために 40 種類中 5 つの属性を使用し，ネガティブな値を 0 とする．使用する属性は，比較的変化が分かりやすい，Male (性別)，Eyeglasses (メガネ)，Smiling (笑顔)，Goatee (ヒゲ)，Bangs (前髪) を使用する．Generator に与える潜在変数の次元数は，DCGAN をベースとしたとき 100 次元，PGGAN をベースとしたとき 512 次元とする．

定量的な画質評価は，主観評価及び客観評価によって従来手法と比較する．主観評価では，21 人の被験者により画質と顔属性を考慮した評価をする．主観評価の詳細は，3.3.2 で述べる．客観評価には，Inception score (IS) [23] と Fréchet inception distance (FID) [24] を用いて評価する．IS 及び FID は，2.1.4 で解説したとおりである．

顔属性認識の実験では，101 層の ResNet を 300 エポック学習をする．最適化関数は momentum Stochastic Gradient Descent (momentum SGD) を初期学習率を 0.1，momentum を 0.9 として使用する．学習率は，$\{150, 225\}$ エポックで $1/10$ に減衰させる．認識性能は，Baseline と Active learning の有無を比較する．Baseline は，CelebA データセットから 9 割の画像を用いて学習したモデルの認識率である．追加データは 40 属性全てを用いて学習した提案手法によって顔画像を生成する．データを追加する属性は，Baseline で認識率が 90%を下回る顔属性を対象として 4,000 枚の顔画像を加える．画像生成のとき，対象クラスにポジティブなラベルが割り当てられているサンプルの教師信号を条件として使用する．対象クラス以外の 39 属性をランダムに決めた場合，現実的にあり得ない顔属性の組み合わせが生じるため，このような処理を使用する．この処理は対象クラスをランダムに変更しながら，追加データを収集する．Active learning 無しは，提案手法の生成画像を全て用いて Baseline

の結果から追加学習した際の認識率である．Active learning 有りは，性能向上に寄与すると考えられる画像を人手で選別して追加学習した認識率である．この時，生成画像の合計が 4,000 枚になるまで生成と選別を繰り返す．生成画像に対する教師信号は，画像生成時に与えた条件とすることで，アノテーションコストを削減する．

3.3.2 生成画像の主観評価

IS や FID などの客観評価は，主に生成画像の品質を重視して数値的に評価するため，生成画像が条件を満たしているかの評価は困難である．従って，人間によって品質と条件を満たしているかの判定をする主観評価が必要である．

cDCGAN と WcDCGAN を例に挙げて主観評価の説明をする．まず，cDCGAN と WcDCGAN それぞれで生成した画像を 1 枚づつ被験者に提示する．被験者は，提示された画像がどちらの画像が高品質で条件を満たしているかを選択する．この工程を 150 枚の画像に対して行う．最終的なスコアは，以下に示す式で計算する．

$$S = \frac{n}{150} \times 100 \tag{3.4}$$

ここで，n は cDCGAN または WcDCGAN それぞれの選択数を示している．cPGGAN と WcPGGAN も同様にして評価する．

3.3.3 実験結果

まず，各手法で生成した顔画像を図 3.3 に示す．目視による判断では，全ての手法において，顔と判断可能な画像が生成されており，入力された条件を反映した顔画像が生成されていることが確認できる．cDCGAN 及び WcDCGAN は，提案手法を導入することによって画質が微小に劣化した．一方，cPGGAN と WcPGGAN は，提案手法を導入しても画質が劣化することなく顔画像生成ができた．cPGGAN の生成画像である図 3.3(c) に着目すると，赤枠で囲った顔画像は上手く条件が反映されていない．つまり，最終層に至るまでに条件が消失している．このことから，DCGAN 程度のシンプル，かつ浅いネットワークであれば，通常通り条件を与えることが適しているといえる．

WcPGGAN は図 3.4 に示すように，さらに高解像度な画像を生成した場合でも，条件を満たしつつ自然で高品質な顔画像の生成ができる．高解像度の顔画像を生成することで，顔の詳細な部位まで鮮明に生成することが可能である．

次に，生成画像を定量的に評価した結果を表 3.1 に示す．生成した画像サイズは，全ての手法 128×128 [pixels] とした．cDCGAN と WcDCGAN の比較では，IS は同程度のスコアであり FID は 0.5 ポイント良いスコアで，主観評価は従来手法が高くなる．言い換えれば，求めた結果を得ることができなかった．これは，Wc-layer を各層に導入して画像生成するには Generator の層が少なすぎたためだと考える．一方，cPGGAN と WcPGGAN の比較では，IS は同等であるが，主観評価及び FID は WcPG-GAN が高いスコアとなっている．特に FID は cPGGAN と比べて 5 ポイント以上優れたス



(a) Fused-GANの顔画像生成例 (b) cDC-GANの顔画像生成例 (c) cPG-GANの顔画像生成例

(d) WcDC-GANの顔画像生成例 (e) WcPG-GANの顔画像生成例

図 3.3: 各手法の顔画像生成例 [128 × 128 pixels].

コアを達成した．定性的な評価からも確認できるとおり，PGGAN をベースにすることで DCGAN より画質が改善される．これは，PGGAN が DCGAN と比較して倍以上の畳み込み層で構成されているためであると言える．しかしながら，表 3.1 の全ての定量評価において，cPGGAN は WcPGGAN のスコアより悪いスコアであることから，深いネットワーク構造が仇となり入力層のみへ与えた条件が消失していると言える．よって，深い構造の Generator で条件を満たしつつ高精細な顔画像生成をするためには，提案手法を導入することが有効であると言える．

また，ACGAN の IS および FID は WcDCGAN のスコアと同程度である．このことから，Discriminator 側の改良は生成画像の品質にあまり影響しないことが想定されたが，ACGAN の結果は WcDCGAN より劣ったスコアであるため，僅かながら有効性があると言える．

3.3.4 Ablation study

本節では，Wc-layer および Recognition branch の効果を検証するための実験をする．Wc-layer に関する実験は以下に示す 3 種類で比較する．

- Input only: 入力層のみに条件をそのまま与えるモデル．

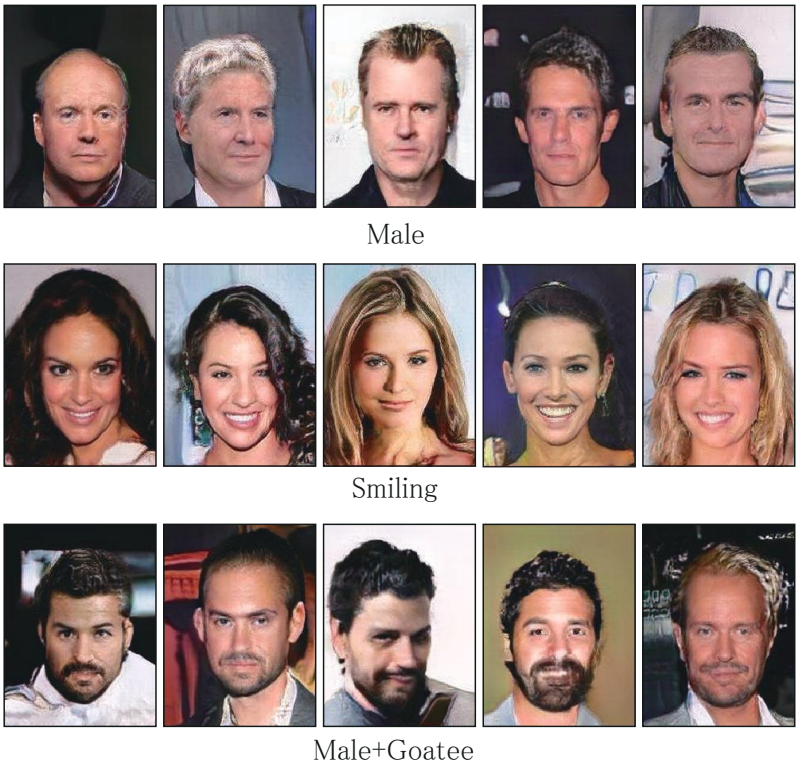


図 3.4: WcPGGAN で 192 × 256 [pixels] の顔画像生成例.

- All layers: 全ての層に条件をそのまま与えるモデル.
- All layers + Wc-layer: Wc-layer で重み付けした条件を全ての層に与えるモデル.

このとき，全てのモデルは Recognition branch を使用して学習する．従って，Input only は ACGAN と同じ構造となる．Recognition branch に関する実験は，Recognition branch の有無で比較する．このとき，Generator の Wc-layer は切除しないものとする．

表 3.1: 定量的画質評価結果.

Method	IS ↑	FID ↓	主観評価 (21 人) ↑
Real image	3.21	–	–
ACGAN	1.59	17.19	–
cDCGAN	1.70	17.22	53.1
WcDCGAN (Proposed)	1.67	**16.70**	46.9
cPGGAN	1.68	11.90	44.5
WcPGGAN (Proposed)	**1.73**	**6.10**	**55.5**

表 3.2: 異なる条件入力方法の定量的評価.

	IS ↑	FID ↓
Input only	1.59	17.19
All layers	1.62	19.23
All layers+Wc-layer	**1.73**	**6.10**

表 3.3: Recognition branch の有無の定量的評価.

Recognition branch	IS ↑	FID ↓
	1.65	10.82
✓	**1.73**	**6.10**

■ Wc-layer の効果の検証

表 3.2 に異なる条件入力方法で生成した画像に対する定量的評価結果を示す．Wc-layer を用いた提案手法は，従来手法と同様に入力層のみに条件を与えたときより IS と FID 共に優れていることが確認できる．また，提案手法から Wc-layer を除去して Sage らの手法 [47] と同じ設計にした場合も，Wc-layer を用いたときに劣る結果である．特に FID のスコアに着目すると，条件を単純に全ての層に与えたモデルの生成画像は入力層のみに条件を与えた結果より劣化したにもかかわらず，Wc-layer を導入することでスコアが飛躍的に改善していることが確認できる．このことから，全ての層に条件を与えて画像生成をする場合は，Wc-layer によって重み付けした条件を用いて画像生成することが有効であるといえる．

■ Recognition branch の効果の検証

表 3.3 に Recognition branch の有無で学習したモデルの生成画像の定量的評価結果を示す．提案手法から Recognition branch を切除することによって，IS のスコアは大差ないが FID のスコアが劣化した．これは，Recognition branch の誤差を最小化することが，Generator が鮮明な画像を生成することを促進しているといえる．従って，Recognition branch は鮮明な画像生成には必要な要素だといえる．

3.3.5 生成画像を用いた顔属性認識

顔属性識別へ生成画像を使用した際の結果を表 3.4 に示す．任意の属性の認識率 Acc_{attr} は以下に示す式によって求める．

$$Acc_{attr} = 100 \times \left(\frac{1}{N} \sum_{i=1}^{N} s_i \right), \tag{3.5}$$

$$s = \begin{cases} 1 & \arg\max_i p(x) = t \\ 0 & \text{otherwise} \end{cases} \tag{3.6}$$

表 3.4: 顔属性認識の結果（%）．w/o AL は active learning 無し，w/ AL は active learning 有りを指している．太字は，Baseline から精度が向上したことを表している．

	5 o Clock Shadow	Arched Eyebrows	Attractive	Bags Under Eyes	Bald	Bangs	Big Lips	Big Nose	Black Hair	Blond Hair	Blurry	Brown Hair	Bushy Eyebrows	Chubby	Double Chin	Eyeglasses	Goatee	Gray Hair	Heavy Makeup	High Cheekbones
Baseline	94.6	84.0	81.9	84.8	98.8	95.7	71.6	84.8	89.8	95.8	96.1	88.0	92.1	95.1	96.3	99.6	97.5	97.9	91.6	87.3
Fused-GAN	94.3	82.9	80.6	83.7	**98.9**	95.5	70.5	82.9	88.7	95.2	95.6	86.7	91.9	**95.2**	96.0	**99.7**	97.4	**98.2**	91.0	86.2
cPGGAN (w/ AL)	94.3	83.0	80.6	83.7	**98.9**	95.4	70.6	83.1	88.9	95.4	95.8	86.4	91.9	95.1	95.9	**99.7**	97.3	**98.1**	91.0	86.1
ours (w/o AL)	**94.8**	83.6	81.4	84.6	98.8	**95.8**	71.4	84.0	89.7	**95.9**	95.5	**88.5**	**92.5**	**95.3**	96.1	**99.7**	97.5	**98.2**	91.3	87.3
ours (w/ AL)	**94.8**	83.2	80.6	84.7	**99.0**	95.7	71.2	83.9	89.8	95.8	96.0	**88.5**	**92.4**	**95.3**	96.3	**99.7**	97.5	**98.2**	91.4	**87.4**

	Male	Mouth Slightly Open	Mustache	Narrow Eyes	No Beard	Oval Face	Pale Skin	Pointy Nose	Receding Hairline	Rosy Cheeks	Sideburns	Smiling	Straight Hair	Wavy Hair	Wearing Earrings	Wearing Hat	Wearing Lipstick	Wearing Necklace	Wearing Necktie	Young
Baseline	98.2	93.7	97.0	86.6	96.2	75.9	97.1	76.1	93.4	94.3	97.6	92.9	83.0	83.5	90.6	99.0	94.1	87.6	96.8	86.8
Fused-GAN	**98.4**	93.6	96.8	**86.8**	96.0	73.2	96.7	74.1	93.2	**94.5**	**97.7**	92.4	82.2	82.8	89.9	99.0	93.2	86.4	96.7	**87.2**
cPGGAN (w/ AL)	**98.4**	93.3	96.7	**86.8**	96.0	72.1	96.6	74.6	93.1	**94.6**	**97.7**	92.2	82.2	82.5	89.9	**99.1**	93.1	86.6	96.6	**87.2**
ours (w/o AL)	**98.4**	**93.9**	97.0	**87.4**	96.2	73.9	97.0	75.4	**93.6**	**95.2**	**97.8**	92.8	**83.8**	**83.9**	90.6	**99.1**	93.2	87.3	**96.9**	**87.3**
ours (w/ AL)	**98.5**	93.7	96.9	**87.1**	96.1	74.7	97.0	**76.4**	**93.6**	**94.9**	**97.9**	92.7	**83.5**	**83.6**	90.4	**99.1**	93.6	87.4	**97.0**	**87.5**

ここで，$p(x)$ は ResNet-101 へ推論画像 x を与えて出力するクラス確率，N は推論データの総数，t は ground truth のインデックスである．

まず，データ選別なしの提案手法の結果に着目すると，19 個の顔属性が Baseline から精度向上していることが確認できる．特に，”Narrow Eyes”，”Rosy Cheeks”，”Straight Hair”，”Young”は 0.5 ポイント以上の精度向上を達成した．一方，データ選別なしの Fused-GAN は，精度向上した顔属性が 9 個であった．また，cPGGAN はデータ選別をしたにも関わらず，Fused-GAN と変わらない結果であることが確認できる．Fused-GAN や cPGGAN の結果は，データを追加しても大幅に精度向上する属性が少ないことがわかる．この結果は，生成画像が学習データとして適切でないか，入力した条件を満たせていない画像が多いことが原因だと考える．

次に，データ選別ありの提案手法の結果に着目すると，精度向上した属性数はデータ選別なしの時と同様であった．しかしながら，”Pointy Nose”の認識精度はデータ選別なしの時より 1 ポイント向上しており，Baseline よりも高い精度である．これは，データ選別により鮮明な生成画像を収集したことで，モデルが細かいテクスチャを捉えることができた恩恵だといえる．

以上の結果より，提案手法は生成画像の選別しない場合でも，Baseline や他手法の結果よりも精度向上に寄与するが，鮮明な生成画像を多く学習に用いることで特徴が捉えづらい属性の認識精度向上に寄与することがわかった．また，定義が曖昧な属性はデータ選別によって鮮明な画像を追加したとしても，精度が向上しないこともわかった．

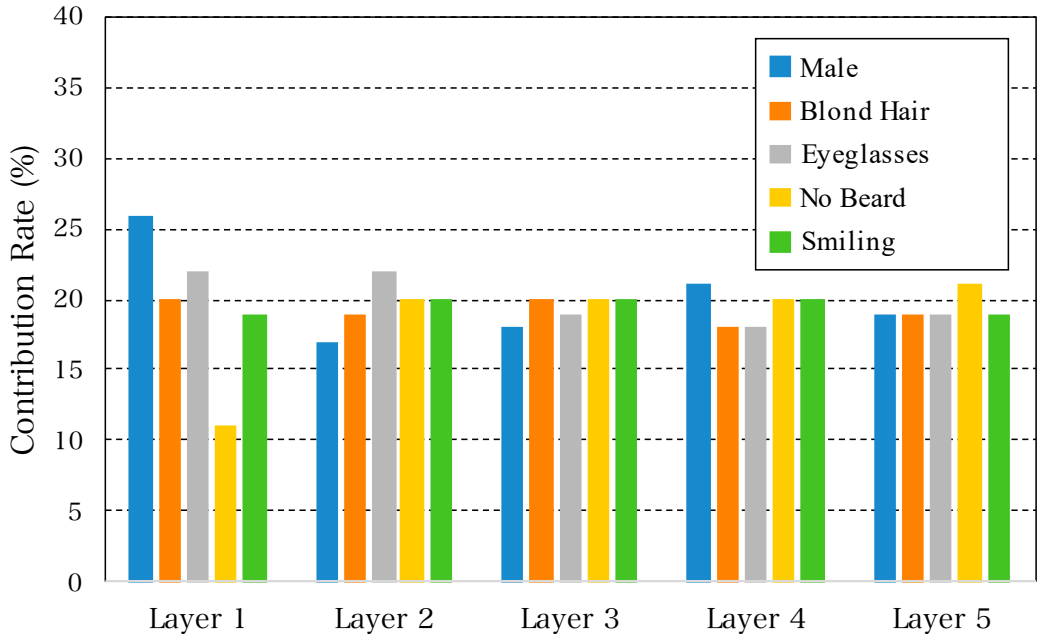


図 3.5: WcDCGAN における層ごとの顔属性の寄与率.

3.3.6 考察

提案手法で条件を満たした顔画像生成ができる要因として，重み付き条件により各層で最適な顔属性を反映していると考えられる．そこで，Generator の条件の寄与率を可視化する．寄与率は，各 Wc-layer の畳み込み層の重みフィルタを用いて算出する．寄与率 C_t は，重みフィルタ数を N，属性数を M，重みフィルタを W，寄与率を求める属性を t とすると式 (3.7) によって算出される．

$$C_t = \frac{1}{N} \sum_{n=1}^{N} \frac{|W_{t,n}|}{\sum_{m=1}^{M} |W_{m,n}|} \tag{3.7}$$

WcDCGAN および WcPGGAN の各層における各属性の寄与率を図 3.5 および図 3.6 にそれぞれ示す．ここで，WcPGGAN では，各解像度の中間生成画像が取得できるため，図 3.6 中にそれぞれ示した．中間生成画像は，Blond Hair+No Beard+Smiling を生成した時の顔画像である．

まず，WcDC-GAN の寄与率に着目すると，1 層目の性別の寄与率が高いが，基本的に全ての層において寄与率に大きなばらつきがない事が確認できる．従って，全ての層に均等な強度で属性を与えることが重要であると言える．

次に，図 3.6 の WcPGGAN の寄与率に着目すると，Male (性別) は入力層付近での寄与率が高く，出力層に近づくにつれて低くなる．したがって，中間層では既に性別が決定していると考えられる．中間生成画像においても，性別の寄与率が高くなる層から顔の輪郭が鮮明になることが確認できる．Eyeglasses (メガネ) は出力層付近で高い寄与率であり，入力層付近では，ほとんど条件として使用されていないことが確認できる．そのため，高解像度時に Eyeglasses に最も着目し，顔画像を生成して

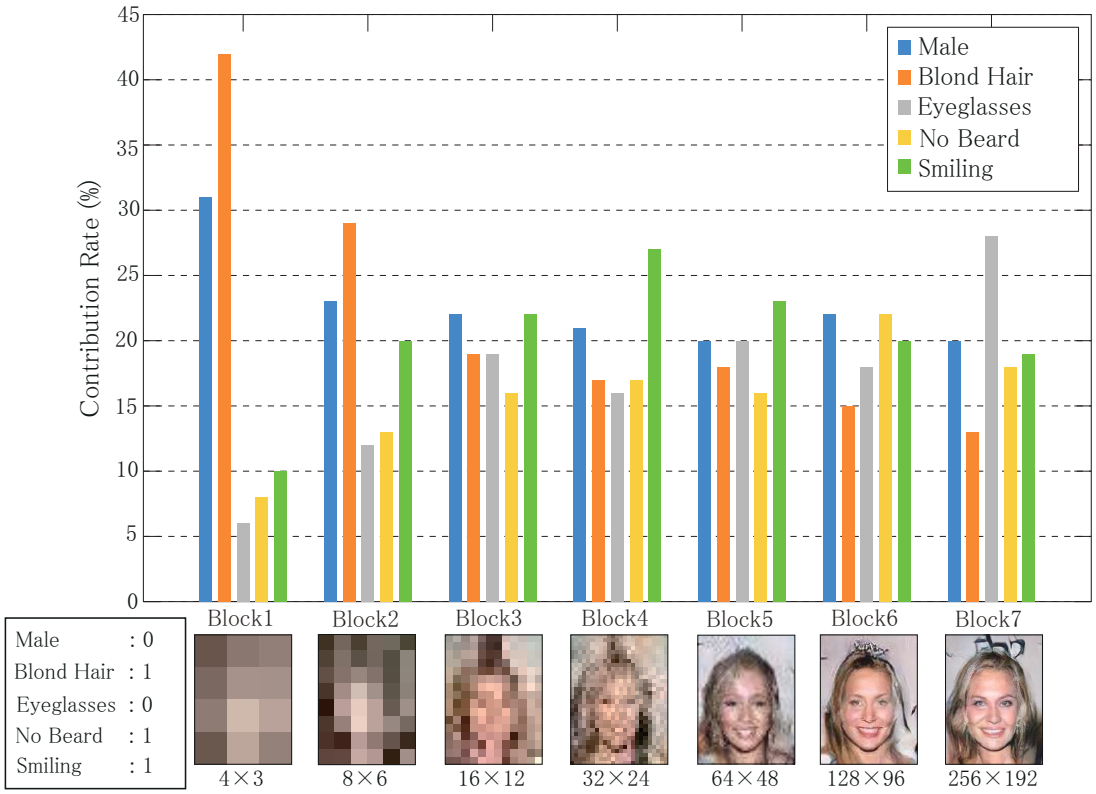


図 3.6: WcPGGAN における顔属性の寄与率と中間生成画像.

いると考えられる．Smiling (表情) は中間層で寄与率が増加し，入力及び出力層では寄与率が低い傾向にあることが確認できる．中間生成画像でも，中間層に近づくにつれ顔の表情が鮮明になる．No Beard (ヒゲ) は，各層で寄与率に大きな差はない．Blond Hair (金髪) は，入力層で多く寄与しており出力層へ近づくにつれ寄与率が低くなる．Male と Eyeglasses に着目すると，ネットワーク全体を通して寄与率が反比例していることが確認できる．したがって，性別が決定した後に装飾品等の詳細な顔属性が生成されると言える．このように，顔属性が各層で異なる寄与率であることから，学習を通じて Generator が最適な条件の反映位置を決定できたと言える．

3.4 まとめ

本章では，Weighted conditional layer (Wc-layer) を導入することで段階的に条件を反映することが可能な Weighted conditional GAN (Wc-GAN) を提案した．Wc-layer は 1×1 の畳み込み処理で構成され，各顔属性に対する重み付けを可能とした．Wc-layer を Generator の各層へ組み込むことによって，各顔属性の適切な入力位置を自動で決定することを可能とした．

定量的評価より，提案手法は PGGAN のような深いネットワーク構造を用いた際に効果的であることを確認した．また，各層に導入した Wc-layer で反映される顔属性を調査した結果，低解像度時

に性別などの大域的な属性，高解像度になるにつれて装飾品などの詳細な属性が反映される傾向にあることが判明した．生成画像を顔属性認識の学習データとして用いた場合，提案手法は人手によるデータ選別なしでも Baseline より優れた認識率を達成するが，データ選別することでいくつかの顔属性において更なる認識率向上が確認できた．

しかしながら，GAN の特性上，学習用データに含まれるサンプル数が顕著に少ない属性を上手く生成することが困難である．これは，提案手法を含めた多くの GAN の派生手法に共通した問題である．したがって，今後の課題として，学習データのサンプル数に依存しない画像生成が可能な学習方法の考案が挙げられる．

第4章

注視領域を考慮したGANによる画像生成

深層学習は膨大かつ多様性に富んだデータを学習することによって優れた性能を達成している．人手によるデータ収集やアノテーションは，データ量を増やすための直感的な解決策として考えられるが，多様なデータが収集できるとは限らないことに加えて，高コストである．この問題点を解消するために，画像処理分野では，既存データ上手く利用してデータ数を増幅することが一般的である．

データの増幅は幾何変化 (並進移動，回転) やコントラスト変化などを訓練データに適用することで，データ量と多様性を増強するテクニックである．幾何変化を適用するだけでなく，2 つのデータを任意の内挿比で合成する mixup [16] や，画像の一部を欠落する Random erasing [123] などの方法も提案されている．これらのデータ増幅は識別モデルに対して効果的である一方，既存データに任意の変換を施すため，基本的にテクスチャや識別対象の外見は不変である．そこで本研究では，GAN [12] を利用したデータ増幅に焦点を当てる．

GAN は任意の潜在変数から訓練データ分布を補間するような画像生成ができるため，高精細な画像生成 [26, 31, 29, 32, 1]，スタイル変換 [124, 125, 126]，超解像度化 [127] などに利用されている．さらに，データ増幅に有効な画像生成が可能な GAN の学習方法も提案されている [128, 129]．増幅データとしての生成画像は，高精細なほど識別モデルに良い影響を及ぼすが，CNN は識別対象の特徴的な領域に注視して識別する傾向がある [130, 131, 132]．つまり，識別対象の特徴的な領域を強調した生成画像を学習できれば，通常の生成画像でデータ増幅するより性能が向上すると考える．

そこで，本研究では CNN の識別時の注視領域を GAN の学習に組み込んだ，Discriminator-Driven Attention-Aware GAN (D2A2GAN) を提案する．我々は GAN の学習に注視領域を組み込むために，Discriminator にアテンション機構を導入する．アテンション機構は，特徴マップと注視領域を乗算することで特徴的な領域を強調する．注視領域は，Attention Branch Network (ABN) [132] の Attention branch を Discriminator の最終層へ追加することで獲得する．Discriminator は識別対象の特徴的な領域を意識して真贋判定するため，Generator は間接的に注視領域を考慮した画像生成が可能となる．GAN が生成する画像はしばしば形状が崩れる事があるため，生成画像に対する教師信号を学習データと同様に onehot ベクトルとして定義することは適切ではない．そこで，我々は Discriminator が出力する予測分布を生成画像に対する教師信号として付与する．D2A2GAN による生成画像は，サンプル数が限られたデータセットを増幅データとして利用することで性能向上させる．

4.1 関連研究

本章では，まず 4.1.1 でデータ増幅，特に GAN を用いたデータ増幅について述べ，従来手法と提案手法の違いを明確にする．次に，4.1.2 で CNN における視覚的説明について述べる．

4.1.1 データ増幅

データ増幅はモデルを訓練するためのデータに対して，学習中に幾何変化やコントラストの変化を適用することで，未知データに対する汎化性能や過学習を予防するテクニックである．幾何変化を適用したデータよりもバリエーション豊富なデータとするために，多くの研究者によって解きたいタスクに適した増幅方法が考案されている [16, 133, 123, 134]．また，GAN を用いて本質的にデータを増幅する手法も提案されている [128, 129].

Antoniou らは，訓練データを任意の次元へ投影したベクトルと潜在変数を用いて画像を生成して，データ増幅に使用する Data Augmentation GAN (DAGAN) [128] を提案した．Han らは，まず，PGGAN [28] を用いて潜在変数から画像生成をしたのちに，画像のリファイメント手法 [135, 136] によって訓練データの分布に近づける 2 段階の画像生成をしている．さらに，GAN は与えた条件を満たす画像生成ができるため，データ増幅を目的としない手法であっても生成画像を増幅データとして利用が可能である．例えば，DCGAN [26] に条件入力を加えた conditional DCGAN (cDCGAN) や，Discriminator にクラス分類器を含む ACGAN [44] の入力条件を生成画像の教師信号とすることで，訓練データを生成画像で拡張することができる．

一方，提案手法は生成画像のリファイメントなしで潜在変数と条件から画像生成するため，実画像と潜在変数から画像生成する DAGAN や 2 段階で増幅データを獲得をする Han らの手法とアプローチが異なる．また，我々は Discriminator でクラス分類することで cDCGAN より優れた画像を生成し，ACGAN と異なる設計の分類器から得る事後確率を利用してクラスが曖昧な生成画像に適した教師信号を表現する．

4.1.2 視覚的説明

CNN は優れた認識性能を発揮する一方，認識結果の根拠を解析することが困難である．Class Activation Mapping (CAM) [130] は，CNN の認識結果に対する視覚的な説明が可能な手法である．CAM は全結合層の重みと特徴マップから注視領域を求めるため，任意のクラスに対する注視領域を獲得することができる．Attention Branch Network (ABN) [132] は，注視領域を出力するブランチ (以降，Attention branch)，注視領域を反映した特徴マップから認識結果を出力するブランチ (以降，Perception branch) を用いて高性能な認識を実現した手法である．CAM は入力画像の正解クラス以外の注視領域も可視化することができるが，ABN は 1 つの注視領域のみ出力するため正解クラス以外の注視領域を可視化することができない．

ABN は優れた認識性能を達成するために，アテンション機構によって特徴レベルで認識対象の特

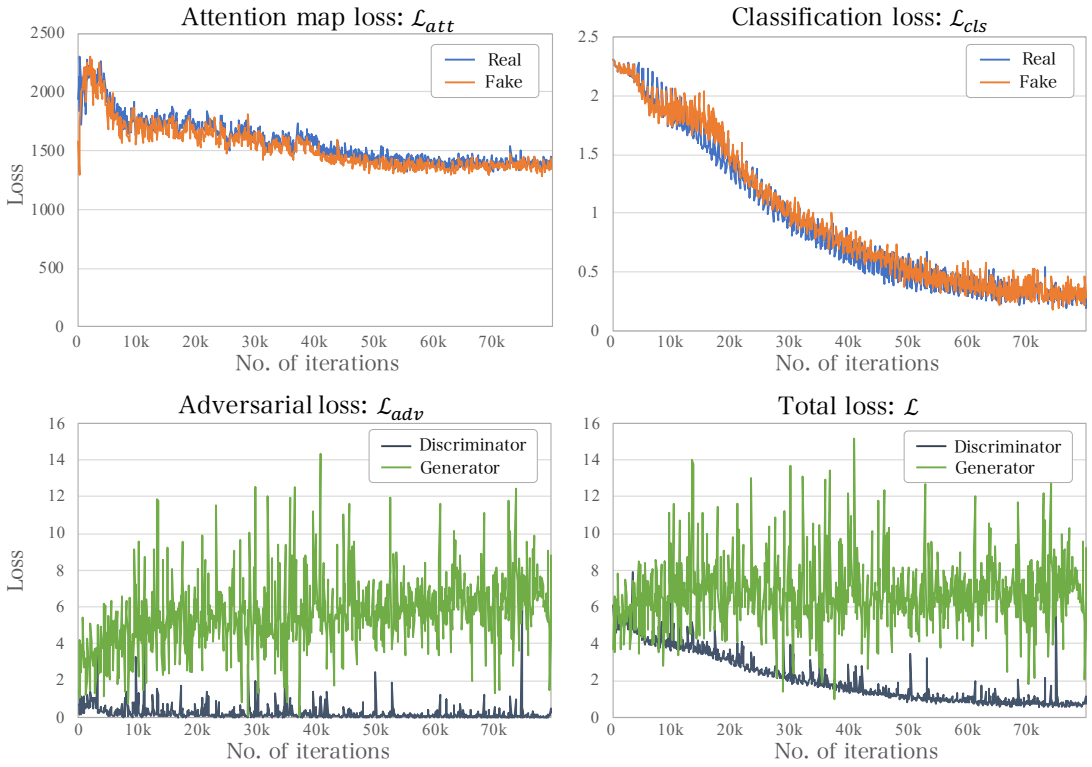


図 4.1: 学習時の提案手法の損失の推移

徴的な領域を強調している．一方，提案手法は認識性能向上が目的でなく，認識対象の特徴的な領域を考慮した画像生成することを目的としてアテンション機構を Discriminator に導入する．

4.2 提案手法

本研究では，CNN が識別時に着目する領域を考慮した画像生成が可能な Discriminator-Driven Attention-Aware GAN (D2A2GAN) を提案する．

本章では，4.2.1 で提案手法の問題設定を述べて，4.2.2 以降で提案手法の詳細について述べる．

4.2.1 問題設定

本研究では，Discriminator による真贋判定前のチャネル数が c の特徴マップ $f(\boldsymbol{x}) \in \mathbb{R}^{c \times h \times w}$ に対して，識別時の注視領域 $M(\boldsymbol{x}) \in \mathbb{R}^{h \times w}$ を反映することで，注視領域を考慮した GAN による画像生成を目的とする．Generator は，$N(0,1)$ からサンプリングした d 次元の潜在変数 $\boldsymbol{z} \in \mathbb{R}^d$ と，正解クラスが 1 でそれ以外が 0 の n 次元の条件ベクトル $\boldsymbol{y} \in \{0,1\}^n$ から画像生成する．ここで，n は分類する全クラス数を表している．Discriminator は実画像 $\boldsymbol{x} \in \mathbb{R}^{3 \times h \times w}$ または，Generator の生成画像

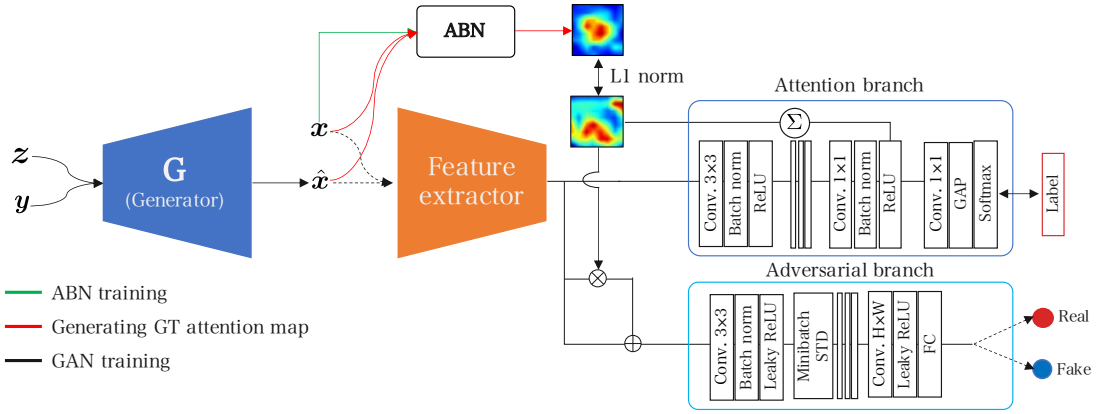


図 4.2: 提案手法の構造.

$\hat{\boldsymbol{x}} \in \mathbb{R}^{3\times h\times w}$ を入力して，真贋判定及びクラス識別を行う．クラス識別と真偽判定は，Discriminator の最終層を Attention branch と Adversarial branch の 2 つに分割して同時に行う．

提案手法は，画像生成に関する損失 $\mathcal{L}_{adv}$，クラス識別に関する損失 $\mathcal{L}_{cls}$，注視領域に対する損失 $\mathcal{L}_{att}$ を全て加算したものを最終的な損失とする．ここで，$\mathcal{L}_{att}$ は図 4.1 に示すように，他の損失とスケールが異なるため，重み $\lambda = 1 \times 10^{-4}$ を乗算することでスケールを合わせる．

$$\mathcal{L} = \mathcal{L}_{adv} + \mathcal{L}_{cls} + \lambda\mathcal{L}_{att} \tag{4.1}$$

4.2.2 注視領域を考慮した Discriminator

Discriminator は注視領域を考慮した真贋判定をするために，ABN の Attention branch とアテンション機構を導入する．提案手法のネットワーク構造を図 4.2 に示す．

■ Feature extractor

Feature extractor は画像を入力して，畳み込み処理を繰り返して任意のサイズの特徴マップを出力する．活性化関数は，最終層を除いて全ての層で Leaky ReLU を使用する．最終層の特徴マップは，Attention branch が出力した注視領域と乗算するため，非負でないと上手く特徴を強調できない．そのため，最終層の活性化関数を ReLU とすることで，特徴マップの取り得る値が $[0, \infty)$ となり注視領域をうまく反映することが可能となる．Feature extractor の処理は，各層の重みパラメータを W，Feature extractor の層数を L，$a(\cdot)$ を Leaky ReLU とすると，以下の式で表すことができる．

$$f(\boldsymbol{x};\boldsymbol{\theta}) = [W^L a_L(W^{L-1}a_{L-1}(\dots a_1(W^1\boldsymbol{x})\dots))]_+ \tag{4.2}$$

$$\text{where } [\boldsymbol{x}]_+ = \begin{cases} \boldsymbol{x} & \text{if } \boldsymbol{x} \geq 0 \\ 0 & \text{otherwise} \end{cases} \tag{4.3}$$

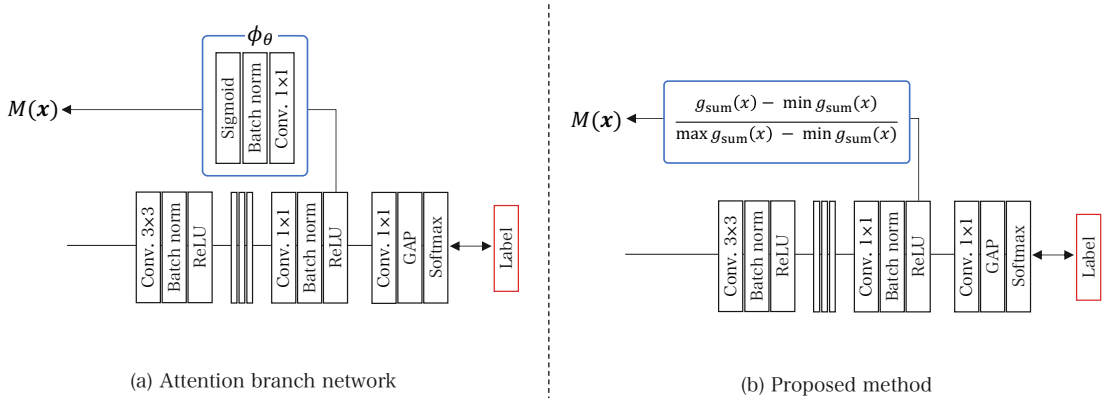


図 4.3: ABN と提案手法の Attention branch の設計の違い．

■ Attention branch

Attention branch は ABN と同様で，Feature extractor が出力した特徴マップを入力して Global Average Pooling (GAP) [137] を介したクラス識別と，注視領域を出力する．クラス識別の損失 $\mathcal{L}_{cls}$ は，サンプル数を N，Softmax 関数を $\sigma(\cdot)$ として，式 (4.4) のクロスエントロピー誤差で表される．

$$\begin{aligned} \mathcal{L}_{cls} &= -\frac{1}{N}\sum_{i=1}^{N} \boldsymbol{y}_i^\top \log \sigma(f(\boldsymbol{x}_i)) \\ &= -\frac{1}{N}\sum_{i=1}^{N} \log \sigma_{y_i}(f(\boldsymbol{x}_i)) \end{aligned} \tag{4.4}$$

ABN の Attention branch は，複数回の畳み込み処理を経て獲得した $g(\boldsymbol{x}) \in \mathbb{R}^{n\times h\times w}$ を，図 4.3(a) のように重みパラメータが $\boldsymbol{\theta}$ の畳み込み処理 ϕ を用いて，以下の式で注視領域 $M(\boldsymbol{x})$ を求める．

$$M(\boldsymbol{x}) = \phi\left(g(\boldsymbol{x}); \boldsymbol{\theta}\right) \ \text{ s.t. } \phi : \mathbb{R}^{n\times h\times w} \mapsto \mathbb{R}^{h\times w} \tag{4.5}$$

最終的に，$M(\boldsymbol{x})$ は sigmoid 関数で値域を $(0, 1)$ に正規化する．ϕ はクラス識別から独立しているが，ABN では Perception branch でもクラス識別することで，識別誤差を考慮して式 (4.5) のパラメータ $\boldsymbol{\theta}$ を更新するため，識別時の注視領域をうまく表現することができる．しかしながら，提案手法は Attention branch と Adversarial branch で異なるタスクを解くため，ABN の構造をそのまま利用すると真贋判定に関する損失のみで式 (4.5) のパラメータ $\boldsymbol{\theta}$ を更新することになり，識別時の注視領域を獲得することが困難になる．

この問題点に対処するために，提案手法では図 4.3(b) および式 (4.6) に示すように，特徴マップの最大値と最小値によって $[0, 1]$ に正規化した $M(\boldsymbol{x})$ を注視領域として扱う．

$$M(\boldsymbol{x}) = \frac{g_{\text{sum}} - \min\left(g_{\text{sum}}\right)}{\max\left(g_{\text{sum}}\right) - \min\left(g_{\text{sum}}\right)} \tag{4.6}$$

ここで，$g_{\mathrm{sum}} = \sum_{i=0}^{c} g(\boldsymbol{x})_i$, $g_{\mathrm{sum}} \in \mathbb{R}^{h \times w}$ である．ϕ_θ のカーネルサイズは 1×1 であるため空間的な特徴の集約ではなく，$g(\boldsymbol{x})$ の各ピクセルに重み付けすることでチャネル方向の情報を集約している捉えることができる．また，$g(\boldsymbol{x})$ のチャネル数と GAP 直前の畳み込み処理の入力および出力次元数は同じであるため，クラス識別に誤りがなければ $g(\boldsymbol{x})$ は識別に必要な特徴を表現しているといえる．したがって，式 (4.6) を利用して特徴を集約することで，クラス識別時の特徴量を表現した注視領域が獲得できる．このとき，式 (4.6) は学習可能なパラメータがないため，敵対誤差のみ考慮することを予防できる．

■ Adversarial branch

Adversarial branch は，Feature extractor の出力した特徴マップ $f(\boldsymbol{x})$ に注視領域 $M(\boldsymbol{x})$ を以下の式によって反映した特徴マップ $f'(\boldsymbol{x}) \in \mathbb{R}^{c \times h \times w}$ を入力とする．

$$f'(\boldsymbol{x}) = (1 + M(\boldsymbol{x}))f(\boldsymbol{x}) \tag{4.7}$$

また，Generator のモード崩壊を予防するために，ミニバッチ内の標準偏差を求めて，新たな特徴マップとして結合する Minibatch standard deviation (Minibatch STD) [28] を導入する．Adversarial branch 内の最後の畳み込み層は，特徴マップと同じサイズの重みフィルタを用いて $\mathbb{R}^{c \times h \times w} \mapsto \mathbb{R}$ とする．画像生成に関連する敵対的な損失は，従来手法と同様に式 (2.7) によって求める．

■ 注視領域に関する一貫性損失

提案手法では，ABN とは異なる方法で注視領域を求めることを可能としている．さらに正確な注視領域を学習するために，事前学習済み ABN の注視領域 $\hat{M}(\boldsymbol{x})$ を教師信号として以下の式を最小化する．

$$\mathcal{L}_{att} = \|M(\boldsymbol{x}) - \hat{M}(\boldsymbol{x})\|_1 \tag{4.8}$$

ABN と GAN の画像生成は，1 エポックごとに交互にパラメータを更新する．また，GAN のパラメータを更新するときは，ABN の重みパラメータは固定して更新しない．これによって，より正確な識別時の注視領域を考慮した画像生成が実現できる．

■ 生成画像に対する教師信号の付与

生成画像に対する教師信号は，画像生成時に用いた $\boldsymbol{y}$ を使用することが最も簡単な方法である．しかしながら，生成画像は，しばしば認識が困難なときや，クラスが曖昧なときがある．そこで，以下に示す式を用いて訓練データ，推論データ，各手法の生成画像，それぞれのエントロピーから傾向を

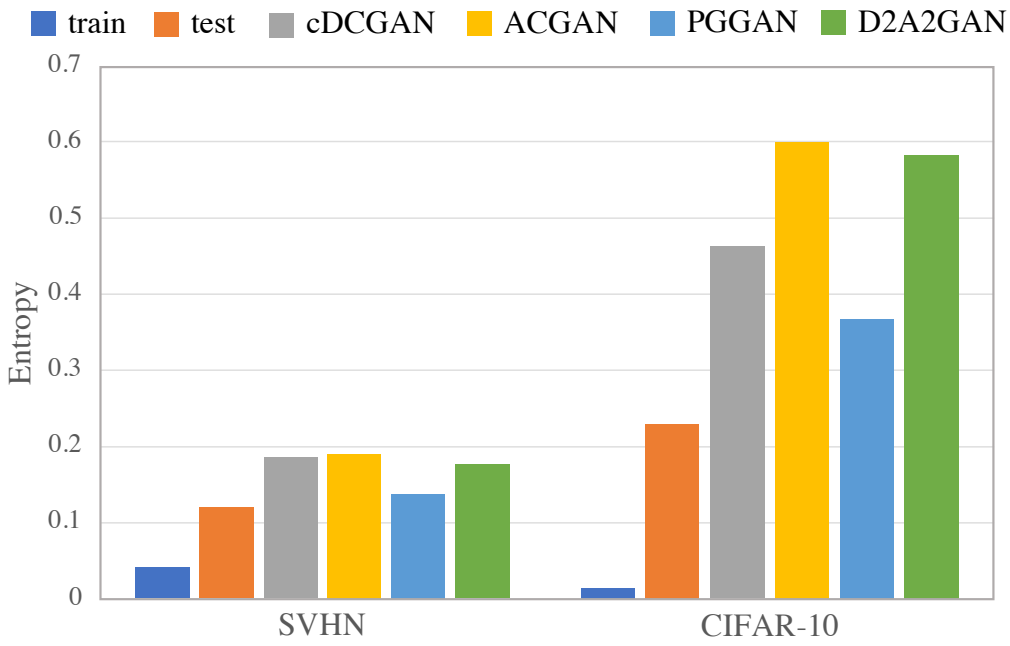


図 4.4: 実画像および各手法の生成画像の情報量エントロピー.

調査する.

$$H(p(y|x)) = -\sum_{i=1}^{N} p(y=i|x) \log p(y=i|x) \tag{4.9}$$

それぞれの画像のエントロピーを図 4.4 に示す. 2 つのデータセットとも, 生成画像のエントロピーは訓練データや推論データより高い事が確認できる. 特に, CIFAR-10 の生成画像は訓練データと比較して非常に高いエントロピーである. この結果から, 生成画像は訓練データや推論データよりも認識が困難であると言える. 従って, onehot ベクトルよりも soft な教師信号を付与する必要がある.

本研究では, 提案手法の生成画像に対して Discriminator の Attention branch が出力する事後確率を教師信号として付与する. これによって, たとえクラスが曖昧な画像であっても, モデルはうまく認識する事ができる.

4.3 評価実験

本章では, 提案手法の優位性を示すために従来手法と比較する. 実験に使用するデータセットは, CIFAR-10 と Street view house number (SVHN) [138] とする. CIFAR-10 は学習用として 50,000 サンプル, 推論用として 10,000 サンプルを有する 10 クラスの自然画像である. 各クラスのサンプル数は学習用として 5,000 サンプル, 推論用として 1,000 サンプルである. SVHN は 0 から 9 までの 10 クラスの digit データセットである. 学習用のサンプル数は約 70,000 で推論用は約 26,000 である. さらに, SVHN は追加データとして 500,000 サンプル用意されている. CIFAR-10 と SVHN は, 共に画

表 4.1: Inception score と FID の比較.

	IS ↑	FID ↓	
	CIFAR-10	CIFAR-10	SVHN
Real	9.67	–	–
cDCGAN	3.12 ± 0.03	63.0	75.7
ACGAN	4.27 ± 0.03	28.1	15.2
PGGAN	5.49 ± 0.04	19.2	12.5
D2A2GAN	4.51 ± 0.05	21.1	15.3

像サイズが 32×32 の RGB 画像である.

4.3.1 実験の詳細

本実験では，提案手法がベースにしている DCGAN [26] を条件付きにした conditional DCGAN (cDCGAN) と，提案手法と同様で Discriminator にクラス識別器が導入されている ACGAN [44]，および PGGAN を先行研究として用いる．また，提案手法の生成画像に対する教師信号を onehot ベクトルとしたとき (Hard target) と Discriminator の Attention branch が出力した予測分布としたとき (Soft target) の 2 種類を比較する．

生成画像が増幅データとして有効であるかどうかを確認するために，畳み込み処理が 18 層の Residual network (ResNet) [5] の学習に用いて推論性能を比較する．ベースとなる訓練データは {100, 1000} として，10 クラス全てのサンプル数が等しくなるようにランダムに抽出する．増幅データである生成画像は，{100, 1000, 10000, 50000} とする．cDCGAN の生成画像に対する教師信号は Hard target のみ使用する．ACGAN は提案手法と同様で，Discriminator にクラス識別を含むため，Hard target および Soft target の双方の性能を比較する．

生成画像の画質評価には，Inception score (IS) [23] および Fréchet inception distance (FID) [24] を使用する．IS は計算方法の性質上 SVHN を評価するために Inception network [56] を再度学習する必要があるため，CIFAR-10 のみ評価する．各評価指標の詳細な計算方法は 2.1.4 のとおりである．

4.3.2 生成画像の画質評価

表 4.1 に各手法の生成画像の IS と FID を示す．cDCGAN と提案手法を比較すると，提案手法は IS と FID 共に優れた数値である．ACGAN と提案手法は，IS の値と SVHN の FID の値が同程度の値であるが，IS の偏差の値が高いことから生成画像のバリエーションが豊富であると言える．CIFAR-10 の FID の値は，ACGAN よりも 7 ポイント良い結果である．PGGAN の IS の値は提案手法より 1 ポイント近く優れているが，FID の値は大幅な違いはなかった．

次に，各手法の生成画像と提案手法で獲得した注視領域を図 4.5 に示す．SVHN の生成画像に着目す



図 4.5: 各手法の生成画像と提案手法によって獲得した注視領域.

ると，全ての手法で数字と認識可能な画像が獲得できている．各手法で画像を比較すると，cDCGAN は生成画像のバリエーションが不足しているといえる．一方，cDCGAN 以外の手法で生成した画像は同程度のバリエーションであるが，視覚的なクォリティは PGGAN(c) が最も良い．CIFAR-10 の生成画像は，全ての手法が視覚的にクラスを特定することが困難な画像である．提案手法の注視領域に着目すると，各クラスの特徴的な領域が強く発火していることが確認できる．特に，赤枠で示した注視領域は生成したクラスのシルエットを捉えるような領域に注視している．

以上の結果をまとめると，視覚的には全手法で同程度のクォリティで，定量的には PGGAN に次いで提案手法が優れた結果である．

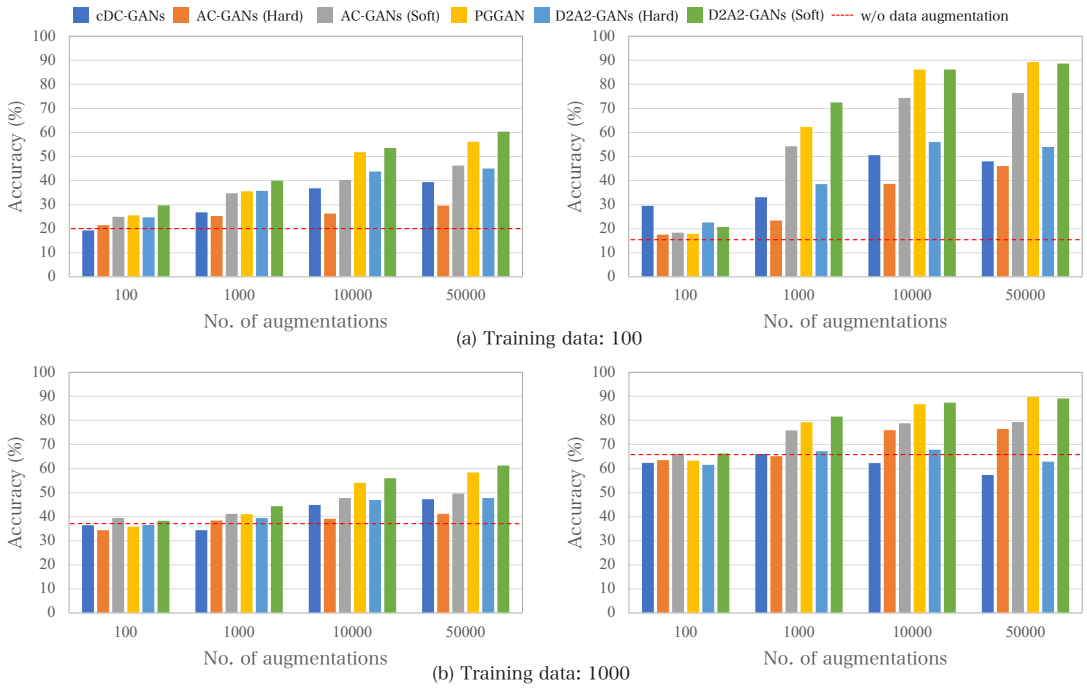


図 4.6: 各手法で生成した画像を用いて学習したモデルの識別精度比較．(a) および (b) 共に左が CIFAR-10，右が SVHN の認識精度を表している．点線はデータ増幅を用いずに学習したモデルの認識精度を表している．

4.3.3 先行研究との識別精度の比較

各手法の生成画像を増幅データとして用いて学習したモデルの識別精度を図 4.6 に示す．まず，ベースとなる学習データが 100 サンプルの CIFAR-10 の識別精度に着目すると，提案手法の Hard target の精度は，全ての増幅数で ACGAN の生成画像に Soft target を用いて学習した時と同程度である．また，ACGAN の生成画像に Hard target を用いた結果は，cDCGAN よりも低い事が確認できる．これは，ACGAN の生成画像は cDCGAN の生成画像よりもエントロピーが高いにも関わらず Hard な教師信号を用いたことに起因する．そのため，Soft target を用いるだけで性能が cDCGAN よりも良い精度を達成する．一方，PGGAN の結果は増幅数が 1000 サンプルの時に提案手法に Hard target を付与した結果を上回るが，提案手法に Soft target を利用した結果に劣る結果である．我々の提案手法は，Soft target の付与が性能向上に貢献しており，画質が最も優れていた PGGAN に勝る結果を達成することができる．

学習データが 100 サンプルの SVHN の結果は，先に述べた傾向がより顕著に現れている．ACGAN に Soft target を用いた結果は，Hard target よりも性能が劇的に改善しており，提案手法の Hard target の性能よりも高い．しかしながら，提案手法の生成画像に Soft target を付与して学習することによって，増幅数が 1,000 サンプル以上の時に最も良い性能で，生成画像の増幅なしから約 70 ポイントの精度向上を達成した．しかしながら，提案手法の生成画像に Soft target を付与して学習することで，

表 4.2: Ablation study.

	MSE		Acc.(%)	
	SVHN	CIFAR-10	SVHN	CIFAR-10
Baseline	0.0419	0.0437	88.9	60.34
w/o $\mathcal{L}_{att}$	0.0885	0.1006	89.7	52.7
w/o $\mathcal{L}_{cls}$	0.0483	0.0487	6.1	10.0

増幅数が 1,000 サンプル以上の時に優れた性能を実現しており，生成画像の増幅なしから約 70 ポイントの性能向上を達成した．PGGAN と Soft target を用いた提案手法の結果を比較すると同程度である．

ベースとなる学習データの数が 1,000 サンプルに増加したときは，100 サンプルの時と比較して，SVHN と CIFAR-10 共に生成画像でデータ増幅をしていない結果からの精度向上が少ない．ただ，2 つのデータセットともに，提案手法に Soft target を用いることで飛躍的な性能を改善することができる．特に，SVHN の結果に着目すると，提案手法に Hard target を付与すると，増幅データ数が増加するつれて性能が劣化する．一方，Soft target を付与することで精度が単調増加して先行研究や増幅なしの精度よりも良い結果を達成した．PGGAN と Soft target を用いた提案手法の結果は全ての増幅数において同程度であるが，PGGAN は提案手法より緻密なモデル設計によって高精細な画像生成を実現している．実際，表 4.1 から提案手法の画質は PGGAN に劣ることがわかる．

以上の結果から，提案手法に限らず Soft target を用いて学習することによって精度向上する事が判明した．ただ，提案手法は物体の特徴的な領域を捉えた画像生成であるため，特徴が捉えやすく Soft target で問題設定を簡略化することでデータ増幅として大きな貢献をしたと言える．さらに，提案手法より鮮明な画像の生成が可能な手法であっても，Soft target を用いることで同程度または勝る結果を実現することが可能である．

4.3.4 Ablation study

注視領域に関する一貫性損失 $\mathcal{L}_{att}$，および Attention branch のクラス識別に関する損失 $\mathcal{L}_{cls}$，それぞれを提案手法から切除して学習することで各損失計算の貢献度を CIFAR-10 と SVHN を用いて調査する．定量的評価指標は，生成画像を増幅データとして学習した ResNet-18 の識別精度と，以下に示す平均二乗誤差 (MSE) を注視領域に対する評価指標として使用する．

$$\mathrm{MSE} = \frac{1}{N}\|M(\boldsymbol{x}) - \hat{M}(\boldsymbol{x})\|_2^2 \tag{4.10}$$

EMSE は，ABN の注視領域 $\hat{M}(\boldsymbol{x})$ を基準として，Discriminator の注視領域 $M(\boldsymbol{x})$ が優れているかどうかを測る指標である．ここで，$M(\boldsymbol{x}), \hat{M}(\boldsymbol{x}) \in [0,1]^{N\times H\times W}$ で，N はサンプル数である．ResNet-18 は学習データ 100 サンプルをベースとして，50,000 サンプルの生成画像で増幅して学習した．また，MSE は全ての推論データを使用して導出した．それぞれの評価結果を表 4.2 に示す．

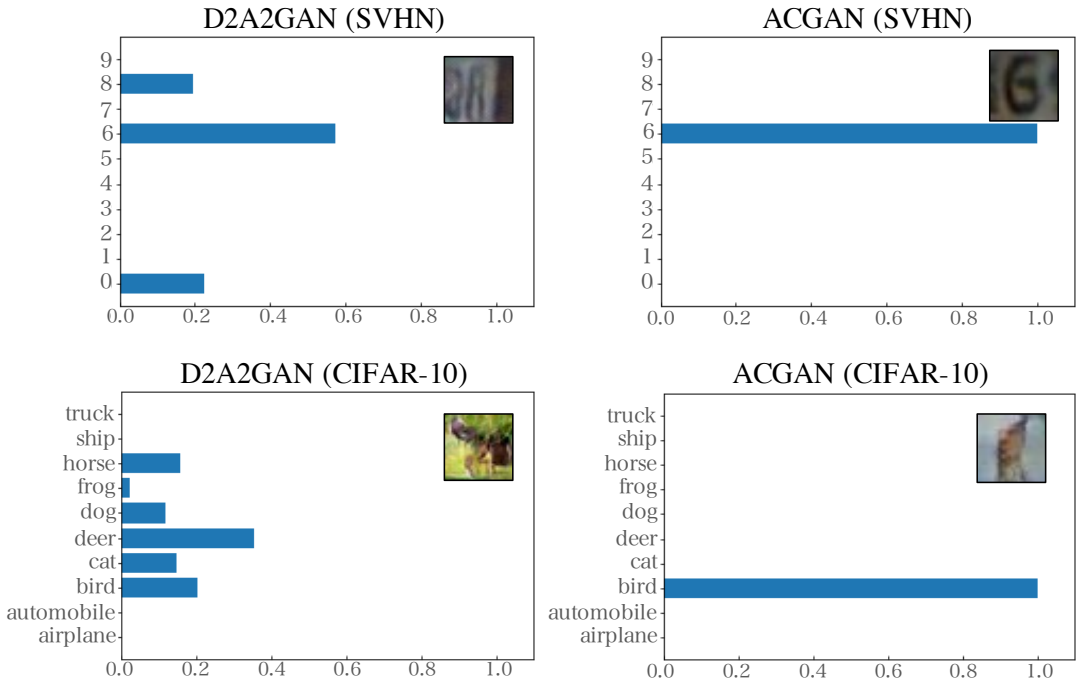


図 4.7: D2A2GAN と ACGAN，それぞれの生成画像に関する Discriminator の事後分布．

$\mathcal{L}_{att}$ が損失として含まれていない提案手法は，MSE が SVHN および CIFAR-10 ともに倍以上の数値となることから，Discriminator でうまく注視領域を獲得できていない．識別精度は，SVHN は同程度の値であるが，CIFAR-10 の精度は若干の低下を確認した．この結果は，SVHN が CIFAR-10 よりも識別対象が単純であるため，特徴的な領域が捉えれなくても精度が保証されることを示唆している．一方，CIFAR-10 は識別対象が複雑であるため，特徴的な領域をうまく生成するような注視領域を利用することが重要だと言える．

$\mathcal{L}_{cls}$ なしで学習した提案手法は，MSE が Baseline の値と同程度である．しかしながら，$\mathcal{L}_{cls}$ を切除することは，クラス識別を Discriminator から切除することに等しいため，識別精度の劇的な低下を確認した．これは，クラス識別を切除したことで，入力した条件に対応した画像を得ることが困難になり，生成画像に偏りが生じているためであると考える．実際，識別対象が単純な SVHN が CIFAR-10 よりも低い精度であることは，先に述べた理由に起因すると考える．

以上より，$\mathcal{L}_{att}$ および $\mathcal{L}_{cls}$ の切除は共に，生成画像を増幅データとして学習したモデルの性能劣化を引き起こすきっかけとなる事がわかった．

4.3.5 Soft label に関する考察

ACGAN と提案手法の生成画像ともに，Hard target よりも Soft target の結果の方が良い．また，提案手法は ACGAN よりも著しく性能向上している．2 つの手法の Soft target と識別精度の関係を調査するために，我々は各手法の Discirminator から獲得する，任意のサンプルに関する予測確率を分析

する．

ACGAN および提案手法の事後確率を図 4.7 に示す．図から確認できるとおり，ACGAN の事後確率は限りなく onehot ベクトルに近いことが確認できる．一方，提案手法は SVHN および CIFAR-10 ともに正解クラス以外にも，類似したクラスに確率が分布している事が確認できる．これは，それぞれの Discriminator 内のクラス識別器の設計方法に起因している．ACGAN は，Discriminator の最終層に全結合を採用して, 真贋判定とクラス識別をするため，正解クラス以外のユニットが発火しないように重みパラメータを更新する．従って，教師信号として用いた onehot ベクトルに限りなく近似する．

一方，提案手法の Discriminator は Attention branch 内で GAP を用いてクラス識別するため，特徴マップの平均値が最終的な予測確率となる．つまり，最終的な識別結果に直結する重みパラメータが存在しないため，ACGAN よりも Soft な分布を獲得する事ができる．

以上より，たとえ実画像に忠実な画像を生成したとしても，生成画像には若干の曖昧さが生じる．そのため，類似したクラスに対しても確率が生じている提案手法の Soft target の方が教師信号として優れていると言える．

4.4 まとめ

本章では，識別時の注視領域を考慮した画像生成が可能な Discriminator-Driven Attention-Aware GAN (D2A2GAN) を提案した．提案手法は，ベースとなる学習用データが少ない時に生成画像に対して onehot ベクトルの教師信号を付与して学習することで先行研究と同程度の精度を達成した．また，提案手法の Attention branch から得た予測分布を教師信号として学習した場合，先行研究より著しい性能向上が確認できた．先行研究である ACGAN においても同様に Discriminator が出力した予測分布を教師信号として学習したところ，提案手法に及ばない結果であることが判明した．

ACGAN と提案手法の予測分布を分析したところ，ACGAN はネットワークの設計上 onehot ベクトルに限りなく近い分布が出力されることを確認した．一方，提案手法は類似したクラスに対しても確率が割り当てられるような予測分布であるため，ラベルスムージングのような正則化効果が暗黙的に導入され，性能向上につながったと考えられる．

提案手法は増幅データとして優れた性能を発揮できるが，GAN の生成画像は訓練データが十分なとき学習に悪影響を与えることが知られている [139]．また，GAN による画像生成は CIFAR-10 程度のサンプル数が必要となるため，限られたサンプル数のデータセットを学習し，データセットを増幅することが困難である．したがって，今後の課題として，限られたデータ数でうまく画像生成が可能な学習方法の提案や先行研究 [140] との組み合わせが挙げられる．

第5章

敵対的サンプルの多様性を増強した敵対的学習

CNN において優れた画像分類を実現するためには，データ増幅 [16, 133, 134] によってデータの量とバリエーションを増幅して学習することが一般的である．これによって，CNN は入力画像に回転や並進移動，照明変化などの自然発生的なノイズに対して頑健になるが，adversarial examples として名を馳せている悪意のある摂動によって変化させられた画像 [10] に脆弱である．adversarial examples は画像分類だけでなく，物体検出や距離画像推定 [141]，セマンティックセグメンテーション [142] においても誤った推論を誘発できる．この摂動による微小な変化は，一般的に人間には知覚困難であるため，CNN をベースとしたアプリケーション (自動運転車両やマルウェア検出) のセキュリティの脅威となる．そのため，数多くの防御策が提案されている [61, 143, 110, 111, 2, 15]．その中でも，Adversarial Training [11] は CNN の脆弱性を改善するための手法としてポピュラーかつ効果的な手法である．Adversarial Training は摂動画像をベースにモデルのパラメータを更新することによって，adversarial attack に対してロバストなモデルを構築することができる．しかし，CNN は Adversarial Training 中に厳しい摂動を付与しながら学習をすることで，摂動に対する防御性能が向上する一方，摂動がない通常サンプルに対する分類精度が著しく劣化する．この頑健性と分類精度のトレードオフに関して，多くの研究者によって様々な観点から理論的および経験的な証明が進められている．

Schmidt ら [14] は頑健性のためには標準的な学習よりも膨大で複雑なデータが必要であることを理論的に証明した．この理論に基づいて，教師信号がないデータを大量に集めて教師なし学習を行うことで，従来の Adversarial Training よりも性能向上することが実験的に示されている [109, 108]. Yin ら [144] は adversarial examples，ガウスノイズやコントラスト変化や fog を適用したデータ増幅，clean なデータを周波数解析することで，問題を経験的に示した．Yin ら [144] によると，adversarial examples は通常の画像よりも高周波成分が多く含むため，通常の画像と捉える周波数が異なり，トレードオフが発生すると主張した．Tsipras ら [145] は，adversarial training と通常の学習でモデルがとらえる特徴量が異なること様々な実験によって示した．Tsipras ら [145] の結果は周波数帯域の違いに由来していると考えられる．Lee ら [15] は Adversarial Training によってネットワークの重みが予期せぬ方向に最適化される Adversarial Feature Overfitting (AFO) を示し，AFO 問題点の回避策として AVmixup を提案した．

これら先行研究の主張をまとめると，トレードオフの発生は狭義なカバレッジを持つデータ集合を用いた学習によって起こるロバスト過適合だと断定できる．そこで，本研究ではこの問題をモチベーションとして，通常の分類精度を保ちつつ，優れた頑健性を獲得できる Adversarial Training を提

案する．AVmixup は仮想的に定義した摂動との mixup [16] および label smoothing [56] を利用することで，モデルに正則化効果を導入しつつ学習データのカバレッジを拡張して過適合を予防した．Lee ら [15] は AVmixup の効果に対する分析をしていないが，Guo ら [146] によって mixup したサンプルは元の多様体と異なる多様体へ写像されることが示されている．AVmixup は優秀な性能を発揮しているが，Adversarial Training によって訓練する摂動画像は単に係数倍したものであるため限定的である．そのため本研究では，学習中の摂動画像のバリエーションをさらに豊富にするために，画像内で摂動の強度に不均一さを持たせた Masking and Mixing Adversarial Training (M^2AT) を提案する．提案手法は以下に示す 2 つの工程を経て adversarial examples を定義する．

1. 矩形内外が摂動画像となるように定義したバイナリマスクによる摂動の **Masking**.
2. 一部だけ摂動画像の 2 つを任意の内挿比を用いて **Mixing**.

adversarial examples に教師信号として通常のラベルスムージングを使用することは，クラス数の増加に伴ってしばしば冗長な表現となる．そこで，通常サンプルと adversarial examples それぞれの予測分布を分析して傾向調査することで，Adversarial Training に適したラベルスムージングを提供する．本研究の貢献をまとめると以下の通りである．

- 経験的な実験によってラベルスムージングの特性および，adversarial examples を CNN に与えた時の振る舞いを明らかにし，新たなラベルスムージング方法を提供する．
- 訓練中の adverarial examples のバリエーションを先行研究よりもリッチすることで，精度と頑健性のギャップ緩和ができる強力な Adversarial Training を提案する．
- CIFAR-10 データセットを用いた実験において，先行研究を遥かに上回る最先端の性能を達成したことを報告し，提案手法の興味深い現象について議論する．

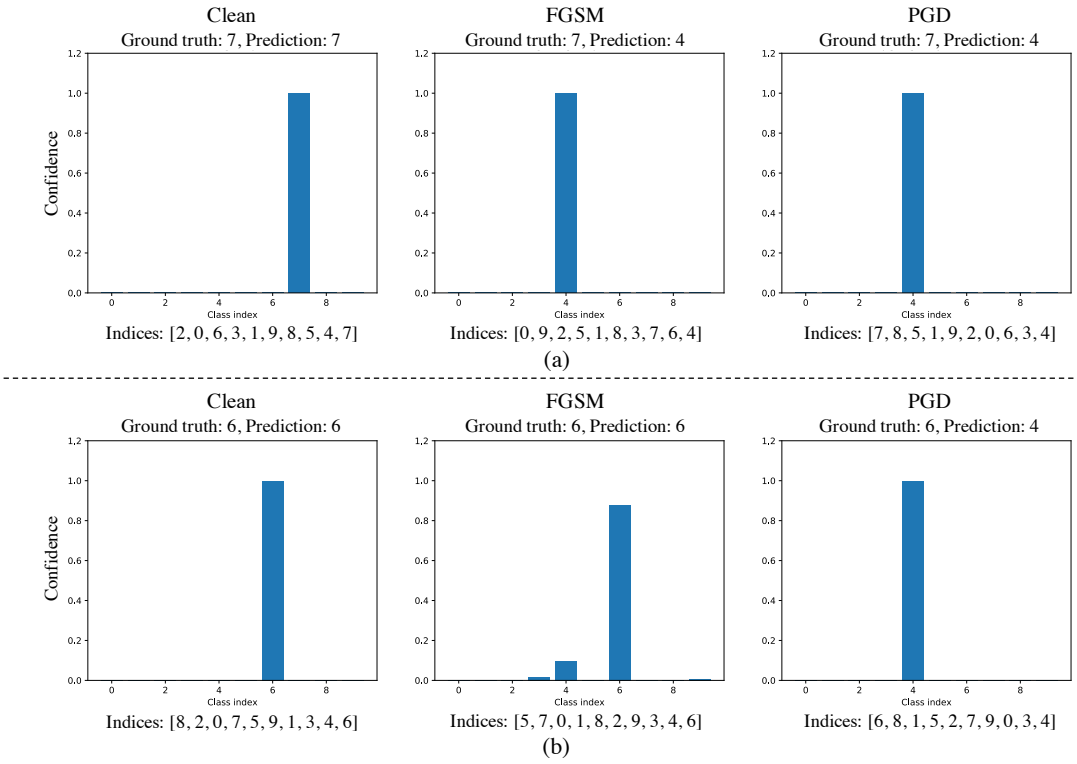


図 5.1: 通常のサンプル，FGSM と PGD で求めた adversarial examples，それぞれをモデルに入力して得られる予測分布と予測結果．

5.1 予測分布の分析

まず,「分類器は adversarial examples をどのようなルールで不正解クラスとして誤分類するのか？」ということが純粋な疑問点として挙げられる．Zhang ら [83] は識別境界付近のサンプルが微小な摂動によって不適切な方向へ移動するため，線形変換した先の特徴量に齟齬が生じることを主張した．この主張は 2 クラス分類に限定すれば直感的に解釈できるが，多クラス分類時は解釈が困難になる．したがって，通常サンプルと adversarial examples に対する予測分布を分析することで，2 つのサンプル間の関係性を明らかにする．

図 5.1 は CIFAR-10 で学習したモデルに摂動なしのサンプル，FGSM による adversarial examples，PGD による adversarial examples を入力して得られる予測分布と予測結果を示している．教師信号と予測クラスそれぞれのインデックスは各グラフ上部に記している．また，各グラフ下部にはそれぞれのサンプルをモデルに入力した際に予測されたクラス順位を昇順にして示している．図 5.1(a)(b) はそれぞれ，FGSM と PGD 共に誤分類が生じた例と PGD のみに誤分類が生じた例を表している．

図 5.1(a) に示すように，FGSM や PGD によって求めた adversarial examples は，主に通常サンプルを分類したときの 2 番目に確率が高いクラスと誤分類する傾向にあることが確認できる．この結果は，ϵ-ball 内に内在している識別境界付近のサンプルが，微小な摂動によって隣接クラスに移動させ

られたと捉えることができる．対照的に，正解クラスを除いた全てのクラス確率が限りなく等しい場合，敵対的攻撃のリスクを軽減することができる．

図 5.1(b) に示すように，FGSM は 1 ステップで摂動を求めるため，しばしば攻撃に失敗することがあるが，PGD を用いることで 2 番目に高いクラスに騙されるような摂動を求めることができる．これは，予測分布において正解クラスを除くどこかのクラス確率が僅かに高いため，PGD を用いた慎重な摂動探索により適切に誤分類させる摂動が求めらる．

これらの結果に基づくと，モデルは一様分布を用いて予測分布の正解クラス以外の確率を平坦にすることで，敵対的攻撃に対する頑健性が向上するといえる．実際，Fu ら [147] はラベルスムージングを用いた学習によって，いくつかの敵対的攻撃を防御できることを示している．以上より，優れた頑健性を獲得するためには Soft label などを用いてモデルの出力分布を平坦にすることが重要であると強く主張する．

5.2 提案手法

本章では，Masking and Mixing Adversarial Training ($\mathrm{M}^2\mathrm{AT}$) を提案し詳細を述べる．$\mathrm{M}^2\mathrm{AT}$ は，学習中にバリエーション豊富な adversarial exmaples を学習し，正解クラスを除いた確率分布をフラットにすることが目的である．この学習方法によって，通常の分類精度を維持しながら，優れたモデルの頑健性を獲得する．

5.2.1 Overview

提案手法は Madry ら [11] と同様で，式 (2.21) に示すように，任意の回数の反復によって摂動 $\boldsymbol{\delta} \in \mathbb{R}^{3\times h\times w}$ をモデルの勾配から作成する．求めた摂動は従来のようにサンプルに直接付与してアピアランスを微小に変動させるのではなく，提案する 2 つの処理を介して摂動を付与する．

まず，求めた摂動はバイナリマスクによって一部のみ摂動を抜き出し，対象の画像へ付与する．つまり，この工程を通じて作成する adversarial examples は，画像中の一部のみ微小な変動が適用されていることになる．バイナリマスクを反転させて同様の処理を行い，2 つの一部分のみ摂動による変化施された画像を得る．バイナリマスクの定義や摂動付与の詳細は 5.2.2 で数式を交えて詳細に述べる．さらに，クラス数が増加した際の冗長な表現を回避するためのラベルスムージングを導入する．

次に，矩形内外が微小な摂動によって変化させられた，2 つのサンプルを任意の比率で線形補間し，1 つの adversarial examples を作成する．先行研究では，求めた摂動を対象のサンプルに直接加算するが，提案手法は 2 種類の強度の摂動が混在するように adversarial examples を作成するため，モデルが学習する摂動のバリエーションを増幅することができる．この adverarial examples に対する教師信号は，一部のみ変動が施された画像に対して定義した教師信号を，さらに線形補間することによって定義する．これらの処理の詳細は 5.2.3 で述べる．

最後に，図 5.2 に $\mathrm{M}^2\mathrm{AT}$ の adversarial examples 作成手順の概念図，Algorithm 1 に提案手法の処理

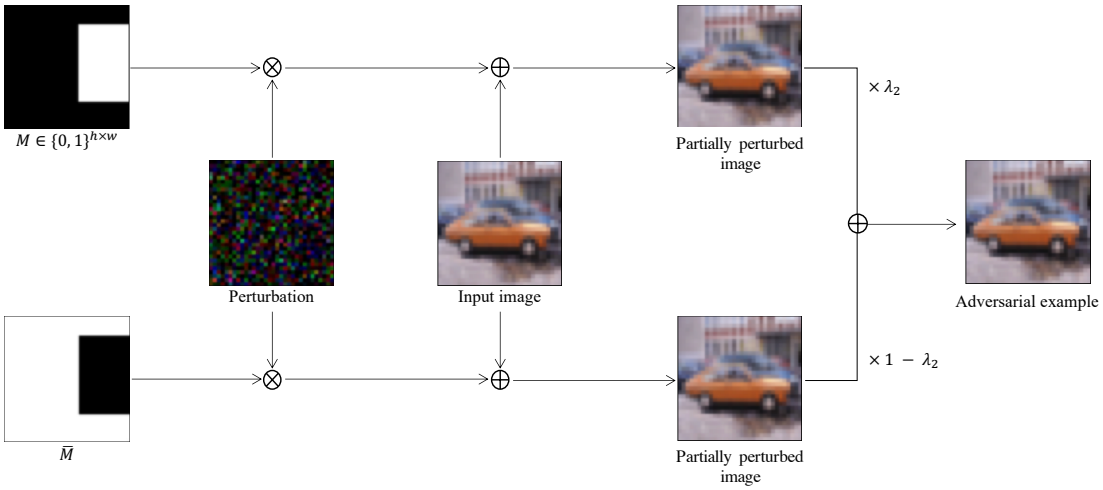


図 5.2: M^{2}AT を用いた adversarial examples の作成のコンセプト図.

の疑似コードを示す．図 5.2 において，M はバイナリマスク，$\bar{M} = 1 - M$，$\otimes$ と $\oplus$ はそれぞれ，要素積と要素ごとの和を表している．また，λ_2 は $\mathrm{Beta}(\alpha, \alpha)$ からサンプリングした内挿比である．

5.2.2 Masking phase

摂動をマスクする段階では，まず，バイナリマスク $M \in \{0,1\}^{h \times w}$ を用いて式 (2.21) で求めた摂動 $\boldsymbol{\delta}$ の一定領域を抽出し，サンプル $\boldsymbol{x}$ に付与することで一部のみ摂動された画像を作成する．この処理をバイナリマスクの反対，つまり $1 - M$ の場合も行う．これにより，M を用いた場合は矩形内，$1 - M$ を用いた場合は矩形外が摂動されることになる．これらの処理は以下の式で表される．

$$\boldsymbol{\xi} = \boldsymbol{x} + \boldsymbol{\delta} \odot M \tag{5.1}$$

$$\bar{\boldsymbol{\xi}} = \boldsymbol{x} + \boldsymbol{\delta} \odot (1 - M) \tag{5.2}$$

バイナリマスク M の矩形の大きさ $B = (r_{x_1}, r_{y_1}, r_{x_2}, r_{y_2})$ は，CutMix [133] と同様に，一様分布 $U[0,1]$ からサンプリングした任意の確率 λ_1 を用いて以下のように決定する．

$$r_{x_1} \sim U[0, W], \ r_{x_2} = \min\left(W, W\sqrt{1-\lambda_1} + r_{x_1}\right) \tag{5.3}$$

$$r_{y_1} \sim U[0, H], \ r_{y_2} = \min\left(H, H\sqrt{1-\lambda_1} + r_{y_1}\right) \tag{5.4}$$

ミニバッチ内で λ_1 を一様に等しくサンプリングしたとしても，矩形の幅と高さ r_{x_1} と r_{y_1} がランダムに決定されるため，ミニバッチ内の全てサンプルで異なる矩形位置と大きさが求められる．求めた矩形の座標を用いて，以下の式で最終的なバイナリマスク M を獲得する．

$$M = \begin{cases} 1 & \text{if } r_{x_1} < M_{:,j} < r_{x_2}, r_{y_1} < M_{i,:} < r_{y_2} \\ 0 & \text{otherwise} \end{cases} \tag{5.5}$$

Algorithm 1 Masking and Mixing Adversarial Training

Require: Training dataset $\mathcal{D}$, batch size n, training epochs T, learning rate η, model parameter $\boldsymbol{\theta}$, hyper-parameter of beta distribution α

Require: The function deriving adversarial perturbation $\mathcal{A}$

Require: Masking function ϕ

for $t = 1, \ldots, T$ **do**
  for $\{\boldsymbol{x}_i, \boldsymbol{y}_i | i = 1, \ldots, n\} \sim \mathcal{D}$ **do**
    $\hat{\boldsymbol{x}}_i \leftarrow \mathcal{A}(\boldsymbol{x}_i, \boldsymbol{y}_i; \boldsymbol{\theta})$
    $\boldsymbol{\delta}_i \leftarrow \hat{\boldsymbol{x}}_i - \boldsymbol{x}_i,\ \lambda_1 \sim U[0, 1]$
    data masking and label smoothing phase:
    $\boldsymbol{\xi}_i, \bar{\boldsymbol{\xi}}_i, \boldsymbol{t}_i, \bar{\boldsymbol{t}}_i \leftarrow \phi(\hat{\boldsymbol{x}}_i, \boldsymbol{\delta}_i, \boldsymbol{y}_i, \lambda_1),\ \lambda_2 \sim \mathrm{Beta}(\alpha, \alpha)$
    data mixing phase:
    $\tilde{\boldsymbol{x}}_i \leftarrow \lambda_2 \boldsymbol{\xi}_i + (1 - \lambda_2)\bar{\boldsymbol{\xi}}_i$
    $\tilde{\boldsymbol{y}}_i \leftarrow \lambda_2 \boldsymbol{t}_i + (1 - \lambda_2)\bar{\boldsymbol{t}}_i$
    model update:
    $\boldsymbol{\theta}_{t+1} \leftarrow \boldsymbol{\theta}_t - \eta \cdot \frac{1}{n} \sum_{i=1}^{n} \nabla_\theta \mathcal{L}(\tilde{\boldsymbol{x}}_i, \tilde{\boldsymbol{y}}_i; \boldsymbol{\theta}_t)$
  end for
end for
return model parameter $\boldsymbol{\theta}$

ここで，$r_{x_1} < M_{:,j} < r_{x_2}$ は範囲内の x 軸方向の要素に 1 を代入することを表しており，$r_{y_1} < M_{i,:} < r_{y_2}$ は y 軸方向に同様の処理を行うことを表している．

式 (5.1) および式 (5.2) で求めたそれぞれの一部摂動画像に対する教師信号は，矩形の大きさを決定する時に用いた λ_1 ではなく，摂動されてない領域と摂動領域の面積比 $\lambda_1' = \frac{(r_{x_2} - r_{x_1}) \times (r_{y_2} - r_{y_1})}{H \times W}$ を平滑化パラメータとして，次式のラベルスムージングによって決定する．

$$\boldsymbol{t} = \lambda_1' \boldsymbol{y} + \bar{\boldsymbol{y}}(1 - \lambda_1')\boldsymbol{s} \tag{5.6}$$

$$\bar{\boldsymbol{t}} = \bar{\boldsymbol{y}}\lambda_1' \boldsymbol{s} + (1 - \lambda_1')\boldsymbol{y} \tag{5.7}$$

ここで，$\boldsymbol{y}$ は正解クラスが 1 の onehot ベクトル，$\bar{\boldsymbol{y}} = 1 - \boldsymbol{y}$ は正解クラスを除くクラスが全て 1 のベクトルを表している．

通常のラベルスムージングは全てのクラスに等しく確率を割り当てるため，クラス数が増加した場合，1 クラスに割り当てられる確率が小さくなり，しばしば冗長な表現となる．この冗長な表現を避けるために，提案するラベルスムージングでは確率を割り当てるクラスを限定する．クラスを限定する方法として，以下の 2 つを提供する．

1. 正解クラス $\boldsymbol{y}$ を除いたクラスからランダムに決定する．
2. Algorithm 1 中の $\hat{\boldsymbol{x}}$ を入力した時の予測確率が低いクラスに確率を割り当てる．

確率を割り当てるクラスインテックス集合を $\mathcal{M}$，クラス数を K とすると，提案手法のラベルスムージングで使用する確率分布 $\boldsymbol{s}$ は次式によって表される．

$$\boldsymbol{s}_{\{i=0,\dots K\}} = \begin{cases} \frac{1}{|\mathcal{M}|} & i \in \mathcal{M} \\ 0 & \text{otherwise} \end{cases} \tag{5.8}$$

このような確率分布とラベルスムージングすることによって，冗長な表現を避けつつ，従来のラベルスムージングと同様の効果が期待できる．

5.2.3 Mixing phase

AVmixup は通常のサンプルと任意の係数を乗算した摂動である敵対的頂点と内装することによって adversarial examples を作成する．しかし，画像全体の摂動強度は求めた摂動そのものである，つまり直線的なバリエーションのみ扱うため，摂動画像のバリエーションという観点で限定的である．

我々は adversarial examples のバリエーションを増強することによってトレードオフの解消を試みるため，Masking phase で求めた 2 つのサンプルを合成して 1 つの adversarial examples を作成する必要がある．そのため，画像を混ぜ合わせる段階では，AVmixup や mixup を参考にして，任意の内挿比で 2 つの一部のみ摂動された画像を混合することで画像全体を摂動する．ここで，提案手法を通じて獲得できる adverarial examples は，1 つの画像内に 2 つの強度の摂動が内在した画像となることに注意されたい．

$\lambda_2 \sim \mathrm{Beta}(1,1)$ を内挿比とすると，mixup したサンプル $\tilde{\boldsymbol{x}}$ とそれに対する教師信号 $\tilde{\boldsymbol{y}}$ は次式によって求められる．

$$\tilde{\boldsymbol{x}} = \lambda_2 \boldsymbol{\xi} + (1-\lambda_2)\bar{\boldsymbol{\xi}} \tag{5.9}$$

$$\tilde{\boldsymbol{y}} = \lambda_2 \boldsymbol{t} + (1-\lambda_2)\bar{\boldsymbol{t}} \tag{5.10}$$

この処理を通じて作成した adversarial examples に関して，$\lambda_1 = 0$ または $\lambda_1 = 1$ の時，AVmixup の $\gamma = 1$ と等価な処理となる．そのため，AVmixup よりもバリエーション豊富な Adversarial examples を学習することが可能となり，更なる頑健性能向上が期待できる．

5.3 評価実験

本章では，提案手法の防御性能を評価するために，いくつかの従来手法と性能比較する．本実験では，データセットとして CIFAR-10 と CIFAR-100 を使用する．CIFAR-10 は，32×32 の RGB で，50,000 の学習用サンプルおよび 10,000 の推論用サンプルを有する 10 クラスの自然画像データセットである．各クラスに対するサンプル数は，学習用として 5,000 サンプル，推論用として 1,000 サンプル用意されている．一方，CIFAR-100 は，各クラスのサンプル数が学習用として 500 サンプル，推論用として 100 サンプル用意されていることを除いて，CIFAR-10 と同じである．

5.3.1 実験の詳細な設定

本実験では，CIFAR-10 および CIFAR-100 ともに WRN34-10 (Wide Residual Networks) [148] を使用して学習する．128 のバッチサイズを用いて 200 エポック学習する．最適化関数は初期学習率が 0.1，モメンタムが 0.9，weight decay が 2.0×10^{-4} のモメンタム付き確率的勾配降下法 (momentum SGD) を使用する．CIFAR-10 において，学習率は学習回数の 50%と 75%で 1/10 に減衰する．CIFAR-100 においては，cosine annealing を用いて学習率をスケジューリングする．学習時のデータ増幅はランダムクロップと Horizontal flip を使用し，各ピクセルは $[0, 1]$ の範囲に収まるように正規化する．学習中の摂動は，反復回数が $k = 10$，摂動許容範囲が $\epsilon = 8$，ステップサイズが $\alpha = 2$ の PGD を使用して求める．摂動許容範囲 ϵ とステップサイズ α は，入力サンプルのスケールに合わせる必要があるため，それぞれをピクセルの最大値である 255 で除算する．

モデルの頑健性を評価するための敵対的攻撃は，FGSM [2]，PGD [11]，CW [68] を使用する．PGD-k や CW-k の k は反復回数を表している．提案手法との比較対象は以下に示すとおりである．

- Standard: 通常の学習データのみで学習したモデル．
- PGD: $k = 10, \epsilon = 8/255, \alpha = 2/255$ の PGD を用いた通常の Adversarial Training．
- PGD with LS: ラベルスムージングを用いた通常の Adversarial Training．スムージングパラメータは，学習中，$\mathrm{Beta}(1, 1)$ からサンプリングする．
- AVmixup: Lee ら [15] と同様の設定で学習した AVmixup．

CIFAR-10 において，提案手法は正解クラスを除く全てのクラスに一様に確率を割り当てた一様分布を用いてラベルスムージングする．言い換えると，この分布は式 (5.8) において，クラスインデックス集合の大きさが $|\mathcal{M}| = 9$ と等価な処理である．CIFAR-100 において，提案手法は正解クラスを除いて 10 クラスに確率を割り当てる．本実験において，ランダムにクラスを選択する提案手法を M^2AT (rand)，クラス確率が低いクラスから選択する提案手法を M^2AT (a.s.) と表記する．

5.3.2 精度比較結果

表 5.1 の上のブロックに CIFAR-10 の結果を示す．M^2AT は FGSM と PGD-10 および PGD-20 を用いた敵対的攻撃に関して劇的に頑健性が向上した．また，提案手法の通常の分類精度は我々の実装した AVmixup より 1 ポイント程度劣る結果であったが，AVmixup の論文値とは同程度の精度を達成した．CW の結果に関して，AVmixup の結果から約 3 ポイント程度しか頑健性が改善されなかった．一方，提案手法において，PGD-10 および PGD-20 ともに 80%を上回る結果であり，PGD の結果から約 30 ポイント，AVmixup の結果から約 20 ポイント頑健性が向上した．さらに，図 5.4 に示すように，M^2AT は AVmixup と同様でロバスト過適合を回避できることが確認できた．AVmixup はロバスト過適合が生じないと主張されているが，実装した AVmixup は論文値と同程度の性能にも関わらず，図 5.3 に示すように，ロバスト過適合が生じることを確認した．提案手法は CW 以外の攻撃

表 5.1: 通常サンプルに対する分類精度と敵対的攻撃に対する頑健性の比較．太字は最高性能の手法，$*$ は論文値の精度を引用していることを表している．

Dataset	Model	Clean	FGSM	PGD-10	PGD-20	CW-20
CIFAR-10	Standard	95.48	7.25	0.0	0.0	0.0
	PGD	85.83	58.66	52.09	50.80	30.16
	PGD with LS	86.33	61.67	55.87	54.78	30.36
	BAT [75]*	91.2	70.7	–	57.5	56.2
	AVmixup*	93.24	78.25	62.67	58.23	53.63
	AVmixup	94.81	80.28	69.29	65.01	54.8
	M^2AT (WRN28-10)	92.09	73.67	65.83	63.06	55.04
	M^2AT	93.16	**83.35**	**82.29**	**80.66**	**56.90**
CIFAR-100	PGD	61.29	46.01	–	25.17	–
	AVmixup*	74.81	62.76	–	38.49	–
	AVmixup	77.15	53.32	–	27.00	–
	M^2AT (a.s.)	67.76	43.05	35.62	33.80	–
	M^2AT (rand)	68.76	44.91	36.62	34.66	–

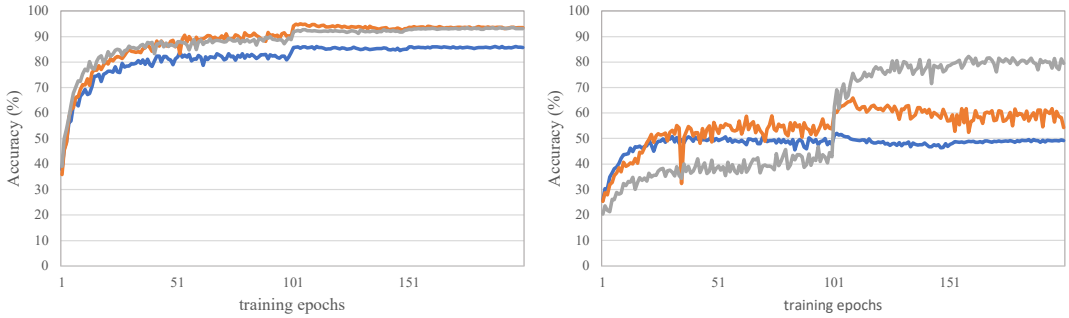
に関して，通常の分類精度と頑健性のギャップを劇的に改善できていることがわかる．また，ベースモデルを BAT [75] に合わせて学習した結果においても，通常の分類精度および頑健性が向上した．

表 5.1 において，提案手法除く全ての手法は FGSM と PGD-10 や PGD-20 の精度に 10 ポイント程度のギャップが生じている．対照的に，提案手法はそれらの精度差がほとんどないことが確認できる．この現象は非常に興味深いため，5.3.4 で詳細に議論する．

表 5.1 の下のブロックに CIFAR-100 の結果に着目すると，AVmixup の論文値を上回る結果を獲得することができなかったが，実装した AVmixup より高性能なモデルを獲得することができる．AVmixup の結果は，著者らによる CIFAR-10 を用いた実装[1] を CIFAR-100 に変更したとしても，論文値をうまく再現することができなかった．PGD の結果と提案手法を比較すると，FGSM はほとんど同じ精度であり，通常の分類精度と PGD-20 は 7 ポイントと 10 ポイント程度改善された．これは，提案手法が PGD より分類精度と頑健性の間のギャップを緩和するために適しているといえる．ラベルスムージングの違いに関して，昇順クラスから確率を割り当てる結果はランダムにクラスを決めるよりもわずかに劣る結果であった．これは，昇順クラスへ確率を割り当てる方法が学習終盤に向かって，固定されてランダム性が乏しくなることが原因である．

表 5.2 に TRADES との性能比較を示す．これは，AVmixup や提案手法によって学習するための摂動を作成して，TRADES の損失関数を最小化するように学習する実験である．提案手法における通常の分類精度は TRADES を用いた AVmixup に劣る結果であるが，PGD-20 対する頑健性は 10 ポイント程度向上した．さらに，提案手法は通常の分類精度と PGD による攻撃に対する頑健性の間の

[1]The official implementation of AVmixup: https://github.com/Saehyung-Lee/cifar10_challenge



(a) Clean data　　(b) Adversarial examples

図 5.3: 通常サンプルと adversarial examples に対する分類精度の推移．

表 5.2: CIFAR-10 における TRADES との性能比較．太字は最高性能を表している．

	Clean	PGD-20
PGD	87.3	47.04
TRADES ($1/\lambda = 1$)	88.64	49.14
TRADES ($1/\lambda = 6$)	84.92	56.61
AVmixup	**90.36**	58.27
M^2AT	89.35	**69.76**

ギャップを AVmixup よりも緩和することを確認した．

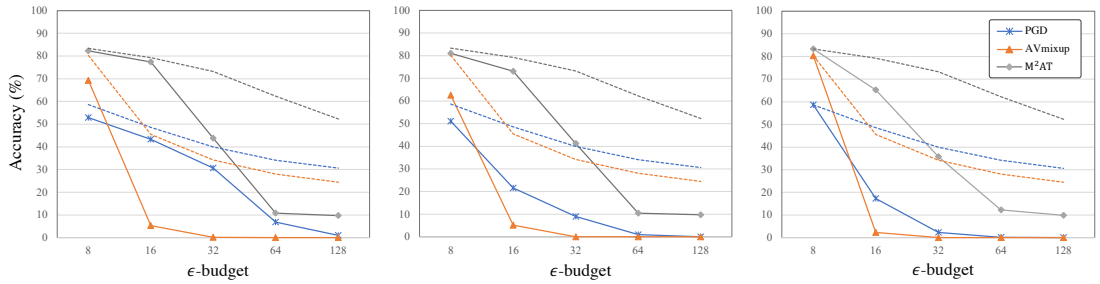
5.3.3 Ablation study

表 5.3 は提案手法からマスク処理 (Masking) と 2 つのデータの混合処理 (Mixing)，ラベルスムージング (LS) を切除したときの全ての組み合わせにおける性能を示している．各モデルは**??**で述べた実験設定を使用して学習しており，ラベルスムージングのパラメータは $\mathrm{Beta}(1, 1)$ からサンプリングした値とする．

表 5.3 より，ラベルスムージングを使用して学習するだけでも，表 5.1 で示した AVmixup と同程

表 5.3: 提案手法の Ablation study．太字は最高性能を表している．

Masking	Mixing	LS	Clean	FGSM	PGD-20
		✓	86.33	61.67	54.78
	✓		89.60	54.75	44.70
	✓	✓	**93.97**	74.81	60.96
✓			89.92	56.36	43.72
✓		✓	93.36	65.80	42.46
✓	✓		90.21	60.14	49.25
✓	✓	✓	93.16	**83.35**	**80.66**



(a) $\alpha = 2.0$ (b) $\alpha = 4.0$ (c) $\alpha = 8.0$

図 5.4: 各摂動許容範囲 ϵ を用いた adverarial examples に対する精度の推移．FGSM は全てのグラフで同じ結果である．

度の精度を獲得できた．ラベルスムージングとデータの混合処理を組み合わせることで，PGD-20 の頑健性は AVmixup の論文値を上回る精度を達成した．この結果は，スムージングパラメータが変動値であることを除いて，$\gamma = 1$ の AVmixup と等価な学習である．

バイナリマスクを用いて一部だけ摂動した adversarial examples をそのまま学習すると，頑健性は表 5.1 の PGD の結果に劣るものの，通常の分類精度が僅かに向上した．この結果は，摂動された領域と摂動されてない領域の異なる周波数をうまく吸収することで，通常の分類精度の低下を緩和できることを示唆している．adversarial examples と通常のサンプルを線形補間することに対しても同様のことが言える．

一部の領域のみ摂動した画像はラベルスムージングと併せて学習することで，通常の分類精度と FGSM に対する頑健性が向上するが，PGD-20 の結果は同程度である．一方，2 つの一部摂動画像を線形補間して 1 つの adversarial examples にまとめることで，PGD に対する頑健性は向上するがその他の精度が劣化することが表 5.3 確認できる．まとめると，これらの結果は全ての処理を利用した提案手法が最も最適な学習ができることを示している．

5.3.4 Additional discussion

M^2AT における FGSM と PGD の関係　一般的に，PGD を用いて求める adversarial examples は ϵ-ball の空間内を慎重に探索するため，FGSM よりも強い摂動を用いて攻撃されることになる．したがって，これら 2 つの攻撃に対する頑健性の間には大きなギャップが生じる．一方，5.3.2 で述べたように，提案手法は FGSM と PGD に対する頑健性が同程度になる．この現象を解釈するために，モデルを攻撃する際の摂動許容範囲 ϵ を大きくしたときの精度推移を調査する．

図 5.3 は，各摂動許容範囲を用いて FGSM と PGD-10 それぞれで求めた adversarial examples に対する分類精度を表している．図 5.3 の全てのグラフにおいて，破線が FGSM，実線が PGD-10 を表している．AVmixup は $\epsilon = 8$ を用いたとき優れた性能を達成できるが，$\epsilon = 8$ よりも大きな摂動許容範囲では FGSM と PGD-10 に対する頑健性がともに劇的に劣化することが確認できる．これは AVmixup が学習中に用いた摂動許容範囲に依存しているため，より広い空間で定義された強い摂動に脆弱と

表 5.4: PGD-20 を用いた Transfer ベースのブラックボックスアタックの結果．横の手法は攻撃モデル，縦の手法は防御モデルを表している．

Defense model	Attack model		
	PGD	PGD with LS	AVmixup
PGD	–	50.83	50.86
PGD with LS	54.89	–	54.86
AVmixup	64.86	64.76	–
M^2AT	80.77	80.62	80.82

なる．PGD において，著しい劣化はなかったが，AVmixup と同様で大きな摂動許容範囲に対して脆弱である．対照的に，提案手法は全ての節度許容範囲において多手法よりも優れた性能を維持できる．さらに，提案手法において $\epsilon = 32$ までの性能は著しい劣化することなく，高い頑健性を維持できることが確認できた．つまり，AVmixup は学習中に使用した摂動許容範囲外の空間に損失が最大となるポイントが多く存在していると考えられる．

Black-box attack 最後にモデルの Black-box な攻撃に対する頑健性を調査するために，任意の手法で学習したモデルの摂動を用いて別のモデルを攻撃する．white-box な攻撃はモデルの重みパラメータや学習設定など全て公開された状態の攻撃であるため，非現実的である．そのため，攻撃対象のパラメータが全て未知の状態に対して頑健であることが重要となる．表 5.4 に示すように，提案手法は別のモデルで求めた摂動に対して最も優れたパフォーマンスを達成できた．従って，我々の手法を用いて学習したモデルは自分自身以外のモデルの勾配から作成した摂動からの影響を受けづに安定して画像分類を行うことができる．

5.4 まとめ

本章では，バイナリマスクを用いて摂動を 2 つに分解して，任意の内挿比で混合する Masking and Mixing Adversarial Training (M^2AT) を提案した．さらに，クラス数が増加した際のラベルムージングは冗長な表現になるため，この問題を回避するための新たなラベルスムージングを提案した．提案手法は CIFAR-10 データセットにおいて，従来法よりも劇的に頑健性を向上させるだけでなく，通常の分類精度の維持も可能とした．しかし，CIFAR-100 データセットでは，従来法に匹敵する頑健性が得られたものの，CIFAR-10 のような著しい性能向上は達成できなかった．これは，提案手法において摂動を分解して任意の比率で合成した adversarial examples が，分解する前の摂動と異なるため，学習が不安定になっていると考えられる．また，バイナリマスクによって切り出した摂動領域のみでモデルが誤分類するようになっていないことも原因の 1 つとして考えられる．そのため，今後の課題として，矩形内外の摂動領域のみでモデルを誤分類させられるような摂動導出方法の考案や，ラベルスムージング方法の再検討などが挙げられる．

第6章

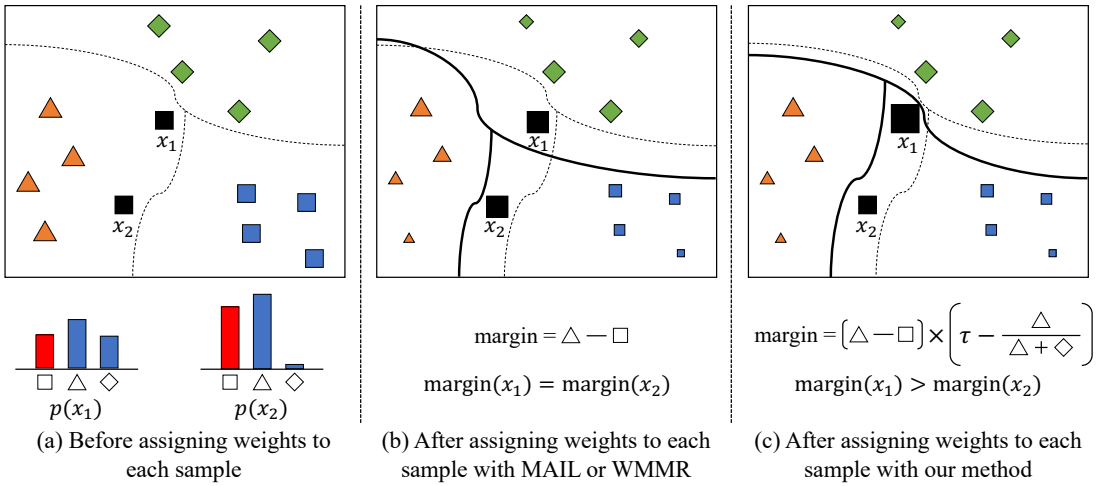
多クラス間のマージンを考慮した敵対的学習

敵対的防御手法の 1 つである Adversarial Training [2, 11] はシンプルな学習方法ながら優れた頑健性を獲得できる一方，ロバスト過適合や通常サンプルに対する分類精度を劣化させることが知られている [83, 149]．5 章では，多様な adversarial examples を学習中に作成することを目的としていた．本章では，ロバスト過適合を防ぐための方法の一つである Instance-Reweighted Adversarial Training (IRAT) [101, 105, 102, 103, 104] に着目した研究を行う．

IRAT は識別境界と各サンプルの間のマージンを重要度として計算し，非線形増加関数を用いて重みへ変換する．そして，IRAT は求めた重みを各サンプルの損失へ割り当て，重み付き分類誤差を最小化することで優れたモデルの頑健性を獲得する．Zhang らによって提案された Geometry-Aware Instance-Reweighted Adversarial Training (GAIRAT) [102] は，PGD の最小ステップ数 (LPS) によって，各サンプルの重要度を定義する．LPS は通常サンプルを始点として，PGD によって攻撃した時に最も初めに誤分類したステップ数を表しているため，入力空間における識別境界とのマージンを表していると捉えることができる．マージンが小さいサンプルは識別境界と近く，攻撃のリスクが高いため，大きな重みが割り当てられる．GAIRAT は従来の Adversarial Training よりも優れた頑健性を獲得できるが，LPS に基づくマージン定義が仇となり，PGD 以外の攻撃に対して脆弱である．さらに，GARIAT は著しいロバスト過適合が生じる．

Margin-Aware Instance Reweighting Learning (MAIL) [103] は入力に対する予測確率を用いて識別境界とのマージンを定義することで GAIRAT の弱点を克服した．具体的には，MAIL は正解クラスと最も迷ったクラスそれぞれの確率の差を非線形増加関数を用いて重みへ変換する．MAIL と同様の処理でマージンを求める手法として，Weighted Minimax Risk (WMMR) [101] が提案されている．これらは，非線形増加関数の設計に違いがあり，MAIL の方が優れた性能を獲得できる．離散的な重み定義をする GAIRAT とは異なり，MAIL は連続値の重み定義ができるためロバスト過適合を緩和することができる．しかし，MAIL や WMMR は常に正解クラスと最も迷ったクラスのみに着目してマージンを求めるため，多クラス分類おいて適切な表現ができない．この致命的な問題は，図 6.1(b) に示すように，同じ大きさ，または限りなく等しいマージンが求められるサンプルに対して生じる．直感的には，このようなサンプルに関して，マージンが求められたとしても，複数クラスの識別境界の交点付近に分布するサンプルほど大きな重み付けがされるべきであるが，従来法においてこのような表現が無視されている．

従来法の弱点を解消するために，本研究では，具体的な例を用いて前述したような問題が生じる



(a) Before assigning weights to each sample　(b) After assigning weights to each sample with MAIL or WMMR　(c) After assigning weights to each sample with our method

図 6.1: 従来法と提案手法を組み込んだ表現の違いを表したコンセプト図．図形の大きさは重みの大きさ，実線と破線はそれぞれパラメータ更新前後の識別境界を表している．また，図形の種類は異なるクラスを表している．

ことを明らかにする．そして，図 6.1(c) に示すように，従来の算出方法で求めたマージンを多クラス分類に適したマージンに変換するための *Margin Reweighting* を提案する．多クラス分類に適した重みを求める直感的なアプローチとして，正解クラスとその他の全クラスとのマージンを計算することが挙げられるが，マルチクラスのマージンを 1 つの重みに集約するための非線形増加関数を慎重に設計する必要があるため実現が困難である．そのため，本研究では「不正解率に含まれる top2 の確率の割合」という新たな指標を提案する．クラス確率がクラス中心との関係を表しているという先行研究の考えに従えば，この指標は任意のサンプルが複数クラスの識別協会の交点付近のサンプルかどうかを特定することができる．したがって，特別な非線形増加関数を設計することなく，この指標を従来のマージンに乗算するだけで適切な表現に変換することができる．提案手法は従来法に組み込むことで，いくつかの攻撃に対する頑健性を底上げする．まとめると，本研究は以下に示すような貢献をしている．

- 本研究では従来法の確率に基づくマージンが多クラス分類を想定した場合，不十分な表現であることを証明する．具体的には，異なるクラス確率をもつサンプルから限りなく等しいマージンが求められることを示す．
- 本研究では最も迷ったクラスと不正解クラス確率の関係を活用して従来のマージンを適切な表現に変換が可能な *Margin Reweighting* を提案する．
- 実験によって，提案手法は従来手法の頑健性を向上させるために有効であることを示す．

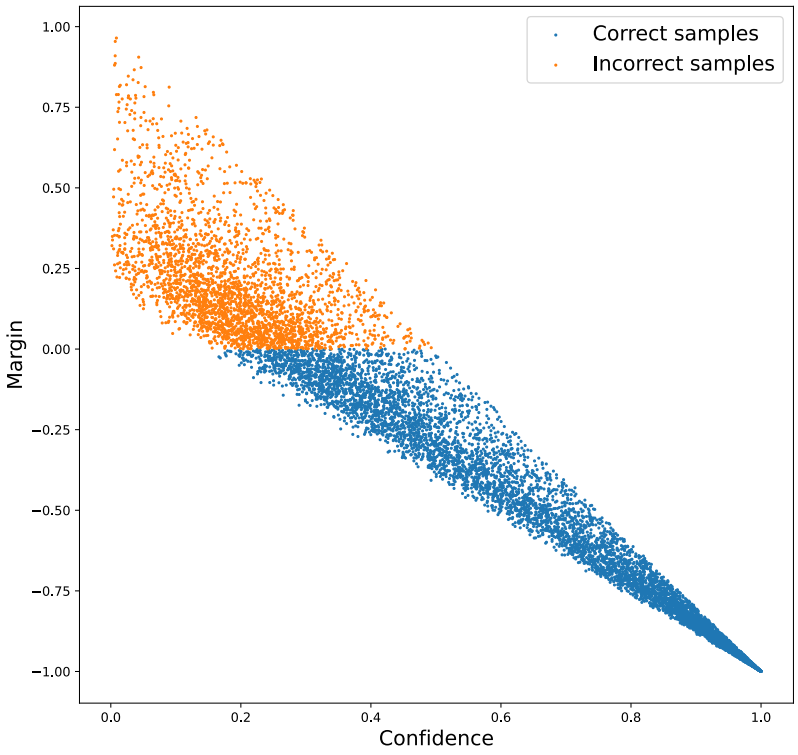


図 6.2: CIFAR-10 を用いて通常の Adversarial Training した ResNet-18 おける，正解クラス確率 $p_{y_i}(\boldsymbol{x}_i)$ と式 (2.37) によって計算されたマージンの相関．オレンジとブルーの点は，それぞれ正解サンプルと不正解サンプルを表している．

6.1 従来の IRAT における弱点

Definition 1 (top2 の確率)**.** $(\boldsymbol{x}_i, y_i) \sim \mathcal{D}$ を入力データ，$p(\boldsymbol{x}_i) \in [0,1]^K$ を $\boldsymbol{x}_i$ に対する予測分布とする．そして，top2 の確率は $p_2(\boldsymbol{x}_i) = \arg\max_{k \neq y_i} p_k(\boldsymbol{x}_i)$ として定義される．

厳密に言えば，Definition 1 の top2 の確率は $p_{y_i}(\boldsymbol{x}_i) < p_2(\boldsymbol{x}_i)$ の時，全てのクラスの中で最も高いクラス確率となる．すなわち，top1 の確率となる．しかしながら，言葉の混同を避けるために，本研究ではこのような状況下においても top2 の確率として表記することに注意されたい．

クラス確率に基づく従来の IRAT は優れた頑健性を得られる反面，最も迷ったクラス以外のクラスとの関係は考慮されていない．よほど簡単な問題設定でない限り，多クラスを扱う場合，複数クラスの識別境界の交点付近に分布するサンプルは必ず存在するはずである．分類クラス数の増加や複雑な画像分類のように問題設定が難しくなるにつれて，このようなサンプルは増加するはずである．そのため，WMMR や MAIL は多クラス分類を扱う場合において，しばしば重み表現が不十分であることが容易に想定できる．

Lemma 1. 式 (2.37) によって算出されたマージンと学習用データセット $\mathcal{D} := \{(\boldsymbol{x}_i, y_i)\}_{i=1}^n$ に関して，

以下の式を満足するサンプルペアが少なくとも 1 つは存在する．

$$m(\boldsymbol{x}_i, y_i) = m(\boldsymbol{x}_j, y_j) \ \text{ or } \ m(\boldsymbol{x}_i, y_i) \approx m(\boldsymbol{x}_j, y_j) \tag{6.1}$$

Proof: 学習用データセット $\mathcal{D} := \{\boldsymbol{x}_i, y_i\}_{i=1}^n$ を，上手く分類できたサンプル集合 $\mathcal{S}^+ := \{(\boldsymbol{x}_i, y_i) \in \mathcal{D} \mid \arg\max p(\boldsymbol{x}_i) = y_i\}$ と，誤分類が生じているサンプル集合 $\mathcal{S}^- := \{(\boldsymbol{x}_i, y_i) \in \mathcal{D} \mid \arg\max p(\boldsymbol{x}_i) \neq y_i\}$ の 2 つのサンプル集合に分割する．まず，$\mathcal{S}^+$ および $\mathcal{S}^-$ 共に，$p_{y_i}(\boldsymbol{x}_i) = p_{y_j}(\boldsymbol{x}_j)$ の時，Lemma 1 が成立することは明らかである．

次に，異なる確率を持つサンプルペア，すなわち $p_{y_i}(\boldsymbol{x}_i) \neq p_{y_j}(\boldsymbol{x}_j)$ において Lemma 1 が成立することを示す．異なる確率を持つサンプルに関して，$\mathcal{S}^+$ および $\mathcal{S}^+$ に含まれる全てのサンプルは，$p_{y_i}(\boldsymbol{x}_i) > p_2(\boldsymbol{x_i})$ または $p_{y_i}(\boldsymbol{x}_i) < p_2(\boldsymbol{x_i})$ を必ず満たすため，以下の等式を満足するならば Lemma 1 は必ず成立する．

$$|p_{y_i}(\boldsymbol{x}_i) - p_{y_j}(\boldsymbol{x}_j)| = |p_2(\boldsymbol{x}_i) - p_2(\boldsymbol{x}_j)|. \tag{6.2}$$

したがって，異なる正解クラス確率と top2 の確率をもつにもかかわらず，等しいまたは限りなく等しいマージンが算出されるサンプルペアが存在する． □

Lemma 1 の条件下の元，式 (2.37) によって算出されたマージンを調査する．まず，正解クラスまたは top2 の確率どちらか一方に高い信頼度が割り当てられるサンプルは，他のクラスから離れていることが表される．つまり，このようなサンプルは正解クラスまたは top2 のクラスどちらか一方のクラス中心付近に分布しているサンプルである．この場合，高い信頼度が割り当てられたサンプルと低い信頼度が割り当てられたサンプル間で式 (6.1) が成立しないことは，図 6.2 より明らかである．そのため，限りなく 2 クラス分類に等しいと考えて，式 (2.37) を用いてマージンを算出しても問題ない．

一方，正解クラスおよび top2 のクラス共に低い信頼度が割り当てられた場合は，複数クラスの識別境界の交点付近に存在すると捉えられる．そのため，このようなサンプルは 2 つのクラスに割り当てられる確率が限りなく等しい場合だけでなく，明らかに異なる確率 ($p_{y_i}(\boldsymbol{x}_i) \ll p_{y_j}(\boldsymbol{x}_j)$) が割り当てられた場合においても，式 (6.1) が成立する．図 6.2 に示すように，例えば，信頼度が $[0.2, 0.6]$ の範囲に着目すると，限りなく等しいマージンが算出されていることが観測できる．これは，$\boldsymbol{x}_i$ が複数クラスの識別境界の交点に近く，$\boldsymbol{x}_j$ よりも攻撃リスクが高いことを表している．故に，このようなサンプルに関して，式 (2.37) によって算出されたマージンのまま学習に使用することは，多クラス分類を扱う場合において，不適切であると言える．このようなサンプルに対して慎重にマージンを定義することで，頑健性の更なる改善が期待できるが，WMMR や MAIL ではこれらが考慮されていない．

6.2 提案手法：Margin Reweighting

本章では，提案手法の詳細を述べる．まず，最も迷ったクラスと不正解率の関係を表現できる新たな指標を提案する．次に，提案した指標に所望の動作をさせるための更なる重み付けについて述べる．最後に，提案手法を従来法に組み込んだ場合の非線形増加関数について述べる．

本研究の目標 我々の最終目標は，正解クラスと隣接クラス確率から求めたマージンの影響を保ちつつ，更なる重み付けによって同じマージンを適切な表現に変換することである．

6.2.1 マージンに対する重要度の定量化

異なる確率を持ついくつかのサンプルに関して Lemma 1 が成立することを思い起こすと，従来のマージン定義が不十分であることは明らかである．この問題を解決するための直感的なアプローチは，式 (2.37) を用いて正解クラスを除く全てのクラスとのマージンを計算することである．しかし，マルチクラスマージンを 1 つの適切な重みに集約する非線形増加関数を慎重に設計する必要があるため，このアプローチを実現することが困難である．Holtz ら [106] はメタ学習を用いてマルチクラスマージンを重みへ変換しているが，複雑な学習かつ学習時間が増加する．

Definition 2 (不正解率)**.** $(\boldsymbol{x}_i, y_i) \sim \mathcal{D}$ を入力データ，$p_{y_i}(\boldsymbol{x}_i) \in [0,1]$ を正解クラス確率とする．そして，不正解率は以下のように定義される．

$$\begin{aligned} \bar{p}_{y_i}(\boldsymbol{x}_i) &= 1 - p_{y_i}(\boldsymbol{x}_i) \\ &= \sum\nolimits_{k \neq y_i} p_k(\boldsymbol{x}_i) \end{aligned} \tag{6.3}$$

そのため，本研究では top2 の確率と不正解率を利用して，最も迷ったクラスとその他のクラスの関係を定量化するための指標を提案する．Lemma 1 が異なる信頼度をもつサンプルペアに対する成立を想定すると，不正解率に含まれる top2 の確率の割合に関して以下続く不等式が成立する．

Proposition 1. 入力データ $(\boldsymbol{x}_i, y_i)$ と $(\boldsymbol{x}_j, y_j)$ に関して，$m(\boldsymbol{x}_i, y_i) = m(\boldsymbol{x}_j, y_j)$，$p_{y_i}(\boldsymbol{x}_i) > p_{y_j}(\boldsymbol{x}_j)$，$p_{y_j}(\boldsymbol{x}_j) > \frac{1}{K}$ であるとする．そして，各サンプルにおける不正解率に含まれる top2 の割合について以下の不等式が成立する．

$$o(\boldsymbol{x}_i, y_i) > o(\boldsymbol{x}_j, y_j), \text{ where } o(\boldsymbol{x}, y) = \frac{p_2(\boldsymbol{x})}{\bar{p}_y(\boldsymbol{x})}. \tag{6.4}$$

Proof: 入力データ $(\boldsymbol{x}_i, y_i)$ と $(\boldsymbol{x}_j, y_j)$ におけるそれぞれの正解クラス確率が $p_{y_i}(\boldsymbol{x}_i) > p_{y_j}(\boldsymbol{x}_j)$ の関係のもとで，Lemma 1 が成立することを想定すると，以下に続く top2 の確率に関する不等式を満たすことは明らかである．

$$p_2(\boldsymbol{x}_i) > p_2(\boldsymbol{x}_j) \tag{6.5}$$

故に，全てのデータに関する不正解率は

$$\bar{p}_{y_i}(\boldsymbol{x}_i) < \bar{p}_{y_j}(\boldsymbol{x}_j) \tag{6.6}$$

を満足する．top2 の確率は如何なる場合においても不正解率を超えないことが保証できるため，以下に続く不等式が成立する．

$$\frac{p_2(\boldsymbol{x}_i)}{\bar{p}_{y_i}(\boldsymbol{x}_i)} > \frac{p_2(\boldsymbol{x}_j)}{\bar{p}_{y_j}(\boldsymbol{x}_j)} \tag{6.7}$$

したがって，$o(\boldsymbol{x}, y) = \frac{p_2(\boldsymbol{x})}{\bar{p}_y(\boldsymbol{x})}$ を用いることで，$o(\boldsymbol{x}_i, y_i) > o(\boldsymbol{x}_j, y_j)$ を獲得することができる． □

式 (2.37) で求めたマージンは，識別境界との関係や任意のサンプルが誤分類しているかどうかのみ表現している．一方，同じマージンが算出されたサンプルに関して，Proposition 1 は不正解率中に含まれる top2 の確率の割合を用いることで，更なる区別ができることを表している．MAIL や WMMR の考え方，すなわち式 (2.37) によって識別クラスと任意のサンプル間のマージンを暗に示していることに従えば，top2 の割合が低いサンプルは複数のクラスの交点付近に分布していると判断できる．したがって，不正解率に含まれる top2 の確率を式 (2.37) によって計算されたマージンに上手く適用することで問題を解決することができる．

さて，次に top2 の割合 $o(\boldsymbol{x}_i, y_i)$ の取り得る値の範囲について議論する．top2 の確率 $p_2(\boldsymbol{x}_i)$ の取り得る値の範囲を $P_{\text{top2}} := \left(\frac{\bar{p}_{y_i}(\boldsymbol{x}_i)}{K-1}, \bar{p}_{y_i}(\boldsymbol{x}_i)\right)$ とすると，$o(\boldsymbol{x}_i, y_i)$ を直接用いた時の上限値および下限値は以下のように定義される．

$$\begin{cases} \sup\limits_{p_2(\boldsymbol{x}_i) \in P_{\text{top2}}} o(\boldsymbol{x}_i, y_i) = 1 & p_2(\boldsymbol{x}_i) = \bar{p}_{y_i}(\boldsymbol{x}_i) \\ \inf\limits_{p_2(\boldsymbol{x}_i) \in P_{\text{top2}}} o(\boldsymbol{x}_i, y_i) = \frac{1}{K-1} & p_2(\boldsymbol{x}_i) = \frac{\bar{p}_{y_i}(\boldsymbol{x}_i)}{K-1} \end{cases} \tag{6.8}$$

式 (6.8) によれば，$o(\boldsymbol{x}_i, y_i)$ は $p_2(\boldsymbol{x}_i)$ が限りなく大きいとき，または小さいときに上限値と下限値に近づくため，top2 の割合を直接使用することは本研究の目的と反した振る舞いとなる．故に，任意の係数 $\tau \in \mathbb{N}$ によって top2 の割合を反転させることで適切な範囲となる．しかし，$\tau = 1$ を考えた場合の上限値と下限値は次式のように求まるため，まだ適切な表現ができていないことになる．

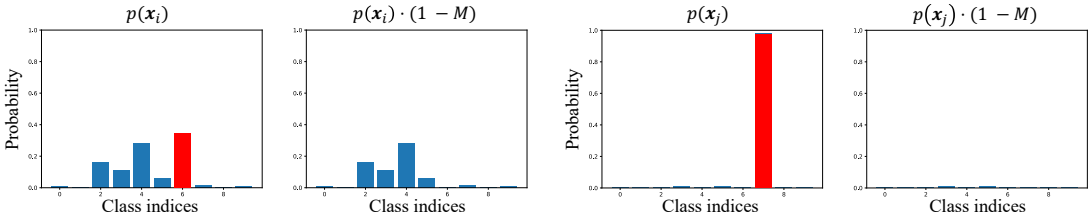
$$\begin{cases} \sup\limits_{p_2(\boldsymbol{x}_i) \in P_{\text{top2}}} (\tau - o(\boldsymbol{x}_i, y_i)) = 1 - \frac{1}{K-1} & p_2(\boldsymbol{x}_i) = \frac{\bar{p}_{y_i}(\boldsymbol{x}_i)}{K-1} \\ \inf\limits_{p_2(\boldsymbol{x}_i) \in P_{\text{top2}}} (\tau - o(\boldsymbol{x}_i, y_i)) = 0 & p_2(\boldsymbol{x}_i) = \bar{p}_{y_i}(\boldsymbol{x}_i) \end{cases} \tag{6.9}$$

式 (6.9) は，$p_2(\boldsymbol{x}) \approx \bar{p}_y(\boldsymbol{x})$ の top2 の確率を持つサンプル，つまり著しく誤分類が生じている adversarial examples が，もはや重みパラメータ更新に含まれないことを暗に示している．

従って，本研究では以下の式で式 (2.37) によって求めたマージンに関する重みを定義する．

$$s(\boldsymbol{x}_i, y_i) = \tau - o(\boldsymbol{x}_i, y_i), \ \text{s.t.}\ \tau \in \mathbb{N}_{\geq 2} \tag{6.10}$$

ここで，$\mathbb{N}_{\geq 2}$ は 2 以上の自然数集合を表している．



(a) Low-confidence example, $c(\boldsymbol{x}_i, y_i) = 0.7083$　(b) High-confidence example, $c(\boldsymbol{x}_j, y_j) = 0.0143$

図 6.3: (a) 低い信頼度および (b) 高い信頼度のサンプル，それぞれの予測分布に関して式 (6.11) で計算したコサイン類似度計算の関係．赤色は各サンプルにおける正解クラスを表している．

6.2.2　不正解率に対する重要度表現

6.2.1 において，複数のクラスの交点に近いサンプルかどうかの表現が可能な尺度を提案したが，式 (6.10) はまだ表現が不十分である．具体的には，高い信頼度で分類に成功したサンプルに関して，不正解クラスに対して一様に等しく確率が割り当てられたならば，top2 の割合は境界付近のサンプルと限りなく等しくなる．つまり，境界付近のサンプルは top2 の割合が高くなり，上手く分類できたサンプルは top2 の割合が低くなる保証がされていない．これらのサンプルは異なる重み定義されるべきであることから，不正解率に対しても重要度が異なることを意味している．そこで，次式で表される予測確率分布に関するコサイン類似度によって不正解率の重要度を表現する．

$$c(\boldsymbol{x}_i, y_i) = \frac{< p(\boldsymbol{x}_i), p'(\boldsymbol{x}_i) >}{\|p(\boldsymbol{x}_i)\|_2 \cdot \|p'(\boldsymbol{x}_i)\|_2} \tag{6.11}$$

ここで，$p(\boldsymbol{x}_i) \in [0, 1]^K$ は $\boldsymbol{x}_i$ に対する予測確率分布を表している．$p'(\boldsymbol{x}_i)$ は正解クラスを排除して，正解クラス以外が 1 であるバイナリマスクを用いて分布を表しており，以下の式によって求められる．

$$p'(\boldsymbol{x}_i) := p(\boldsymbol{x}_i) \cdot (1 - M), \ \text{s.t.} \ \sum_{k=1}^{K} p'_k(\boldsymbol{x}_i) < 1 \tag{6.12}$$

図 6.3(a) に示すように，式 (6.11) によって算出される類似度は，正解クラス確率が低いサンプルに対して高くなる傾向にある．一方，図 6.3(b) に示すように，正解クラス確率が高いサンプルに対しては類似度が低くなる傾向がある．このような傾向は一部のサンプルに対してだけでなく，図 6.4 に示すように，多くのサンプルにおいて同様の傾向が確認できた．図 6.4 に示した，不正解率と式 (6.11) によって算出した類似度の相関を次式を用いて計算したところ $r = 0.98$ と非常に強い正の相関があることを確認した．

$$r = \frac{\frac{1}{n}\sum_{i=1}^{n}(\bar{p}_{y_i}(\boldsymbol{x}_i) - \bar{p}')(c(\boldsymbol{x}_i, y_i) - c')}{\sqrt{\frac{1}{n}\sum_{i=1}^{n}(\bar{p}_{y_i}(\boldsymbol{x}_i) - \bar{p}')^2}\sqrt{\frac{1}{n}\sum_{i=1}^{n}(c(\boldsymbol{x}_i, y_i) - c')^2}}, \tag{6.13}$$

ここで，n はサンプル数，$\bar{p}' = \frac{1}{n}\sum_{i=1}^{n}\bar{p}_{y_i}(\boldsymbol{x}_i)$，$c' = \frac{1}{n}\sum_{i=1}^{n}c(\boldsymbol{x}_i, y_i)$ である．

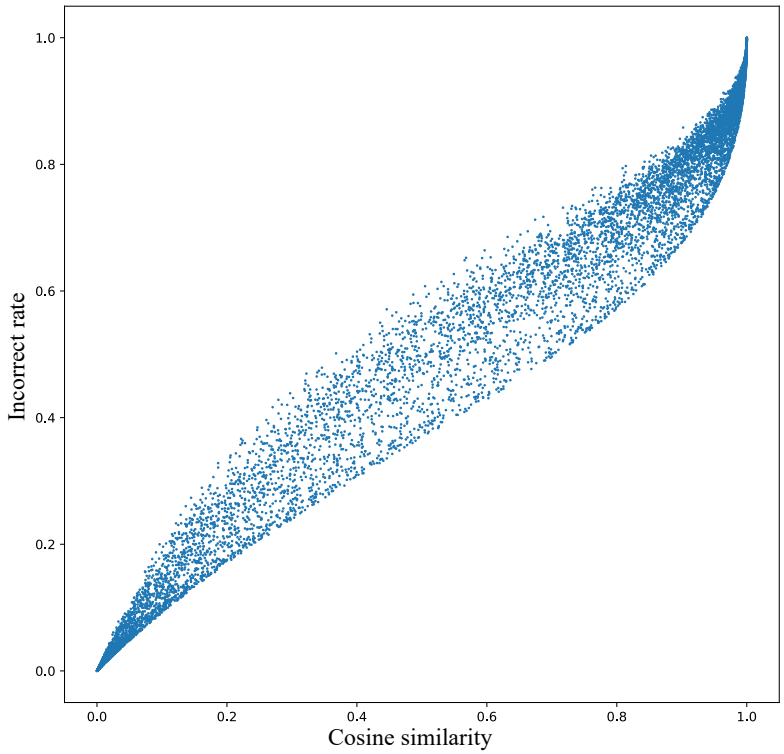


図 6.4: CIFAR-10 における，信頼度と式 (6.11) によって求めた類似度の関係.

従って，式 (6.11) を式 (6.10) に適用することで，式 (2.37) で求めたマージンに適切な重みを割り当てることができる．以上より，以下に示す式によって式 (6.10) を適切な表現へ変換する．

$$s'(\boldsymbol{x}_i, y_i) = s(\boldsymbol{x}_i, y_i) \cdot c(\boldsymbol{x}_i, y_i) \tag{6.14}$$

6.2.3 従来の IRAT への提案手法の導入

これまでで，従来法のマージン算出が不十分であることを明らかにして，式 (2.37) によって求めたマージンを top2 の割合を用いて適切な表現へ変換するための *Margin Reweighting* を提案した．ここでは，MAIL と WMMR へ提案手法を組み込む方法について述べる．

まず，次式に示すように，式 (6.14) によって計算した top2 の割合と式 (2.37) で求めたマージンに乗算する．

$$\tilde{m}(\boldsymbol{x}_i, y_i) = m(\boldsymbol{x}_i, y_i) \cdot s'(\boldsymbol{x}_i, y_i) \tag{6.15}$$

そして，複数クラスの識別境界の交点付近のサンプルに対する重要度をブースティングしたマージンを用いて，従来法の非線形増加関数を用いて重みに変換する．

$$\omega_{\text{WMMR}} = \exp(-\tilde{m}(\boldsymbol{x}_i, y_i)) \tag{6.16}$$

$$\omega_{\text{MAIL}} = \text{sigmoid}(-\gamma \cdot (\tilde{m}(\boldsymbol{x}_i, y_i) + \beta)) \tag{6.17}$$

Algorithm 2 Adversarial training incorporating our approach

Require: Training dataset $\mathcal{D}$, batch size n, training epochs T, learning rate η, model parameter $\boldsymbol{\theta}$, amount of warm-up Ω

Require: Function deriving adversarial perturbation $\mathcal{A}$

Require: Hyperparameters β, γ, τ

for $t = 1, \ldots, T$ **do**
  for $\{\boldsymbol{x}_i, y_i | i = 1, \ldots, n\} \sim \mathcal{D}$ **do**
    $\hat{\boldsymbol{x}}_i \leftarrow \mathcal{A}(\boldsymbol{x}_i, y_i; \boldsymbol{\theta})$
    $s(\hat{\boldsymbol{x}}_i, y_i) = \tau - o(\hat{\boldsymbol{x}}_i, y_i)$
    $c(\hat{\boldsymbol{x}}_i, y_i) = \frac{<p(\hat{\boldsymbol{x}}_i), p'(\hat{\boldsymbol{x}}_i)>}{\|p(\hat{\boldsymbol{x}}_i)\|_2 \cdot \|p'(\hat{\boldsymbol{x}}_i)\|_2}$
    $s'(\boldsymbol{x}_i, y_i) = s(\boldsymbol{x}_i, y_i) \cdot c(\boldsymbol{x}_i, y_i)$
    $\tilde{m}(\hat{\boldsymbol{x}}_i, y_i) = m(\hat{\boldsymbol{x}}_i, y_i) \cdot s'(\hat{\boldsymbol{x}}_i, y_i)$
    if $T > \Omega$ **then**
      if MAIL **then**
        $\omega_i = \mathrm{sigmoid}(\gamma \cdot (\tilde{m}(\hat{\boldsymbol{x}}_i, y_i) + \beta)) \times M$
      else if WMMR **then**
        $\omega_i = \exp(-\tilde{m}(\hat{\boldsymbol{x}}_i, y_i)) \times M$
      end if
    else
      $\omega_i = 1$
    end if
    model update:
    $\boldsymbol{\theta}_{t+1} \leftarrow \boldsymbol{\theta}_t - \eta \cdot \frac{1}{n} \sum_{i=1}^{n} \nabla_\theta \omega_i \cdot \ell(f_{\boldsymbol{\theta}_t}(\tilde{\boldsymbol{x}}_i), y_i)$
  end for
end for
return model parameter $\boldsymbol{\theta}$

ここで，β と γ はハイパーパラメータであり，6.3 の評価実験では従来法で使用されている定数を使用する．したがって，提案手法では τ の適切な値を決定する必要がある．

WMMR は式 (2.39) のように，式 (2.37) で求めたマージンにハイパーパラメータ α が乗算するが，α は提案手法に置き換えることで省略する．学習初期は全てのサンプルで $\omega = 1$ を用いてモデルをウォームアップする．つまり，設定した学習回数以内の学習は通常の Adversarial Training と等しくなる．このテクニックは，学習初期の不適切な重み表現を回避する目的で GAIRAT や MAIL で使われている．提案手法を組み込んだ学習プロセスの詳細は Algorithm 2 に疑似コードとして示す．

6.3 Experiments

本章では WMMR と MAIL に提案手法を組み込むことの有意性を示す．本実験では，通常サンプルに対する分類精度および，様々な攻撃に対するモデルの頑健性を評価するために CIFAR-10/100 [150] データセットを使用する．まず，学習の詳細な設定とどのように敵対的頑健性を評価するかについて 6.3.1 で説明する．そして，従来法との定量的評価を 6.3.2 で報告して，6.3.3 で MAIL や WMMR 以外の IRAT と性能比較する．最後に，ハイパーパラメータ τ や 6.2.2 で導入した類似度計算の必要性などについて 6.3.4 で議論する．

6.3.1 Experimental details

本実験では，比較手法として通常の Adversarial Training (“Standard”) [11]，WMMR，MAIL を使用する．本実験では異なる乱数シードを用いて学習したモデルの平均と標準偏差を用いて各手法を評価する．また，実験結果は学習終了時のモデル (“Last model”) と学習中の最も高性能なモデル (“Best model”) を評価する．本実験における Best model は，検証データで PGD-20 が最高性能となったチェックポイントを指している．公平な評価をするために，CIFAR-10 データセットを用いた WRN34-10 の学習時の設定は Pang ら [151] にしたがって全ての手法で統一する．

学習時の設定： 本実験ではベースネットワークとして ResNet-18 [5] と WideResNet34-10 (WRN34-10) [148] を使用する．各モデルは，128 のバッチサイズとモメンタムが 0.9 の SGD を用いて 120 エポック学習する．SGD の初期学習率は，ResNet-18 の時に 0.01，WRN34-10 の時に 0.1 に設定し，$\{75, 90, 100\}$ エポックで $1/10$ に減衰する．weight decay は ResNet-18 の時に 3.5×10^{-3}，WRN34-10 の時に 0.5×10^{-4} に設定する．最後に，全ての手法に関して，摂動許容範囲 $\epsilon = 8/255$，ステップサイズ $\alpha = \epsilon/4$，反復回数 $N = 10$ の PGD を用いて摂動を作成する．

ハイパーパラメータ： MAIL は論文値にしたがって $\beta = 0.5$ と $\gamma = 10$，WMMR は $\alpha = 2$ を全てのデータセットで使用する．これらの定数は提案手法の有無にかかわらず同じ値を使用する．ここで，提案手法を組み込んだ WMMR は α を含まないことに注意されたい．提案手法では，WRN34-10 に関して CIFAR-10 および CIFAR-100 どちらの場合も $\tau = 2$ を使用する．一方，ResNet-18 に関しては，CIFAR-10 で学習する時に $\tau = 2$，CIFAR-100 で学習する時に $\tau = 3$ を使用する．モデルのウォームアップは 75 エポックとして，それ以降，重み付き損失を最小化する．

敵対的頑健性： 本実験では，通常のサンプルに対する分類精度 (“Clean”) だけでなく，$K = 100$ の PGD-K，CW の損失を用いた PGD-100，クロスエントロピー誤差を用いた AutoPGD (APGD-CE) [152]，AutoAttack (AA) [152] に対する敵対的頑健性を評価する．Wang2021 ら [103] も述べているように，Black-box 攻撃は容易に防御できることが想定されるため，本実験では White-box 攻撃のみに着目して頑健性を評価する．

表 6.1: CIFAR-10 における各手法の Last model の性能 (%).

	Clean	PGD-100	CW-PGD	APGD-CE	AA
Standard	85.69±0.15	48.86±0.22	**49.26±0.20**	48.34±0.23	**46.38±0.26**
WMMR	**85.71±0.17**	49.60±0.41	48.31±0.36	48.99±0.38	45.23±0.37
Ours + WMMR	85.65±0.13	50.31±0.25↑	47.94±0.13	49.59±0.29↑	44.79±0.18
MAIL	82.53±0.30	56.66±0.17	47.71±0.26	55.75±0.13	45.22±0.31
Ours + MAIL	82.66±0.10↑	**57.96±0.13**↑	47.24±0.16	**56.95±0.19**↑	44.78±0.29
Standard	87.89±0.17	48.11±0.50	49.33±0.45	47.66±0.47	46.82±0.38
WMMR	**88.03±0.12**	49.33±0.41	49.95±0.33	48.88±0.43	47.37±0.38
Ours+WMMR	87.99±0.14	49.39±0.19↑	49.97±0.22↑	48.96±0.18↑	47.50±0.17↑
MAIL	86.40±0.13	59.30±0.21	**51.82±0.22**	58.57±0.26	**49.29±0.14**
Ours+MAIL	86.48±0.24↑	**60.56±0.42**↑	51.55±0.30	**59.64±0.45**↑	48.92±0.30

6.3.2 従来法との比較

表 6.1 に CIFAR-10 と表 6.2 に CIFAR-100 の Last model の結果，表 6.3 に CIFAR-10 と表 6.4 に CIFAR-100 の Best model の結果をそれぞれ示す．各表に関して，上段が ResNet-18，下段が WRN34-10 の結果である．また，最高性能の結果は太字で強調しており，↑は提案手法を組み込むことで分類精度が改善されたことを表している．

■ Last model に対する評価結果

ここでは，CIFAR-10 と CIFAR-100 を分割して，Last model における評価結果に対して議論する．

CIFAR-10: 表 6.1 の上段の ResNet-18 の結果に関して，提案手法を組み込むこと (すなわち “Ours+MAIL”) によって PGD-100 と APGD-CE に対する敵対的頑健性能が向上した．さらには，Ours+MAIL は “Clean”の分類精度が MAIL よりも僅かに改善した．Ours+WMMR の結果においても同様の傾向が観測できた．

次に，表 6.1 の下段の WRN34-10 の結果に関して，Ours+WMMR は全ての敵対的攻撃に対して従来の WMMR よりも優れた頑健性を示した．特に，AA に対する頑健性の平均値が 0.13 ポイント改善しており，5 つの結果の標準偏差が著しく低くなる．一方，Ours+MAIL は通常サンプルに対する分類精度が従来と同程度であり，PGD-100 や APGD-CE のような PGD 系統の攻撃に対する頑健性が著しく改善された．

CIFAR-100: 表 6.2 の上段に示す ResNet-18 の結果に関して，Ours+MAIL は通常サンプルに対する分類精度が MAIL と比較して 0.73 ポイント優れた性能であり，PGD-100 や APGD-CE に対する頑

表 6.2: CIFAR-100 における各手法の Last model の性能 (%).

	Clean	PGD-100	CW-PGD	APGD-CE	AA
Standard	60.10±0.22	28.65±0.21	**27.89±0.17**	28.12±0.18	**25.01±0.13**
WMMR	60.42±0.28	28.14±0.16	26.66±0.18	27.47±0.15	23.59±0.14
Ours + WMMR	**60.50±0.18**↑	28.24±0.12↑	26.96±0.20↑	27.58±0.14↑	23.86±0.15↑
MAIL	56.77±0.19	**31.23±0.08**	25.68±0.26	30.45±0.09	22.99±0.09
Ours + MAIL	57.50±0.23↑	**31.23±0.16**	25.30±0.09	**30.46±0.17**↑	22.61±0.10
Standard	62.63±0.33	24.82±0.12	25.80±0.17	24.53±0.13	23.70±0.13
WMMR	63.32±0.12	25.87±0.22	26.29±0.20	25.45±0.19	23.81±0.19
Ours+WMMR	**63.76±0.09**↑	25.45±0.24	26.51±0.15↑	25.14±0.21	23.94±0.18↑
MAIL	62.56±0.21	34.29±0.08	**29.31±0.19**	**33.58±0.09**	**26.50±0.16**
Ours+MAIL	63.19±0.21↑	**34.32±0.13**↑	28.94±0.18	33.47±0.18	26.18±0.14

健性が MAIL と同程度であった．一方，Ours+WMMR は Clean の結果だけでなく，全ての攻撃に対して 0.10 ポイント以上，優れた性能を獲得できた．

表 6.2 の下段に示す WRN34-10 に関して，Ours+WMMR は通常のサンプルに対する分類精度が 0.44 ポイント向上した．さらには，Ours+WMMR は CW-PGD や AA に対する頑健性も改善されたが，PGD 系統の攻撃に対する頑健性は僅かに劣化することを確認した．Ours+MAIL は通常サンプルに対する分類精度が 0.63 改善されており，PGD-100 に対する頑健性が MAIL と同程度であった．これらの結果は，提案手法が通常のサンプルに対する分類精度を改善するだけでなく，従来法と同程度以上の敵対的頑健性を獲得するために有効であることを示している．

■ **Best model に対する評価結果**

ここでは，CIFAR-10 と CIFAR-100 を分割して，Bast model における評価結果に対して議論する．

CIFAR-10: 表 6.3 に示す ResNet-18 および WRN34-10 どちらの結果も，提案手法を組み込むことによって通常サンプルに対する分類精度が改善された．特に，Ours+WMMR は通常サンプルを最も高精度に分類できる．ResNet-18 に関して，Ours+WMMR および Ours+MAIL どちらにおいても PGD-100 と APGD-CE に対する頑健性が 0.5 ポイント以上向上した．WRN34-10 に関して，Ours+WMMR は CW-PGD および AA に対する頑健性が 0.5 ポイント程度改善され，Ours+MAIL は PGD-100 および APGD-CE に対する頑健性が 1 ポイント以上改善された．Ours+MAIL では CW-PGD や AA に対する頑健性の改善が観測できなかったが，MAIL と同程度であった．

CIFAR-100: 表 6.4 に示した ResNet-18 と WRM34-10 の結果は，CIFAR-10 の結果と同様，提案手法によって通常サンプルに対する分類精度を改善することができる．Ours+WMMR は CW-PGD

表 6.3: CIFAR-10 における各手法の Best model の性能 (%).

	Clean	PGD-100	CW-PGD	APGD-CE	AA
Standard	83.84±0.171	50.98±0.21	**50.50±0.31**	50.55±0.22	**47.86±0.13**
WMMR	83.74±0.23	52.33±0.21	49.12±0.12	51.70±0.25	46.57±0.12
Ours + WMMR	**83.87±0.27**↑	52.91±0.16↑	48.52±0.24	52.17±0.16↑	46.04±0.24
MAIL	82.37±0.23	56.99±0.20	47.90±0.19	56.07±0.23	45.44±0.18
Ours + MAIL	82.66±0.09↑	**58.36±0.30**↑	47.43±0.23	**57.24±0.30**↑	44.82±0.36
Standard	86.67±0.21	54.17±0.23	**54.05±0.15**	53.79±0.22	**51.73±0.32**
WMMR	86.57±0.18	55.16±0.37	52.76±0.39	54.64±0.35	50.36±0.33
Ours + WMMR	**86.91±0.31**↑	55.00±0.25	53.27±0.08↑	54.49±0.30	50.73±0.12↑
MAIL	86.28±0.12	59.77±0.21	51.91±0.29	58.93±0.26	49.47±0.33
Ours + MAIL	86.45±0.19↑	**61.05±0.25**↑	51.87±0.17	**60.08±0.21**↑	49.26±0.22

や AA に対する頑健性が WMMR よりも僅かに改善された．一方，Ours+MAIL はどの攻撃に対しても頑健性の向上が確認できなかった．これは，WRN34-10 の結果でも同様である．WRN34-10 を Ours+WMMR で学習すると全ての攻撃に対する頑健性が 0.16 から 0.82 ポイント改善された．

6.3.3 その他の IRAT との比較

本章では MAIL と WMMR を除く IRAT と性能比較する．本実験では，Last model の結果のみに着目して議論する．CIFAR-10 と CIFAR-100 の結果を表 6.5 と表 6.6 それぞれに示す．各表において，上段が ResNet-18，下段が WRN34-10 の結果であり，最高性能の結果を太字で強調している．

まず，表 6.5 に示す CIFAR-10 の結果では，PGD-100 および APGD-CE 対する頑健性は Ours+MAIL が最も優れている．APGD-CE の頑健性は Ours+MAIL に次いで Ours+WMMR が優れている．さらに，Ours+WMMR は通常サンプルに対する分類精度が最も優れている．CW-PGD や AA に対する頑健性に関しては EWAT に劣る結果となった．この傾向は WRN34-10 および ResNet-18 どちらも同様である．

次に，表 6.6 に示す CIFAR-100 の結果では，ResNet-18 と WRN34-10 どちらにおいても Ours+WMMR を用いることにより，通常サンプルに対する精度が最も優れている．Ours+MAIL は ResNet-18 と WRN34-10 どちらにおいても，PGD-100 と APGD-CE の頑健性が最も優れている．特に，WRN34-10 に関しては CW-PGD や AA に対する頑健性も優れている．

以上より，提案手法を組み込むことによって，一部の敵対的攻撃を除けば，MAIL や WMMR 以外の IRAT よりも非常に優れた頑健性が獲得できると結論づけることができる．

表 6.4: CIFAR-100 における各手法の Best model の性能 (%).

	Clean	PGD-100	CW-PGD	APGD-CE	AA
Standard	57.48±0.50	29.51±0.06	**27.76±0.13**	28.71±0.55	**25.16±0.14**
WMMR	57.82±0.45	29.03±0.16	26.84±0.20	28.34±0.21	24.02±0.27
Ours + WMMR	**58.07±1.35**↑	28.94±0.09	27.03±0.28↑	28.30±0.13	24.19±0.04↑
MAIL	56.65±0.18	**31.36±0.07**	25.72±0.18	**30.62±0.09**	23.02±0.17
Ours + MAIL	57.59±0.12↑	31.25±0.16	24.80±0.24	30.44±0.17	22.06±0.24
Standard	62.21±0.19	31.43±0.31	**30.51±0.27**	31.00±0.27	**27.96±0.10**
WMMR	62.08±0.16	30.99±0.12	29.09±0.17	30.50±0.17	26.61±0.24
Ours + WMMR	62.51±0.64↑	31.15±0.29↑	29.91±0.22↑	30.73±0.31↑	27.22±0.35↑
MAIL	62.36±0.21	**34.64±0.12**	29.42±0.13	**33.94±0.14**	26.67±0.08
Ours + MAIL	**63.00±0.11**↑	34.56±0.18	29.07±0.16	33.78±0.19	26.38±0.21

表 6.5: CIFAR-10 における，GAIRAT と EWAT との性能比較 (%).

	Clean	PGD-100	CW-PGD	APGD-CE	AA
GAIRAT	83.56±0.32	52.67±0.35	35.00±0.52	48.85±0.43	31.88±0.45
EWAT	84.77±0.14	49.88±0.10	**50.31±0.12**	49.47±0.11	**47.70±0.13**
Ours + WMMR	**85.65±0.13**	50.31±0.25	47.94±0.13	49.59±0.29	44.79±0.18
Ours + MAIL	82.66±0.10	**57.96±0.13**	47.24±0.16	**56.95±0.19**	44.78±0.29
GAIRAT	85.89±0.13	56.42±0.27	40.86±0.45	50.55±0.47	38.33±0.42
EWAT	86.44±0.21	52.35±0.77	**52.55±1.16**	51.88±0.89	**49.71±1.21**
Ours+WMMR	**87.99±0.14**	49.39±0.19	49.97±0.22	48.96±0.18	47.50±0.17
Ours+MAIL	86.48±0.24	**60.56±0.42**	51.55±0.30	**59.64±0.45**	48.92±0.30

6.3.4 Ablation study

ここでは，式 (6.11) の類似度計算を排除したときや，異なるハイパーパラメータ τ を使用したときの性能に与える影響について議論する．本実験では，これまでの実験同様，5 つの異なる乱数シードを用いて学習した WRN34-10 の平均精度と標準偏差の結果を示す．

表 6.7 に $\tau = \{2, 3, 4\}$ を用いた時の Clean，CW-PGD，APGD-CE の結果を示す．ここで，表 6.7 の上段と下段はそれぞれ，WRN34-10 と ResNet-18 である．CIFAR-10 の結果は，τ が大きくなるにつれて，WRN34-10 では CW-PGD の性能が劣化する反面，APGD-CE の性能は劇的に改善される．ResNet-18 は τ が増加に伴って，CW-PGD が劣化する一方，APGD-CE に対する頑健性は WRN34-10 と同様で著しく増加する．Clean な性能はどちらの場合でも僅かに劣化するだけであり，τ の大きさに依存しない精度である．

表 6.7 の右に示す CIFAR-100 の結果は，τ の値が増加すると通常サンプルの分類精度が改善され，

表 6.6: CIFAR-100 における，GAIRAT と EWAT との性能比較 (%).

	Clean	PGD-100	CW-PGD	APGD-CE	AA
GAIRAT	58.32±0.14	23.10±0.26	18.77±0.12	22.08±0.32	15.66±0.20
EWAT	58.22±0.62	28.69±0.10	26.69±0.22	28.25±0.12	**24.54±0.34**
Ours + WMMR	**60.50±0.18**	28.24±0.12	**26.96±0.20**	27.58±0.14	23.86±0.15
Ours + MAIL	57.50±0.23	**31.23±0.16**	25.30±0.09	**30.46±0.17**	22.61±0.10
GAIRAT	61.33±0.22	24.40±0.38	23.64±0.32	23.88±0.37	21.18±0.26
EWAT	62.59±0.31	25.71±0.16	26.11±0.17	25.36±0.11	23.89±0.16
Ours+WMMR	**63.76±0.09**	25.45±0.24	26.51±0.15	25.14±0.21	23.94±0.18
Ours+MAIL	63.19±0.21	**34.32±0.13**	**28.94±0.18**	**33.47±0.18**	**26.18±0.14**

表 6.7: ハイパーパラメータ τ に関する Ablation study.

	CIFAR-10			CIFAR-100		
	Clean	CW-PGD	APGD-CE	Clean	CW-PGD	APGD-CE
$\tau = 2$	**86.48±0.24**	**51.55±0.30**	59.64±0.45	63.19±0.21	**28.94±0.18**	**33.47±0.18**
$\tau = 3$	86.39±0.14	50.65±0.35	60.77±0.48	63.44±0.22	27.79±0.25	32.65±0.17
$\tau = 4$	86.37±0.25	50.49±0.50	**62.09±0.47**	**63.62±0.29**	26.87±0.23	31.38±0.19
$\tau = 2$	82.66±0.10	**47.24±0.16**	56.95±0.19	56.80±0.26	**25.81±0.13**	30.36±0.14
$\tau = 3$	**82.78±0.16**	47.07±0.18	56.71±0.48	57.50±0.23	25.30±0.09	**30.46±0.17**
$\tau = 4$	82.35±0.32	45.54±0.39	**59.30±0.20**	**57.78±0.16**	24.68±0.11	30.15±0.06

CW-PGD と APGD-CE が劣化する傾向がある．これは ResNet-18 と WRN34-10 どちらも同じような傾向が観測できる．

CIFAR-10 および，CIFAR-10 ともに敵対的頑健性が 1.0 ポイント以上劣化する．したがって，WRN34-10 は CIFAR-10 および CIFAR-100 ともに $\tau = 2$ が適切である．ResNet-18 に関しては，CIFAR-10 において $\tau = 2$，CIFAR-100 において $\tau = 3$ が適切である．

次に，表 6.8 に提案手法から式 (6.11) の類似度計算を排除して学習したモデルに対する Clean，CW-PGD，APGD-CE の結果を示す．ここで，表 6.8 において，sim. は similarity の省略であり，上段と下段はそれぞれ Last model と Best model の結果である．Clean や CW-PGD，AA の結果は類似度計算を排除することによって劣化する一方，PGD-100 および APGD-CE は僅かに頑健性が改善された．この傾向は Last model と Best model ともに同様である．古典的な PGD は非常に強い攻撃ができるが，PGD のみで優劣を判断するとモデルの性能を過大評価する可能性が高い．また，PGD や APGD-CE における性能向上は非常に限定的であることからも，提案手法は類似度を使用して学習することが適切であると考えられる．

表 6.8: 式 (6.11) の類似度計算に関する Ablation study.

	Clean	PGD-100	CW-PGD	APGD-CE	AA
w/ sim.	**86.48±0.24**	60.56±0.42	**51.55±0.30**	59.64±0.45	**48.92±0.30**
w/o sim.	86.40±0.09	**60.88±0.36**	51.07±0.23	**59.91±0.42**	48.73±0.23
w/ sim.	**86.45±0.19**	61.05±0.25	**51.87±0.17**	60.08±0.21	**49.26±0.22**
w/o sim.	86.27±0.17	**61.35±0.17**	51.45±0.21	**60.39±0.17**	48.77±0.26

6.4 Discussion and Limitations

驚くべきことに，ResNet のようにキャパシティが小さいモデルにおいて，通常の Adversarial Training は AA と CW-PGD に対する頑健性が WMMR や MAIL よりも優れていることが 6.3.2 から観測できる．この結果は，十分なキャパシテを持たないモデルに関して，MAIL が強い攻撃に脆弱であることを暗に意味している．そのため，提案手法を MAIL に組み込むことで，PGD に対する頑健性を劇的に改善させる代償として，MAIL 本来が持っている CW や AA に対する潜在的な脆弱性を加速させている可能性が高い．一方，提案手法を組み込んだ WMMR は，式 (6.14) を単純にマージンへ乗算するだけにもかかわらず CW-PGD や AA を含む多くの攻撃に対して WMMR から頑健性が向上させることができる．定数を乗算する代わりに，提案手法は top2 の割合を用いているため，各サンプルに対して異なる係数を乗算することができる．この結果は，提案手法を用いて変換したマージンが学習に直接貢献するため，提案手法を用いてマージンを適切な表現に変換することの有効性をよく表している．

各データセットの結果について，CIFAR-100 における性能向上は CIFAR-10 の結果と比較すると限定的であることが実験より観測できた．直感的には，提案手法は複雑な識別境界が定義されるような，大規模なデータセットで学習するときに重要な役割を果たすはずである．しかし，提案手法は，複数クラス間の関係を間接的に扱っているに過ぎないため，CIFAR-100 の性能向上が限定的だと考えらえる．

6.5 まとめ

本章では，予測確率を用いた従来の IRAT が暗黙的に 2 クラス分類を扱っているため，多クラス分類問題において識別境界とのマージン定義が不十分である問題について取り組んだ．この問題を解決するために，まず，従来の IRAT のマージン定義が不十分であることを具体的な例を用いて明らかにした．この問題点を解消するために，本研究では,「不正解率に含まれる top2 の割合」と「不正解率に対する重要度」を用いて，従来のマージンを多クラス分類に適した表現へ変換する手法を提案した．提案手法を従来法に組み込むことによって，一部の手法を除いて頑健性が向上するだけでなく，通常サンプルに対する分類精度も改善されることが確認できた．ResNet-18 のようにキャパシティが小さいモデルにおいて，WMMR に組み込んだ場合はほぼ全ての頑健性が改善された．

しかし，CIFAR-10 では著しい性能向上が確認できたが，CIFAR-100 は限定的であった．これは，提案手法が全てのクラスとの関係を直接学習に組み込むことができてないためであると考えらえれる．全てのクラスの関係を直接入れ込むための直感的な考えは，マルチクラスマージンを使用することであるが，重み定義関数の設計や各クラスの関係の扱い方など挑戦的な課題が存在する．したがって，今後の課題として，シンプルな設計でマルチクラスマージンを 1 つの重みに集約する方法や，MAIL で使用されているパラメータに依存しない非線形増加関数の慎重な設計などが挙げられる．

第7章

敵対的方策による**mixup**を用いたデータ増幅と学習法

CNN は学習可能なパラメータを適切に更新することで柔軟な画像分類を可能としている．しかし，CNN が持つパラメータは膨大であるため，適切なモデルを得る探索空間があまりにも広すぎる．そのため，モデルを正則化してパラメータ探索の空間を限定するアプローチを使用して学習することが一般的である．Guo ら [146] に基づくと，このモデルに対する正則化は「データに依存する正則化」 [153, 154, 155, 2, 11] と「データに依存しない正則化」 [156, 157, 158] にカテゴリ分けできる．本研究はデータに依存する正則化に着目した研究である．

データに依存する正則化は，学習データに幾何変化などを施すデータ増幅 [153, 154, 155] や，敵対的学習 [2, 11] などが妥当する．敵対的学習は 5 章および 6 章で述べたように，頑健性が向上する反面で通常の分類精度が劣化するため，性能向上目的で使用されることは稀である [159]．そのため，データ増幅はモデルの汎化性能向上を目的として数多くのアプローチが提案されている [160, 16, 161, 133, 162, 163, 123, 164, 134, 165, 166]．3 章および 4 章では，生成画像をデータ増幅に活用することでベースラインの分類性能を向上させることを実現した．特に，2 つ以上の画像を混ぜ合わせて新たなデータを作成するデータ増幅はシンプルながら非常に効果的であるため盛んに研究が進められている．mixup [16] はこれらの研究の先駆けとなった手法である．

mixup は学習データセット $\mathcal{D} := \{(\boldsymbol{x}_i, y_i)\}_{i=1}^n$ から，2 つのサンプル $(\boldsymbol{x}_i, y_i)$，$(\boldsymbol{x}_j, y_j)$ を取り出し，任意の比率 $\lambda \sim \mathrm{Beta}(\alpha, \beta)$ で入力データと教師信号を線形補間する．mixup は優れた性能を獲得できる反面で，manifold intrusion [146] が生じやすいためベータ分布のパラメータを慎重に決定する必要がある．manifold intrusion は mixup によって合成データが別の多様体内部に侵入して，別のクラスと衝突することである．直感的には，manifold intrusion が生じたデータを正確に分類することができれば，優れた分類精度の獲得が期待できる．Adaptive Mixup (AdaMixup) [146] は，合成する 2 つのデータをネットワークに入力して獲得した内挿比で mixup することで通常の mixup よりも優れた性能を達成した．AdaMixup は優れた性能が得られるが，分類器以外にもネットワークが必要になることに加えてデータ入力方法が複雑である．MetaMixup [167] ではメタ学習 [107] を用いて AdaMixup よりもシンプルな学習方法で，AdaMixup を上回る分類精度を達成した．Metamixup では，manifold intrusion を避けるような内挿比が獲得される．

しかし，AdaMixup や MetaMixup は通常の mixup より優れた分類ができる一方，複雑な学習方法や分類器以外にもネットワークが必要になる点で拡張性が乏しい．そこで本研究では，図 7.1(b) に示すように，敵対的方策を利用して損失が最大となる内挿比を求め (inner maximization)，損失が最

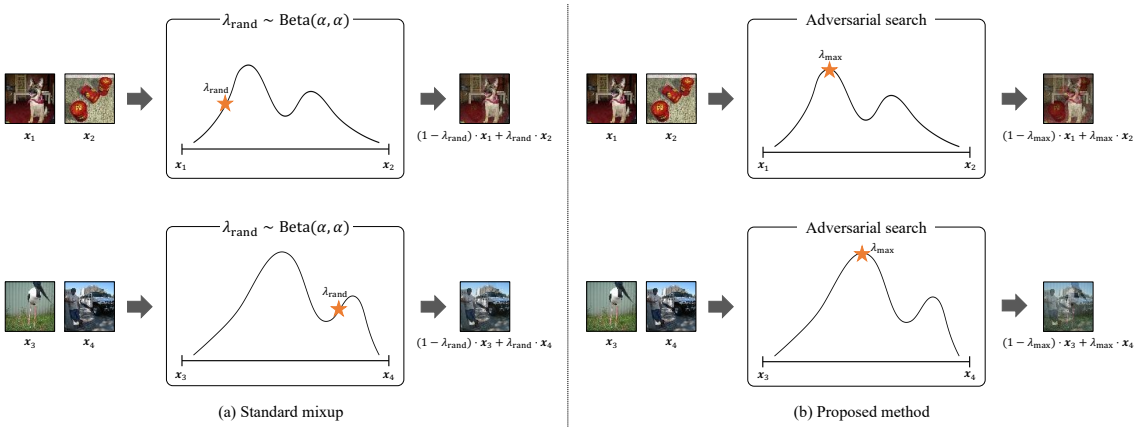


図 7.1: 通常の mixup と提案手法の内挿比のサンプリング方法の違い．

小となるようにモデルパラメータを更新する (outer minimization)．提案手法では，古典的な最小値探索方法を逆向きに利用して内挿比を求める．まとめると，本研究は以下の貢献をしている．

- コンピュータビジョン分野において，敵対的方策を利用した mixup は存在していないため，本研究のアプローチがどのような振る舞いをするか未知である．したがって，本研究はさらに mixup を発展させるために有益な情報を提供することができる．
- AdaMixup や MetaMixup よりもシンプルなアプローチで同程度またはそれ以上の性能を実現できる．
- 提案手法はあらゆるデータセットやベースネットワークにおいて優れた分類性能を獲得できる．

7.1 予備知識と関連研究

Notations. 本研究では，画像 $\boldsymbol{x}_i \in \mathbb{R}^{c\times h\times w}$ と $\boldsymbol{x}_i$ に対する教師信号 $y_i \in \mathcal{Y} := \{0, 1, \ldots, k-1\}$ を n サンプル含むデータセット $\mathcal{D} := \{\boldsymbol{x}_i, y_i\}_{i=1}^n$ を扱う．さらに，画像空間からラベル空間に写像する $\boldsymbol{\theta}$ でパラメータ化されたモデル $f : \mathbb{R}^{c\times h\times w} \to \mathbb{R}^k$ を扱う．推論結果に対する損失関数は，次のクロスエントロピー誤差を使用する:

$$L(f_{\boldsymbol{\theta}}, \mathcal{D}) \quad = \quad \frac{1}{|\mathcal{D}|} \sum_{(\boldsymbol{x}_i, y_i)\in\mathcal{D}} -\log \sigma_{y_i}\left(f_{\boldsymbol{\theta}}(\boldsymbol{x}_i)\right), \tag{7.1}$$

ここで，$\sigma : \mathbb{R}^k \to [0,1]^k$ は softmax 関数であり，σ_{y_i} は正解クラス確率を表している．

mixup [16]: mixup は 2 つのデータセットを任意の内挿比で混ぜ合わせることで，新たなデータを作るデータ増幅である．mixup はベータ分布 $\mathrm{Beta}(\alpha, \beta)$ から取り出した任意の確率 $\lambda \in [0,1]$ を用いて次のように計算する．

$$\tilde{\boldsymbol{x}} \quad := \quad \lambda \cdot \boldsymbol{x}_i + (1-\lambda)\cdot \boldsymbol{x}_j, \tag{7.2}$$

$$\tilde{\boldsymbol{y}} \quad := \quad \lambda \cdot \boldsymbol{y}_i + (1-\lambda)\cdot \boldsymbol{y}_j, \tag{7.3}$$

ここで，$\boldsymbol{y}_i$ と $\boldsymbol{y}_i$ は正解クラスが 1 でそれ以外が 0 の one-hot ベクトルである．したがって，線形補間したデータペア集合を $\tilde{\mathcal{D}} := \{\tilde{\boldsymbol{x}}_i, \tilde{\boldsymbol{y}}_i\}_{i=1}^n$ とすると，mixup では次の最適化式を解くことになる．

$$\min_{\boldsymbol{\theta}} \mathbb{E}_{(\tilde{\boldsymbol{x}}_i, \tilde{\boldsymbol{y}}_i)\sim\tilde{\mathcal{D}}} \left[L'(f_{\boldsymbol{\theta}}(\tilde{\boldsymbol{x}}_i), \tilde{\boldsymbol{y}}_i),\right] \tag{7.4}$$

ここで，$L'(f_{\boldsymbol{\theta}}(\tilde{\boldsymbol{x}}_i), \tilde{\boldsymbol{y}}_i) := \lambda \cdot L(f_{\boldsymbol{\theta}}(\boldsymbol{x}_i), y_i) + (1-\lambda)\cdot L(f_{\boldsymbol{\theta}}(\boldsymbol{x}_j), y_j)$ である．

7.1.1 データ増幅

データ増幅は膨大な学習データを必要とする CNN に多様なデータを仮想的に学習させるためのテクニックであり，古典的なものは既存データに幾何変化 (例えば，回転，並進やノイズ付与など) を施す．一方，mixup [16] のように既存データを合成して新たなデータを作成するデータ増幅も提案されており，mixup を参考に派生した方法 [162, 168, 169, 170, 171]，画像の一部を欠落させる方法 [160, 123]，画像の一部を混ぜ合わせる方法 [133, 172, 134, 173, 174] にカテゴライズできる．

Manifold mixup [162] は特徴量レベルで mixup することで，入力データを mixup する従来法よりも優れた分類精度を達成している．AugMix [175] は 2 つのサンプルを合成するのではなく，1 つのデータに様々な種類の幾何変化を施して，元データを混ぜ合わせる mixup である．SmoothMix [168] は画像全体を内挿比によって一様に等しく合成するのではなく，ガウス分布に基づいて画像を混ぜ合わせる．Co-Mixup [169] や Puzzle Mix [170] は 2 つの画像の顕著性領域をもとに合成する領域を決定している．クラスごとのデータ数が等しいデータセットに対する mixup だけでなく，クラスごとのデータ数が不均衡なデータセットに適した ReMix [171] も提案されている．

さらに，mixup で問題視されている manifold intrusion に対する研究も進められている．Adaptive Mixup (AdaMixup) [146] は，ベータ分布に従う内挿比ではなく，任意のネットワークから適切な内挿比を獲得して mixup する．この時，manifold intrusion が生じているか否かを判定するネットワークを用いることで，モデルに強い正則化をかけている．MetaMixup [167] は AdaMixup の複雑なネットワーク設計を改善するために，メタ学習 [107] を用いて適切な内挿比を決定する．Local Mixup [166] は，遠くのサンプルと mixup した時に manifold intrusion が生じやすいと捉えて，k-nearest neighbor によって求めた距離をもとに近傍サンプルと mixup する．さらに，Sohn ら [165] は manifold intrusion だけでなく，ラベル空間で線形な振る舞いをすることも mixup において問題であることを主張し，線形補間で得た教師信号に更なる変化を施して非線形にする GenLabel を提案した．

従来の mixup を含む多くの派生手法が，どのようにデータを合成するかに着目している．一方，本研究はデータの合成方法ではなく，AdaMixup や MetaMixup と同様に内挿比の決定方法に着目する．MetaMixup は AdaMixup の複雑性を解消して優れた分類性能を獲得しているが，ネットワークを介して内挿比を出力しており，学習したいモデル以外にも別のモデルを用意する必要がある．また，計算方法などまだ複雑な処理が残っている．そこで本研究では，任意の閉区間において損失が極大値となる内挿比を用いて mixup を行い，学習することで従来法よりも優れた分類性能を目指す．直感的には，最大の損失を最小化するようにモデルパラメータを更新することで，モデルの汎化性能が向上することが想定される．以降の章で提案手法について詳細に述べる．

7.2 Adversarial Interpolating Policy

本章では，まず提案手法の概要を 7.2.1 述べて，7.2.2 で提案手法を詳細に説明する．

7.2.1 提案手法の概要

本研究では，データセット $\mathcal{D}$ から取り出したサンプルペア $(\boldsymbol{x}_i, y_i)$ と $(\boldsymbol{x}_j, y_j)$ に関して，閉区間 $\Lambda := [0, 1]$ で損失が極大値となる内挿比 λ_{max} を求める．つまり，内挿比 λ に対して敵対的攻撃を行い極大値を求める．したがって，本研究では以下の最適化式を解くことになる．

$$\min_{\boldsymbol{\theta}} \mathbb{E}_{(\boldsymbol{x}_i, \boldsymbol{y}_i), (\boldsymbol{x}_j, \boldsymbol{y}_j) \sim \mathcal{D}} \left[\max_{\lambda \in \Lambda} L'(f_{\boldsymbol{\theta}}(\lambda \boldsymbol{x}_i + (1-\lambda)\boldsymbol{x}_j), \lambda \boldsymbol{y}_i + (1-\lambda)\boldsymbol{y}_j) \right] \tag{7.5}$$

Adversarial Training 同様に，inner maximization と outer minimizarion を含むが，内挿比を変化させて損失を最大化する点で異なる．敵対的攻撃の場合，任意のノルム空間で微小な摂動を探索するため FGSM や PGD のようなアプローチが効率的である．つまり，摂動を加える対象が高次元であるため，計算的に摂動を求めることが困難である．一方，内挿比 λ は 1 次元のスカラー値であるため，古典的な最小値探索方法を逆向きに利用することで，任意のデータペアにおいて損失を最大にする内挿比の近似解を求めることができる．本稿では，この提案手法を Adversarial Interpolating Policy (AIP) と呼称する．

Algorithm 3 searching mixup policy with the ternary search

Require: Sample pair $\{(\boldsymbol{x}_i, y_i), (\boldsymbol{x}_j, y_j)\}$, model $f_{\boldsymbol{\theta}}$, the number of search K

Upper bound: $\lambda_u \leftarrow 1 + \zeta$
Lower bound: $\lambda_l \leftarrow 0 - \zeta$
while $K > 0$ **do**
  $c_1 \leftarrow (2\lambda_l + \lambda_u)/3$
  $c_2 \leftarrow (\lambda_l + 2\lambda_u)/3$
  data interpolation with c_1 and c_2:
  $\tilde{\boldsymbol{x}}_1 \leftarrow c_1 \cdot \boldsymbol{x}_i + (1 - c_1) \cdot \boldsymbol{x}_j$
  $\tilde{\boldsymbol{x}}_2 \leftarrow c_2 \cdot \boldsymbol{x}_i + (1 - c_2) \cdot \boldsymbol{x}_j$
  compute loss for $\tilde{\boldsymbol{x}}_1$ and $\tilde{\boldsymbol{x}}_2$:
  $L'_1 \leftarrow c_1 \cdot L(f_{\boldsymbol{\theta}}(\tilde{\boldsymbol{x}}_1), y_i) + (1 - c_1) \cdot L(f_{\boldsymbol{\theta}}(\tilde{\boldsymbol{x}}_1), y_j)$
  $L'_2 \leftarrow c_2 \cdot L(f_{\boldsymbol{\theta}}(\tilde{\boldsymbol{x}}_2), y_i) + (1 - c_2) \cdot L(f_{\boldsymbol{\theta}}(\tilde{\boldsymbol{x}}_2), y_j)$
  if $L'_1 \geq L'_2$ **then**
    $\lambda_u \leftarrow c_2$
    $\lambda_l \leftarrow \lambda_l$
  else if $L'_1 < L'_2$ **then**
    $\lambda_u \leftarrow \lambda_u$
    $\lambda_l \leftarrow c_1$
  end if
  $K \leftarrow K - 1$
end while
$\lambda \leftarrow (\lambda_u + \lambda_l)/2$
return interpolation ratio λ

7.2.2 三分探索を用いた極大値探索

mixup では，λ で線形補間した画像 $\tilde{\boldsymbol{x}}$ の予測分布に関して y_i と y_j それぞれでクロスエントロピー誤差を計算して λ によって線形補間したものを損失とする．したがって，極大値はたかだか 1 つしか存在しないことが想定されることから，三分探索を用いて損失が最大となる λ を近似的に求める．提案手法の内挿比探索手順は図 7.2 に示す通りである．

三分探索では，任意の閉区間における探索を行う特性上，まず上限値 λ_u と下限値 λ_l を設定する必要がある．mixup の内挿比は $[0, 1]$ の閉区間で定義されるため，$\lambda_u = 1$ と $\lambda_l = 0$ とすることが直感的に考えられる．三分探索は次式の地点 c_1，c_2 が次の上限値または下限値の候補となる．

$$c_1 = \frac{2\lambda_l + \lambda_u}{3} \tag{7.6}$$

$$c_2 = \frac{\lambda_u + 2\lambda_l}{3} \tag{7.7}$$

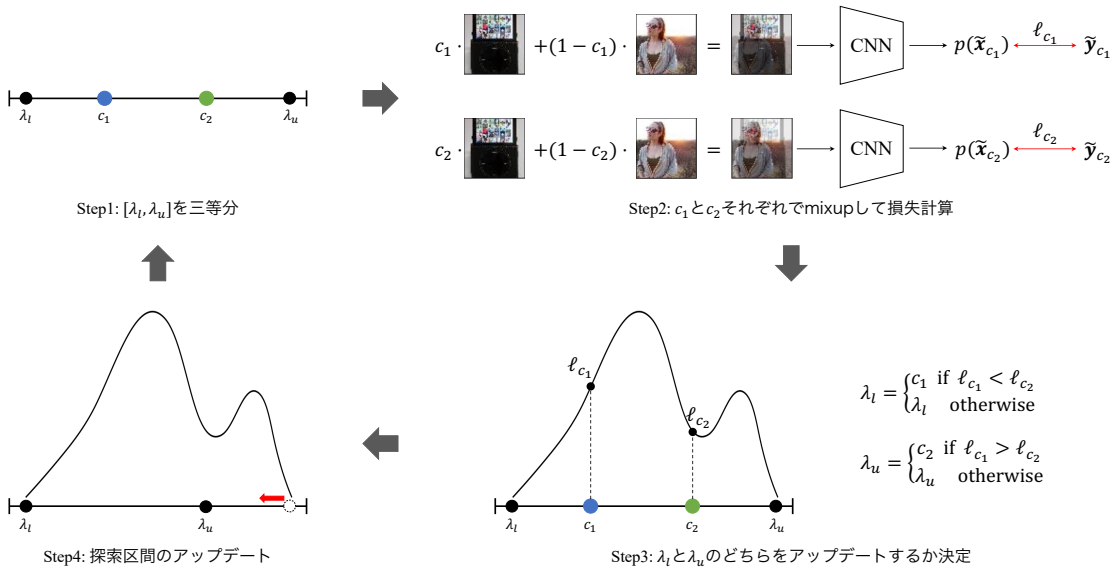


図 7.2: 三分探索を用いた内挿比探索.

従って，$[\lambda_l, c_1)$ または $(c_2, \lambda_u]$ のどちらか一方の範囲は探索されない．しかし，どちらの範囲にも極大値が存在する可能性があるため，本研究では探索空間にダミー領域を追加する．具体的には，任意の係数 ζ を利用して，上限値と下限値の初期値をそれぞれ $\lambda_u = 1 + \zeta$ と $\lambda_l = 0 - \zeta$ と定義して探索範囲を拡張する．この時 $\zeta = 1$ とすることで，初期の上限値と下限値の候補が $c_1 = 0$ と c_1 となるため，$[0, 1]$ の空間を全て探索したことになる．

次に，求めた c_1 と c_2 を用いて，学習用データセット $\mathcal{D}$ から取り出したサンプルペア $(\boldsymbol{x}_i, y_i)$ と $(\boldsymbol{x}_j, y_j)$ を以下のように mixup する．

$$\tilde{\boldsymbol{x}}_1 = c_1 \cdot \boldsymbol{x}_i + (1 - c_1) \cdot \boldsymbol{x}_j \tag{7.8}$$

$$\tilde{\boldsymbol{x}}_2 = c_2 \cdot \boldsymbol{x}_i + (1 - c_2) \cdot \boldsymbol{x}_j \tag{7.9}$$

mixup して求めた $\tilde{\boldsymbol{x}}_1$ と $\tilde{\boldsymbol{x}}_2$ に対する予測分布 $f_{\boldsymbol{\theta}}(\tilde{\boldsymbol{x}}_1)$ と $f_{\boldsymbol{\theta}}(\tilde{\boldsymbol{x}}_2)$ に関して以下のようにクロスエントロピー誤差を計算する．

$$L'_1 = c_1 \cdot L(f_{\boldsymbol{\theta}}(\tilde{\boldsymbol{x}}_1), y_i) + (1 - c_1) \cdot L(f_{\boldsymbol{\theta}}(\tilde{\boldsymbol{x}}_1), y_i) \tag{7.10}$$

$$L'_2 = c_2 \cdot L(f_{\boldsymbol{\theta}}(\tilde{\boldsymbol{x}}_2), y_i) + (1 - c_2) \cdot L(f_{\boldsymbol{\theta}}(\tilde{\boldsymbol{x}}_2), y_i) \tag{7.11}$$

最後に，求めた損失の関係から上限値 λ_u と λ_l を以下のように更新する．

$$\lambda_u = \begin{cases} c_2 & \text{if } L'_1 \geq L'_2 \\ \lambda_u & \text{otherwise} \end{cases}, \quad \lambda_l = \begin{cases} c_1 & \text{if } L'_1 < L'_2 \\ \lambda_l & \text{otherwise} \end{cases} \tag{7.12}$$

この時，λ_u および λ_l は $[0,1]$ の範囲を超えないようにクリッピングする．これらの処理を任意の回数 K 回反復して，最終的な λ_{max} は以下のように上限値と下限値の中点として定義する．

$$\lambda_{max} = \frac{\lambda_u + \lambda_l}{2} \tag{7.13}$$

ここまでの流れは，Algorithm 3 に示す通りである．

提案手法は入力画像に対する従来の mixup だけでなく，特徴空間に対する Manifold mixup にも容易に応用可能である．また，AdaMixup や MetaMixup のように複雑なネットワーク設計を必要としないだけでなく，反復回数 K を決めるだけで学習ができる点でさまざまな手法に応用が可能である．経験的には，$K = \{6, 7, 8\}$ で極大値に収束することがわかっている．

7.2.3 敵対的方策を適用する位置

通常の mixup は多くのデータセットにおいて $\mathrm{Beta}(1,1) = U[0,1]$ から取り出した確率を用いた場合の精度が優れている一方，Manifold mixup [162] は $\mathrm{Beta}(2,2)$ を用いた場合に最高性能を達成する．言い換えると，Manifold mixup は 0.5 付近，つまり最大の損失付近の内挿比でデータを合成することが有効であることを示している．そのため，Manifold mixup を参考に AIP も入力層を含む中間層で使用する．Manifold mixup は任意の 2 つのデータに対してランダムに内挿比が決定される一方，AIP は損失が最大となる内挿比を厳密に求めることが可能となる．

次に，入力層における mixup について議論する．自然言語処理の分野で提案されている，FGSM 攻撃を用いて内挿比に微小な摂動を付与する Adversarial Mixing Policy (AMP) [176] は入力層に適用した場合でも性能向上することが報告されている．文章は任意のベクトルに埋め込んでからネットワークで処理するため，入力層において manifold intrusion が生じづらいと考えられる．しかし，画像処理分野において損失が大きくなる内挿比で合成することは，manifold intrusion 発生の可能性が高くなり，性能劣化につながる．従って，提案手法は入力層を除いた中間層に適用する．入力層における mixup は従来の mixup と同様でベータ分布から取り出した内挿比でデータを合成する．

7.3 評価実験

本章では従来手法と比較することで提案手法の有効性を示す．まず，7.3.3 で実験の設定について詳細に述べて，7.3.2 で従来法と性能比較する．7.3.3 と 7.3.4 では，それぞれ，内挿比探索回数の違いと入力層にも提案手法を含めた場合の分類性能について議論する．

7.3.1 実験の詳細な設定

本実験では，mixup を使用してないモデルの性能をベースラインとして，mixup，Manifold mixup，MetaMixup と提案手法を性能比較する．

Algorithm 4 Adversarial Interpolating Policy

Require: Training dataset $\mathcal{D}$, batch size m, training epochs T, learning rate η, model $f_{\boldsymbol{\theta}}$, hyper-parameter for beta distribution α, the number of search K

Require: Ternary search function ψ

for $t = 1, 2, \ldots, T$ **do**
  for mini-batch $\mathcal{S} := \{\boldsymbol{x}_i, y_i\}_{i=1}^m \sim \mathcal{D}$ **do**
    $l \sim \{0, 1, 2, 3\}$
    if $l = 0$ **then**
      $\lambda \sim \mathrm{Beta}(\alpha, \alpha)$
    else
      $\mathcal{S}' \leftarrow \mathrm{Shuffle}(\mathcal{S})$
      $\lambda \leftarrow \psi(f_{\boldsymbol{\theta}}, \mathcal{S}, \mathcal{S}', K, l)$ % Compute interpolation ratio with ternary search in Algorithm 3
    end if
    $\tilde{\mathcal{Y}} \leftarrow \{\tilde{\boldsymbol{y}}_i \mid \lambda_i \cdot y_i + (1 - \lambda_i) \cdot y'_i\}_{i=1}^m$
    % $\mathcal{Y} := \{y_i \mid y_i \in \mathcal{S}, i = 1, \ldots, m\}$, $\mathcal{Y}' := \{y'_i \mid y'_i \in \mathcal{S}', i = 1, \ldots, m\}$
    $\tilde{\mathcal{X}} \leftarrow \{\tilde{\boldsymbol{x}}_i \mid \lambda_i \cdot f_{\boldsymbol{\theta}}^{(0:l)}(\boldsymbol{x}_i) + (1 - \lambda_i) \cdot f_{\boldsymbol{\theta}}^{(0:l)}(\boldsymbol{x}'_i)\}_{i=1}^m$
    % $\mathcal{X} := \{\boldsymbol{x}_i \mid \boldsymbol{x}_i \in \mathcal{S}, i = 1, \ldots, m\}$, $\mathcal{X}' := \{\boldsymbol{x}'_i \mid \boldsymbol{x}'_i \in \mathcal{S}', i = 1, \ldots, m\}$
    $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \eta \cdot \frac{1}{m} \sum_{i=1}^m \nabla_{\boldsymbol{\theta}} L(f_{\boldsymbol{\theta}}^{(l:)}(\tilde{\boldsymbol{x}}), \tilde{\boldsymbol{y}}_i)$
  end for
end for
return $\boldsymbol{\theta}$

データセット： 本実験では，CIFAR-10/100，Street View House Number (SVHN) と Tiny ImageNet を使用する．CIFAR-10 は 10 クラスの自然画像データセットであり，学習用として 50,000 枚，推論用として 10,000 枚のデータが用意されている．画像サイズは 32×32，各クラスのデータ数は学習用が 5,000 枚，推論データが 1,000 枚用意されている．CIFAR-100 は各クラスのデータ数が学習用として 500 枚，推論データとして 100 枚用意されていることを除いて CIFAR-10 と同じである．SVHN は Google Street View から収集した表札の数字データである．クラス数は 0 から 9 までの 10 クラスあり，画像の中心に映る数字に対して教師信号が付与されている．SVHN は訓練データとして 73,257 枚の RGB 画像とテスト用として 26,032 枚梱包されている．さらに SVHN は，extra data として 531,131 枚の膨大な拡張データが含まれている．画像サイズは CIFAR10 と同様である．Tiny ImageNet は 64×64 の一般物体認識用データセットであり，200 クラスそれぞれに 500 枚のデータが学習用として用意されている．検証用とテスト用画像は，各クラス 50 枚用意されている．

学習条件： 本実験では，18 層と 34 層の PreActResNet [177] と WideResNet28-10 (WRN28-10) [148] を使用して，全てのデータセットで PreActResNet-18/34 のとき 500 エポック，WRN28-10 のとき 400

表 7.1: PreActResNet18 における分類性能比較．† は論文から引用した値である．

	Last model				Best model			
	CIFAR-10	CIFAR-100	SVHN	Tiny ImageNet	CIFAR-10	CIFAR-100	SVHN	Tiny ImageNet
Baseline	95.16±0.14	75.48±0.36	97.07±0.07	57.46±0.34	95.28±0.10	75.75±0.33	97.11±0.05	57.84±0.16
mixup	96.32±0.08	77.28±0.37	97.38±0.06	60.69±0.66	96.50±0.07	78.02±0.31	97.51±0.06	61.16±0.39
Manifold mixup	96.47±0.11	78.82±0.17	97.73±0.04	61.35±0.38	96.68±0.08	79.22±0.10	97.88±0.02	61.71±0.32
MetaMixup†	–	–	–	–	96.88±0.09	79.64±0.16	97.04±0.06	–
AIP	**96.88±0.18**	**79.50±0.23**	**97.77±0.06**	**63.78±0.14**	**96.97±0.16**	**79.96±0.29**	**97.91±0.04**	**64.27±0.18**

エポック学習する．Tiny ImageNet は PreActResNet-18 のみで評価する．最適化手法は，初期学習率が 0.1，momentum が 0.9 の確率的勾配降下法 (SGD) を使用する．学習率は全ての手法で PreActResNet のとき $\{250, 375\}$ エポック，WRN28-10 のとき $\{200, 300\}$ エポックで $1/10$ に減衰し，weight decay は 1×10^{-4} を使用する．CIFAR-10/100 と SVHN の時，mixup は $\mathrm{Beta}(1, 1)$，Manifold mixup は $\mathrm{Beta}(2, 2)$ から内挿比をサンプリングする．Tiny ImageNet の時は，mixup と Manifold mixup 共に $\mathrm{Beta}(0.2, 0.2)$ を使用する．提案手法の探索回数は SVHN のとき $K = 8$，CIFAR-10 のとき $K = 6$，CIFAR-100 と Tiny ImageNet のとき $K = 4$ として，入力層における内挿比は mixup と同様の設定を使用する．全ての手法で，Tiny ImageNet はクロスエントロピー，それ以外のデータセットはバイナリクロスエントロピーを用いて損失計算する．各手法，異なる 5 つの乱数シードを用いて学習した各モデルの精度の平均と偏差を用いて比較する．

7.3.2 精度比較結果

PreActResNet18 の結果を表 7.1 に示す．Last model および Bets model 共に，提案手法が全てのデータセットで最も優れた性能を達成していることが確認できる．提案手法と Manifold mixup を比較すると，CIFAR-10 の Last model で 0.41 ポイント，Best model で 0.29 ポイント，SVHN の Last model で 0.04 ポイント，Best model で 0.03 ポイント，CIFAR-100 の Last model で 0.68 ポイント，Best model で 0.74 ポイント，Tiny ImageNet の Last model で 2.43 ポイント，Best model で 2.56 ポイント分類精度が向上した．

次に，表 7.2 の PreActResNet34 の結果に着目すると，CIFAR-10 の Best model の結果を除いて，提案手法の分類精度が最も優れていることが確認できる．CIFAR-10/100 に関して，Manifold mixup は PreActResNet18 の結果よりも劣化しているが，提案手法は更なる精度向上が確認できる．WRN28-10 の Last model の結果は Manifold mixup に劣る分類精度であるが，Best model は全てのデータセットにおいて優れた分類精度を達成した．これらの実験結果に従うと，提案手法が高精度な画像分類のためのデータ増幅として優れていることがわかる．特に，複雑なデータ，つまりクラス数の増加に伴って，提案手法が有効に働くことが判明した．

表 7.2: PreActResNet34 と WideResNet28-10 における分類性能比較．† は論文から引用した値である．

	Last model			Best model		
PreActResNet34						
	CIFAR-10	CIFAR-100	SVHN	CIFAR-10	CIFAR-100	SVHN
Baseline	95.36±0.17	76.36±0.23	97.13±0.05	95.48±0.14	76.69±0.25	97.20±0.05
mixup	96.74±0.13	78.74±0.40	97.20±0.16	96.94±0.08	79.29±0.20	97.60±0.03
Manifold mixup	96.98±0.12	81.10±0.23	97.69±0.04	97.18±0.12	81.36±0.16	97.85±0.05
MetaMixup†	–	–	–	**97.49±0.15**	81.49±0.20	97.55±0.07
AIP	**97.06±0.16**	**81.52±0.43**	**97.96±0.08**	97.26±0.09	**82.00±0.30**	**98.11±0.05**
WideResNet28-10						
Baseline	96.00±0.16	78.72±0.35	97.38±0.05	96.09±0.11	78.82±0.30	97.48±0.06
mixup	97.21±0.12	81.65±0.21	97.49±0.08	97.38±0.08	81.98±0.11	97.80±0.04
Manifold mixup	**97.40±0.13**	**81.83±0.41**	**97.83±0.03**	97.51±0.07	82.05±0.33	97.99±0.03
MetaMixup†	–	–	–	97.52±0.10	81.55±0.17	97.67±0.08
AIP	97.27±0.10	81.73±0.34	97.76±0.20	**97.57±0.03**	**82.36±0.30**	**98.04±0.07**

表 7.3: 各データセットにおける内挿比探索回数の違いによる分類精度．

	Number of rounds					
	2	4	6	8	10	20
SVHN	97.35±0.03	97.80±0.03	97.80±0.02	**97.91±0.04**	97.87±0.07	97.85±0.06
CIFAR-10	96.11±0.10	96.92±0.07	**96.97±0.16**	96.89±0.07	96.83±0.07	96.86±0.11
CIFAR-100	77.34±0.28	**79.96±0.29**	79.73±0.23	79.64±0.13	79.47±0.23	79.38±0.29
Tiny ImageNet	60.24±0.50	**64.27±0.18**	64.18±0.45	63.75±0.15	64.05±0.32	64.00±0.20

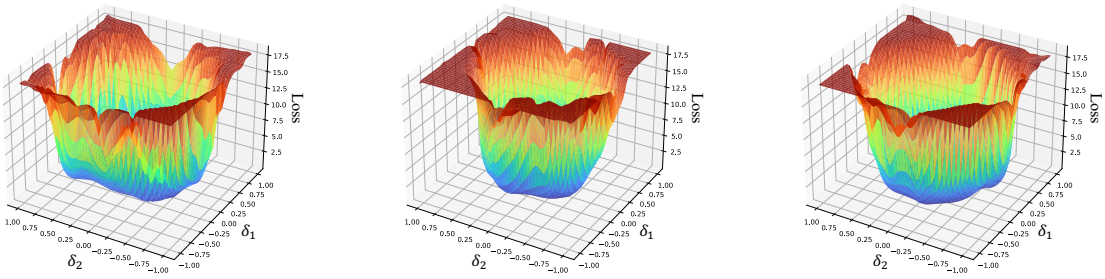
7.3.3 内挿比探索回数の違いによる分類精度

本項では，PreActResNet18 における異なる内挿比探索回数の分類精度について議論する．本実験ではで述べた設定を使用する．内挿比探索は {2,4,6,8,10,20} 回，それぞれで異なる 5 つの乱数シードで学習した Best model の平均精度で比較する．

表 7.3 に各データセットの探索回数の違いによる分類精度を示す．実験結果より，SVHN は 8 回が最高性能であるが，CIFAR-10 では 6 回のとき最高性能である．さらに，より複雑でクラス数が多い CIFAR-100 や Tiny ImageNet は 4 回の探索が最も優れた分類精度を得ることができる．これらの結果より，分類クラス数が多いまたは，データセットが複雑なほど少ない探索回数で最高性能を達成できる．考え方を変えれば，1 クラスあたりのデータ数が少ないほど，少ない探索回数で十分な分類精度を達成できると捉えることもできる．

表 7.4: 入力層における提案手法の有無による分類精度比較.

	Last model				Best model			
Input mixup	CIFAR-10	CIFAR-100	SVHN	Tiny ImageNet	CIFAR-10	CIFAR-100	SVHN	Tiny ImageNet
✓	96.55±0.14	77.85±0.53	97.41±0.24	61.92±0.25	**96.89±0.08**	78.24±0.31	97.85±0.07	63.19±0.18
	96.73±0.08	**79.11±0.06**	**97.77±0.06**	**62.97±0.40**	**96.89±0.07**	**79.64±0.13**	**97.91±0.04**	**63.75±0.15**



(a) mixup (b) Manifold mixup (c) Adversarial mixup

図 7.3: CIFAR-10 を学習した PreActResNet18 の誤差曲面.

7.3.4 入力層における提案手法の効果と誤差曲面の可視化

表 7.4 は入力層においても内挿比探索した場合と，提案手法の精度を示している．すべてのデータセットにおいて，異なる 5 つのランダムシードで学習した PreActResNet18 の結果である．表 7.4 から読み取れる通り，クラス数が少ない CIFAR-10 や SVHN では入力層で内挿比探索したとしても同程度であり，著しい性能劣化はない．一方，クラス数が増加するにつれて，CIFAR-100 では 1 ポイント以上，Tiny ImageNet では 0.5 ポイント以上の分類性能劣化が観測された．従って，厳しい内挿比を入力層で使用することは分類クラス数が多いデータセットほどモデルの underfitting を招く原因となる．

次に各手法で学習したモデルの誤差曲面について議論する．誤差曲面は Li ら [178] の手法を用いて可視化する．図 7.3 の各誤差曲面において，x, y 軸は重みパラメータに加算するノイズの強さ，z 軸は損失を表しており，$(x, y) = (0, 0)$ が学習によって獲得した重みパラメータである．図 7.3(c) の提案手法は，図 7.3(b) の Manifold mixup と比較して底面が広いことが確認できる．また，図 7.3(a) の mixup は提案手法よりも δ_2 方向に長く，シャープな曲面である．このことから，提案手法の誤差曲面は滑らかで，学習済みパラメータ付近だけでなく広範囲で損失が低いため，優れた分類精度を達成できたといえる．

7.4 Discussion and Limitations

7.3.2 で述べたように，提案手法はあらゆるデータセットにおいて従来法よりも優れた性能を達成できることがわかった．特に，Tiny ImageNet に関して，mixup や Manifold mixup では，$\mathrm{Beta}(0.2, 0.2)$ の時が最高性能にも関わらず，提案手法を用いた場合が最も優れている．提案手法は，入力層のみ

表 7.5: 従来法と提案手法の処理速度．($K = 8$)

	mixup	Manifold mixup	AIP
elapsed time [s]	0.035004	0.034844	0.175844

においてベータ分布から取り出した内挿比を用いているものの，中間層では損失が最大となる内挿比を意図的に求めて mixup している．ベータ分布のパラメータを操作して，0.5 付近の値の出現頻度を高くしても，ランダムなサンプリングされるため学習に貢献する合成できる保証がない．これは，学習に高く貢献する内挿比が各データペアによって異なることを表している．したがって，そのような内挿比を厳密に求めて学習することは，分類精度向上に重要な役割を担っているといえる．

提案手法は優れた性能を達成できる反面，三分探索によって内挿比を求めるため探索回数が増加するにつれて処理時間が増加する．経験的には，$K = 8$ で損失が最大値となる内挿比に収束することがわかっている．表 7.5 に Tesla V100 を 1 枚使用して， 128 のバッチサイズ CIFAR-10 データセットを PreActResNet18 で学習した際の 1 バッチに要する処理時間を示す．表 7.5 の提案手法は $K = 8$ を使用しているが，通常の mixup や Manifold mixup と比較して 5 倍以上の処理時間を必要とする．つまり，データ数の増加や深いネットワーク構造ではさらに高い計算コストが必要となることは明らかである．提案手法をさらに効率化するためには，三分探索の高速化や三分探索に代わる高速なアプローチを考案する必要がある．

7.5 まとめ

本章では，データ増幅の一つである mixup において，学習に最も貢献すると考えられる内挿比を用いて学習する新たな mixup を提案した．mixup は任意の確率分布から取り出した内挿比を用いて 2 つのデータをと教師信号を合成して学習する強力なデータ増幅であるが，0.5 付近の内挿比によって合成したデータは manifold intrusion が発生し，分類精度の劣化を招く．この問題を解消するために，適切な内挿比で mixup する AdapMixup や MetaMixup が提案されているが，複雑なネットワーク設計や計算を含むため非常に拡張性が乏しい．直感的には，常に損失が最も高くなる内挿比を用いてデータを合成して学習することで，mixup の効果を最大限に引き出すことができると考えられる．そこで，本研究では，Adversarial Training で用いられる敵対的方策を参考に，損失を最大にする内挿比を意図的に求め，求めた内挿比で合成したデータに対する損失を最小化する Adversarial Interpolating Policy (AIP) を提案する．内挿比は高次元な画像と異なり $[0, 1]$ のスカラー値で定義されるため，三分探索を用いて近似的に計算する．先述したように，入力画像を 0.5 付近の確率で mixup すると manifold intrusion が生じやすいため，Manifold mixup と同様に中間層に提案手法を適用する．提案手法は AdaMixup や MetaMixup のような複雑な計算なしで，あらゆるデータセットとベースネットワークにおいて従来法よりも優れた性能を達成した．

しかし，提案手法は三分探索の探索数が増加すると，モデルパラメータ更新までに要するモデルの順伝播処理が増加する．本実験中によって，データが複雑かつ分類クラス数が多いほど，少ない探

索回数で最高性能が達成できることが確認できたが，通常の mixup や Manifold mixup の 2 倍以上の処理時間を要する．これは大規模なデータセットや巨大なネットワークを用いた場合に致命的な問題となる．従って，提案手法の高速化や三分探索に代わる画期的な最大値探索方法を提案することが今後の課題として挙げられる．

第8章

結論と展望

本稿では，敵対的攻撃に脆弱な CNN の頑健性を向上させるために，Adversarial Training に着目して研究を行い，多様な Adversarial Examples を学習する手法と多クラス分類問題に適した Instance-Reweighted Adversarial Training を提案した．さらに，敵対的方策と強力なデータ増幅の 1 種である mixup を組み合わせた新たな mixup を提案した．以下に，本論文の結論と今後の展望について述べる．

8.1 結論

各章のまとめは以下の通りである．2 章では画像処理分野における敵対的深層学習を，画像生成における敵対的方策と CNN の脆弱性を緩和するための敵対的方策にカテゴライズして，それぞれについて詳細に述べた．敵対的方策を用いた画像生成は，Generative Adversarial Networks (GAN) と呼ばれており，潜在変数を用いて画像生成する Generator と入力画像を真贋判定する Discriminator を敵対させることによって，訓練データセット内を補間するような画像生成が可能である．GAN は学習方法やモデル設計が改善されており，現在では人間の目では生成画像と実画像の区別ができないほど高精細な画像生成を可能としている．特に，モデルを成長させながら画像生成する Progressive Growing GAN (PGGAN) から画像生成に信号処理の概念を組み込んだ StyleGAN3 までの発展は顔画像生成において非常に印象深い．CNN の脆弱性を緩和するための敵対的方策は，Adversarial Training と呼ばれ，優れた頑健性が得られる一方，ロバスト過適合や通常サンプルに対する分類精度が劣化する．Adversarial Training は基本的にモデル構造に依存しない，つまりデータの入力方法や損失の計算方法などが改善されることが多く，特に識別境界とサンプルの関係性を考えたアプローチが数多く提案されている．

3 章では，顔属性認識を対象とした conditional GAN (cGAN) によるデータ増幅に取り組んだ．深層学習においてデータが不足している場合は，既存データに幾何変化などを施してデータを水増しすることで認識性能を向上させているが，幾何変化を用いたデータ増幅は物体の見え方が変わるものの，画像内の物体は一貫して同じである．そのため，cGAN を用いて狙った画像を生成し，条件として入力したベクトルを教師信号とするデータ増幅を考える．顔画像生成の場合，1 つのサンプルに複数の顔属性がアノテーションされており，それぞれの顔属性は適した入力位置があると考えらえる．また，ネットワーク構造が深くなるにつれて出力層付近で条件が消失するため，入力した条件を満たす画像生成が困難になる．そこで，入力層以外の層にも条件を与え，各層で適切な顔属性を反映するために重み付けして条件を入力する Weighted conditional GAN (WcGAN) を提案する．WcGAN

は従来法より，高品質な顔画像生成を実現しただけでなく，生成画像を増幅データとして使用することで多くの顔属性がベースラインの精度よりも向上した．

4 章では，識別対象の特徴的な領域に着目した cGAN による画像生成に取り組んだ．3 章では，画像全体を高精細に生成して増幅データとして活用したが，一般に CNN は物体の特徴的な領域に強く注視して分類することが知られている．言い換えれば，識別対象を丁寧に生成できれば，十分，増幅データとしての活用が期待できる．そこで，識別時の注視領域の取得が可能な Attention Branch Networks (ABN) と cGAN を組み合わせることで識別対象に限定した画像生成を実現する．提案手法によって生成された画像は定量的画質評価で従来法に劣るにも関わらず，学習データが限られた場合の増幅データとして有効であることが確認できた．

5 章では，データ増幅と組み合わせて Adversarial Examples (AEs) を作成し，バリエーション豊富な AEs を学習する Adversarial Training に取り組んだ．Adversarial Training はシンプルなアプローチで優れた頑健性が得られる反面，ロバスト過適合や通常サンプルに対する分類性能が劣化することが問題視されている．この現象を解消するためには，一般的な学習よりも複雑で膨大なデータを学習する必要があることが理論的に証明されている．そこで，データ増幅と組み合わせて AEs のバリエーションを増幅しながら学習する Masking and Mixing Adversarial Training (M^2AT) を提案する．具体的には，M^2AT は求めた摂動を任意の任意のマスクを用いて矩形内外に摂動が付与されるように切り抜き，mixup と同様にベータ分布からサンプリングした確率によって合成する．教師信号は label smoothing を使用して定義することで，頑健性が劇的に向上しただけでなく，通常サンプルに対する分類精度を維持することを可能とした．また，摂動許容範囲が大きくなったときに M^2AT は従来法よりも高い頑健性を維持することが確認できた．

6 章では，Instance-Reweighted Adversarial Training (IRAT) に生じる問題点について取り組んだ．IRAT は各サンプルの攻撃されやすさをそれらに対する重要度として計算し，重要度を非線形増加関数を用いて重みへ変換した後に各サンプルの損失へ重み付けする AT である．特に，クラス確率に基づく IRAT は優れた頑健性を達成している反面，正解クラス確率と最も迷ったクラス確率から重要度を求めるため暗黙的に 2 クラス分類が想定されている．そこで，まず，暗黙的に 2 クラス分類が想定されている確率に基づく IRAT の弱点を経験的な実験によって明らかにする．具体的には，異なるクラス確率をもつサンプルから限りなく等しい重要度が求められることを示した．この結果から，最も迷ったクラス確率が不正解率を占める割合を用いて，求めた重要度を変換する手法を提案する．提案手法を従来法に組み込むことによって，いくつかの敵対的攻撃に対する頑健性が向上しただけでなく，通常サンプルに対する分類精度も向上させることに成功した．

7 章では，データ増幅の一種である mixup の内挿比をランダムではなく，計算的に求めて学習することに取り組んだ．mixup は任意の内挿比で 2 つのデータを合成して新たなデータを作成する強力なデータ増幅である．mixup の派生手法はデータの合成方法に着目した手法が多く，内挿比は常にランダムな確率である．直感的に，最も損失が高くなる内挿比を用いて合成したデータを正確に分類できるように学習したモデルは，推論データに対して優れた画像分類ができると考えられる．そこで，敵対的学習を参考に，損失が最大となる内挿比を意図的に求め，損失を最小化する Adversarial Interpolating Policy (AIP) を提案する．内挿比は高次元な画像データと異なり，$[0, 1]$ の範囲で定義さ

れるスカラー値であるため，古典的な最小値探索を逆向きに利用して最大の損失となる内挿比を求める．これにより，Adversarial Training を直接用いた場合よりも正確かつ，低コストで内挿比を計算することが可能となる．提案手法の適用箇所は任意の中間層を対象として，入力層が選択された場合は通常の mixup と同様の処理を行う．評価実験によって，AIP があらゆるデータセットとベースモデルで最高性能を達成できることを示した．さらに，データが複雑かつ分類クラス数が多いほど，少ない探索回数でも最高性能が達成できることが実験的に判明した．

8.2 展望

本項では，敵対的方策に焦点を当てて，データ増幅による分類精度向上と Adversarial Training による頑健性向上に取り組んだ．本研究で提案した，新しいデータ増幅である AIP はシンプルな設計でありながらも，非常に優れた分類性能が達成できる．このことからも，提案手法は CNN を用いた画像分類の学習で使用される一般的なデータ増幅の一つとして確立されると考えられる．しかし，実験でも確認されているように，通常の mixup に比べて 2 倍以上の処理時間を必要とする．したがって，更なる応用を見据えて，三分探索法の効率化を行い，提案手法を高速化することが望まれる．また，本稿では mixup や画像分類のみを対象としたが，mixup の派生手法への応用や画像分類以外のタスクのためのデータ増幅法として応用されることも望まれる．

謝　辞

本研究は，著者が中部大学大学院 工学研究科 ロボット理工学専攻 博士後期課程在学中に，同大学 工学部 ロボット理工学科 藤吉弘亘教授および同大学 工学部 情報工学科 山下隆義教授の指導のもとに行ったものです．研究の遂行にあたり，常日頃ご指導を賜りました中部大学 工学部 ロボット理工学科 藤吉弘亘教授および同大学 工学部 情報工学科 山下隆義教授，同大学 AI 数理データサイエンスセンター 平川翼講師に深く感謝の意を表します．ご多忙にも関わらず，副査を快く引き受けていただき，有益なご討論やご助言を賜りました中部大学工学部ロボット理工学科 梅崎太造教授，中部大学工学部ロボット理工学科 平田豊教授，九州大学 大学院 システム情報科学研究院 情報知能工学部門 内田誠一教授に謹んで感謝いたします．本稿の 5 章の研究を進めるにあたって，貴重なご意見，ご指導を頂きましたパナソニック株式会社 石井育規氏，小塚和紀氏，石坂隼氏に心から厚く御礼申し上げます．最後に，本研究に関して熱心に議論していただいた機械知覚&ロボティクスグループの皆様と，日頃から研究グループを支えてくださっている中部大学藤吉研究室秘書 宮腰あゆみ氏に深く感謝致します．

本研究の一部は，JST 次世代研究者挑戦的研究プログラム JPMJSP2158 の支援を受けたものです．

参考文献

[1] T. Karras, S. Laine, M. Aittala, J. Hellsten, J. Lehtinen, and T. Aila, “Analyzing and improving the image quality of stylegan”, IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp.8110-8119, 2020.

[2] I. Goodfellow, J. Shlens, and C. Szegedy, “Explaining and harnessing adversarial examples”, International Conference on Learning Representations, 2015.

[3] G. W. Ding, Y. Sharma, K. Y. C. Lui, and R. Huang, “Mma training: Direct input space margin maximization through adversarial training”, International Conference on Learning Representations, pp.1-28, 2020.

[4] K. Alex, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks”, Advances in Neural Information Processing Systems, pp.1097–1105, 2012.

[5] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition”, IEEE Conference on Computer Vision and Pattern Recognition, pp.770–778, 2016.

[6] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You only look once: Unified, real-time object detection”, IEEE Conference on Computer Vision and Pattern Recognition, pp.779–788, 2016.

[7] K. Sun, B. Xiao, D. Liu, and J. Wang, “Deep high-resolution representation learning for human pose estimation”, IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp.5693-5703, 2019.

[8] C. Godard, O. M. Aodha, M. Firman, and G. Brostow, “Digging into self-supervised monocular depth estimation”, IEEE/CVF International Conference on Computer Vision, pp.3828-3838, 2019.

[9] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, “An image is worth 16x16 words: Transformers for image recognition at scale”, International Conference on Learning Representations, pp.1-21, 2021.

[10] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus, “Intriguing properties of neural networks”, International Conference on Learning Representations, pp.1-10, 2013.

[11] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, “Towards deep learning models resistant to adversarial attacks”, International Conference on Learning Representations, pp.1-23, 2018.

[12] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, “Generative adversarial nets”, Advances in Neural Information Processing Systems, 2014.

[13] M. Mirza, and S. Osindero, “Conditional generative adversarial nets”, arXiv preprint arXiv:1411.1784, 2014.

[14] L. Schmidt, S. Santurkar, D. Tsipras, K. Talwar, and A. Madry, “Adversarially robust generalization requires more data”, Advances in Neural Information Processing Systems, pp.5019-5031, 2018.

[15] S. Lee, H. Lee, and S. Yoon, “Adversarial vertex mixup: Toward better adversarially robust generalization”, IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp.269-278, 2020.

[16] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, “mixup: Beyond empirical risk minimization”, International Conference on Learning Representations, pp.1-13, 2018.

[17] D. P. Kingma, and M. Welling, “Auto-encoding variational bayes”, International Conference on Learning Representations, pp.1–14, 2014.

[18] X. Mao, Q. Li, H. Xie, R. Y. Lau, Z. Wang, and S. P. Smolley, “Least squares generative adversarial networks”, IEEE International Conference on Computer Vision, pp.2813-2821, 2017.

[19] M. Arjovsky, S. Chintala, and L. Bottou, “Wasserstein generative adversarial networks”, International Conference on Machine Learning, pp.214-223, 2017.

[20] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. Courville, “Improved training of wasserstein gans”, Advances in Neural Information Processing Systems, pp.5769–5779, 2017.

[21] T. Miyato, T. Kataoka, M. Koyama, and Y. Yoshida, “Spectral normalization for generative adversarial networks”, International Conference on Learning Representations, pp.1-26, 2018.

[22] S. Ioffe, and C. Szegedy, “Batch normalization: Accelerating deep network training by reducing internal covariance shift”, International Conference on Machine Learning, pp.448-456, 2015.

[23] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, “Improved techniques for training gans”, Advances in Neural Information Processing Systems, pp.2234–2242, 2016.

[24] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, “Gans trained by a tow time-scale update rule converge to a local nash equilibrium”, Advances in Neural Information Processing Systems, pp.6629–6640, 2017.

[25] E. Denton, S. Chintala, A. Szlam, and R. Fergus, “Deep generative image models using a laplacian pyramid of adversarial networks”, Advances in Neural Information Processing Systems, pp.1486–1494, 2015.

[26] A. Radford, L. Metz, and S. Chintala, “Unsupervised representation learning with deep convolutional generative adversarial networks”, International Conference on Learning Representations, pp.1-16, 2016.

[27] J. Donahue, P. Krähenbühl, and T. Darrell, “Adversarial feature learning”, International Conference on Learning Representations, pp.1-18, 2017.

[28] T. Karras, T. Aila, S. Laine, and J. Lehtinen, “Progressive growing of gans for improved quality, stability, and variation”, International Conference on Learning Representations, pp.1-26, 2018.

[29] H. Zhang, I. Goodfellow, D. Metaxas, and A. Odena, “Self-attention generative adversarial networks”, International Conference on Machine Learning, pp.7354-7363, 2019.

[30] Z. Ding, X. Y. Liu, M. Yin, and L. Kong, “Tgan: Deep tensor generative adversarial nets for large image generation”, arXiv preprint arXiv:1901.09953, 2019.

[31] T. Karras, S. Laine, and T. Aila, “A style-based generator architecture for generative adversarial networks”, IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp.4401-4410, 2019.

[32] A. Brock, J. Donahue, and K. Simonyan, “Large scale gan training for high fidelity natural image synthesis”, International Conference on Learning Representations, pp.1-35, 2019.

[33] J. Donahue, and K. Simonyan, “Large scale adversarial representation learning”, Advances in Neural Information Processing Systems, pp.10542–10552, 2019.

[34] C. H. Lin, C. C. Chang, Y. S. Chen, D. C. Juan, W. Wei, and H. T. Chen, “Coco-gan: Generation by parts via conditional coordinating”, IEEE/CVF International Conference on Computer Vision, pp.4511-4520, 2019.

[35] A. Karnewar, and O. Wang, “Msg-gan: Multi-scale gradients for generative adversarial networks”, IEEE Conference on Computer Vision and Pattern Recognition, pp.7799-7808, 2020.

[36] E. Schonfeld, B. Schiele, and A. Khoreva, “A u-net based discriminator for generative adversarial networks”, IEEE Conference on Computer Vision and Pattern Recognition, pp.8207-8216, 2020.

[37] T. Karras, M. Aittala, S. Laine, E. Härkönen, J. Hellsten, J. Lehtinen, and T. Aila, “Alias-free generative adversarial networks”, Advances in Neural Information Processing Systems, 2021.

[38] D. P. Kingma, and J. Ba, “Adam: A method for stochastic optimization”, International Conference on Learning Representations, pp.1-15, 2015.

[39] R. Child, S. Gray, A. Radford, and I. Sutskever, “Generating long sequences with sparse transformers”, arXiv preprint arXiv:1904.10509, 2019.

[40] G. Daras, A. Odena, H. Zhang, and A. G. Dimakis, “Your local gan: Designing two dimensional local attention mechanisms for generative models”, IEEE Conference on Computer Vision and Pattern Recognition, pp.14519-14527, 2020.

[41] X. Huang, and S. Belongie, “Arbitrary style transfer in real-time with adaptive instance normalization”, IEEE International Conference on Computer Vision, pp.1501-1510, 2017.

[42] S. Reed, Z. Akata, X. Yan, L. Logeswaran, B. Schiele, and Honglak, “Generative adversarial text to image synthesis”, International Conference on Machine Learning, pp.1060-1069, 2016.

[43] H. Zhang, T. Xu, H. Li, S. Zhang, X. Wang, X. Huang, and D. Metaxas, “Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks”, IEEE International Conference on Computer Vision, pp.5908-5916, 2017.

[44] A. Odena, C. Olah, and J. Shlens, “Conditional image synthesis with auxiliary classifier gans”, International Conference on Machine Learning, pp.2642-2651, 2017.

[45] T. Kaneko, K. Hiramatsu, and K. Kashino, “Generative attribute controller with conditional filtered generative adversarial networks”, IEEE Conference on Computer Vision and Pattern Recognition, pp.7006-7015, 2017.

[46] T. Miyato, and M. Koyama, “cgans with projection discriminator”, International Conference on Learning Representations, pp.1-23, 2018.

[47] A. Sage, E. Agustsson, R. Timofte, and L. V. Gool, “Logo synthesis and manipulation with clustered generative adversarial networks”, IEEE Conference on Computer Vision and Pattern Recognition, pp.5879-5888, 2018.

[48] T. Chen, M. Lucic, N. Houlsby, and S. Gelly, “On self modulation for generative adversarial networks”, International Conference on Learning Representations, pp.1-18, 2018.

[49] T. Xu, P. Zhang, Q. Huang, H. Zhang, Z. Gan, X. Huang, and X. He, “Attngan: Fine-grained text to image generation with attentional generative adversarial networks”, IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp.1316-1324, 2018.

[50] H. Zhang, T. Xu, H. Li, S. Zhang, X. Wang, X. Huang, and D. Metaxas, "Stackgan++: Realistic image synthesis with stacked generative adversarial networks", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol.41, no.08, pp.1947-1962, 2019.

[51] Q. Mao, H. Y. Lee, H. Y. Tseng, S. Ma, and M. H. Yang, "Mode seeking generative adversarial networks for diverse image synthesis", IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp.1429-1437, 2019.

[52] N. Pandey, and A. Savakis, "Poly-gan: Multi-conditioned gan for fashion synthesis", Neurocomputing, vol.414, pp.356-364, 2020.

[53] B. Wang, H. Zheng, X. Liang, Y. Chen, L. Lin, and M. Yang, "Toward characteristic-preserving image-based virtual try-on network", European Conference on Computer Vision, pp.607-623, 2018.

[54] J. Wu, Z. Huang, D. Acharya, W. Li, J. Thoma, D. P. Paudel, and L. V. Gool, "Sliced wasserstein generative models", IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp.3713-3722, 2019.

[55] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "Imagenet large scale visual recognition challenge", International Journal of Computer Vision, vol.115, no.3, pp.211-252, 2015.

[56] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision", IEEE Conference on Computer Vision and Pattern Recognition, pp.2818-2826, 2016.

[57] H. Hosseini, and R. Poovendran, "Semantic adversarial examples", IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp.1614–1619, 2018.

[58] A. Joshi, A. Mukherjee, S. Sarkar, and C. Hegde, "Semantic adversarial attacks: Parametric transformations that fool deep classifiers", IEEE/CVF International Conference on Computer Vision, pp.4772-4782, 2019.

[59] A. Bhattad, M. J. Chong, K. Liang, B. Li, and D. A. Forsyth, "Unrestricted adversarial examples via semantic manipulation", International Conference on Learning Representations, pp.1–19, 2020.

[60] Y. Bakhti, S. A. Fezza, W. Hamidouche, and O. Déforges, "Ddsa: A defense against adversarial attacks using deep denoising sparse autoencoder", IEEE Access, vol.7, pp.160397-160407, 2019.

[61] P. Samangouei, M. Kabkab, and R. Chellappa, "Defense-gan: Protecting classifiers against adversarial attacks using generative models", International Conference on Learning Representations, pp.1-17, 2018.

[62] Y. Song, T. Kim, S. Nowozin, S. Ermon, and N. Kushman, “Pixeldefend: Leveraging generative models to understand and defend against adversarial examples”, International Conference on Learning Representations, pp.1-20, 2018.

[63] W. Xu, D. Evans, and Y. Qi, “Feature squeezing: Detecting adversarial examples in deep neural networks”, Network and Distributed System Security Symposium, 2018.

[64] P. Sperl, C. yu Kao, P. Chen, and K. Böttinger, “Dla: Dense-layer-analysis for adversarial example detection”, IEEE European Symposium on Security and Privacy, pp.198-215, 2020.

[65] C. Kou, H. K. Lee, E. C. Chang, and T. K. Ng, “Enhancing transformation-based defenses against adversarial attacks with a distribution classifier”, International Conference on Learning Representations, pp.1-19, 2020.

[66] N. Papernot, P. McDaniel, X. Wu, S. Jha, and A. Swami, “Distillation as a defense to adversarial perturbations against deep neural networks”, IEEE Symposium on Security and Privacy, pp.582-597, 2016.

[67] A. Kurakin, I. Goodfellow, and S. Bengio, “Adversarial examples in the physical world”, International Conference on Learning Representations Workshop, pp.1-14, 2017.

[68] N. Carlini, and D. Wagner, “Towards evaluating the robustness of neural networks”, IEEE Symposium on Security and Privacy, pp.39-57, 2017.

[69] Y. Dong, F. Liao, T. Pang, H. Su, J. Zhu, X. Hu, and J. Li, “Boosting adversarial attacks with momentum”, IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp.9185-9193, 2018.

[70] C. Xie, Z. Zhang, Y. Zhou, S. Bai, J. Wang, Z. Ren, and A. Yuille, “Improving transferability of adversarial examples with input diversity”, IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp.2730-2739, 2019.

[71] J. Su, D. V. Vargas, and K. Sakurai, “One pixel attack for fooling deep neural networks”, IEEE Transactions on Evolutionary Computation, vol.23, no.5, pp.828-841, 2019.

[72] G. Sriramanan, S. Addepalli, A. Baburaj, and R. V. Babu, “Guided adversarial attack for evaluating and enhancing adversarial defenses”, Advances in Neural Information Processing Systems, vol.33, pp.20297-20308, 2020.

[73] Q. Z. Cai, C. Liu, and D. Song, “Curriculum adversarial training”, International Joint Conference on Artificial Intelligence, pp.3740-3747, 2018.

[74] Y. Bengio, J. Louradour, R. Collobert, and J. Weston, “Curriculum learning”, International Conference on Machine Learning, pp.41-48, 2009.

[75] Y. Wang, X. Ma, J. Bailey, J. Yi, B. Zhou, and Q. Gu, “On the convergence and robustness of adversarial training”, International Conference on Machine Learning, pp.6586-6595, 2019.

[76] J. Zhang, X. Xu, B. Han, G. Niu, L. Cui, M. Sugiyama, and M. Kankanhalli, “Attacks which do not kill training make adversarial learning stronger”, International Conference on Machine Learning, vol.119, pp.11278–11287, 2020.

[77] A. Lamb, V. Verma, J. Kannala, and Y. Bengio, “Interpolated adversarial training: Achieving robust neural networks without sacrificing too much accuracy”, ACM Workshop on Artificial Intelligence and Security, pp.95-103, 2019.

[78] A. Laugros, A. Caplier, and M. Ospici, “Addressing neural network robustness with mixup and targeted labeling adversarial training”, European Conference on Computer Vision Workshops, pp.178-195, 2020.

[79] C. Chen, J. Zhang, X. Xu, T. Hu, G. Niu, G. Chen, and M. Sugiyama, “Guided interpolation for adversarial training”, arXiv preprint arXiv: 2102.07327, 2021.

[80] C. Song, Y. Fan, Y. Yang, B. Wu, Y. Li, Z. Li, and K. He, “Regional adversarial training for better robust generalization”, arXiv preprint arXiv:2109.00678, 2021.

[81] H. Zhang, and J. Wang, “Defense against adversarial attacks using feature scattering-based adversarial training”, Advances in Neural Information Processing Systems, vol.32, pp.1829-1839, 2019.

[82] H. Kannan, A. Kurakin, and I. Goodfellow, “Adversarial logit pairing”, arXiv preprint arXiv:1803.06373, 2018.

[83] H. Zhang, Y. Yu, J. Jiao, E. Xing, L. E. Ghaoui, and M. Jordan, “Theoretically principled trade-off between robustness and accuracy”, International Conference on Machine Learning, vol.97, pp.7472–7482, 2019.

[84] Y. Wang, D. Zou, J. Yi, J. Bailey, X. Ma, and Q. Gu, “Improving adversarial robustness requires revisiting misclassified examples”, International Conference on Learning Representations, pp.1-14, 2020.

[85] C. Mao, Z. Zhong, J. Yang, C. Vondrick, and B. Ray, “Metric learning for adversarial robustness”, Advances in Neural Information Processing Systems, pp.478-489, 2019.

[86] H. Elad, and F. Aasa, “Deep metric learning using triplet network”, Similarity-Based Pattern Recognition, pp.84–92, 2015.

[87] Y. Duan, W. Zheng, X. Lin, J. Lu, and J. Zhou, “Deep adversarial metric learning”, IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp.2780-2789, 2018.

[88] H. Wang, Y. Deng, S. Yoo, H. Ling, and Y. Lin, “Agkd-bml: Defense against adversarial attack by attention guided knowledge distillation and bi-directional metric learning”, IEEE/CVF International Conference on Computer Vision, pp.7658-7667, 2021.

[89] J. Cui, S. Liu, L. Wang, and J. Jia, “Learnable boundary guided adversarial training”, IEEE/CVF International Conference on Computer Vision, pp.15721–15730, 2021.

[90] Y. Balaji, T. Goldstein, and J. Hoffman, “Instance adaptive adversarial training: Improved accuracy tradeoffs in neural nets”, arXiv preprint arXiv: 1910.08051, 2019.

[91] M. Cheng, Q. Lei, P. Y. Chen, I. Dhillon, and C. J. Hsieh, “Cat: Customized adversarial training for improved robustness”, International Joint Conference on Artificial Intelligence, 2022.

[92] S. A. Taghanaki, K. Abhishek, S. Azizi, and G. Hamarneh, “A kernelized manifold mapping to diminish the effect of adversarial perturbations”, IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp.11340-11349, 2019.

[93] Y. Wen, K. Zhang, Z. Li, and Y. Qiao, “A discriminative feature learning approach for deep face recognition”, European Conference on Computer Vision, pp.499–515, 2016.

[94] A. Mustafa, S. Khan, M. Hayat, R. Goecke, J. Shen, and L. Shao, “Adversarial defense by restricting the hidden space of deep neural networks”, IEEE/CVF International Conference on Computer Vision, pp.3384-3393, 2019.

[95] X. Li, X. Li, D. Pan, and D. Zhu, “Improving adversarial robustness via probabilistically compact loss with logit constraints”, AAAI Conference on Artificial Intelligence, vol.35, no.10, pp.8482-8490, 2021.

[96] H. Y. Chen, P. H. Wang, C. H. Liu, S. C. Chang, J. Y. Pan, Y. T. Chen, W. Wei, and D. C. Juan, “Complement objective training”, International Conference on Learning Representations, pp.1-11, 2019.

[97] H. Y. Chen, J. H. Liang, S. C. Chang, J. Y. Pan, Y. T. Chen, W. Wei, and D. C. Juan, “Improving adversarial robustness via guided complement entropy”, IEEE/CVF International Conference on Computer Vision, pp.4880-4888, 2019.

[98] D. Wu, Shu-Tao, and Y. Wang, “Adversarial weight perturbation helps robust generalization”, Advances in Neural Information Processing Systems, pp.2958–2969, 2020.

[99] C. Yu, B. Han, M. Gong, L. Shen, S. Ge, D. Bo, and T. Liu, "Robust weight perturbation for adversarial training", International Joint Conference on Artificial Intelligence, pp.3688-3694, 2022.

[100] C. Yu, B. Han, L. Shen, J. Yu, C. Gong, M. Gong, and T. Liu, "Understanding robust overfitting of adversarial training and beyond", International Conference on Machine Learning, pp.25595-25610, 2022.

[101] H. Zeng, C. Zhu, T. Goldstein, and F. Huang, "Are adversarial examples created equal? a learnable weighted minimax risk for robustness under non-uniform attacks", AAAI Conference on Artificial Intelligence, vol.35, no.12, pp.10815–10823, 2021.

[102] J. Zhang, J. Zhu, G. Niu, B. Han, M. Sugiyama, and M. Kankanhalli, "Geometry-aware instance-reweighted adversarial training", International Conference on Learning Representations, pp.1–29, 2021.

[103] Q. Wang, F. Liu, B. Han, T. Liu, C. Gong, G. Niu, M. Zhou, and M. Sugiyama, "Probabilistic margins for instance reweighting in adversarial training", Advances in Neural Information Processing Systems, pp.1–12, 2021.

[104] R. Gao, F. Liu, K. Zhou, G. Niu, B. Han, and J. Cheng, "Local reweighting for adversarial training", arXiv preprint arXiv:2106.15776, 2021.

[105] M. Kim, J. Tack, J. Shin, and S. J. Hwang, "Entropy weighted adversarial training", International Conference on Machine Learning Workshop, 2021.

[106] C. Holtz, T. W. Weng, and G. Mishne, "Learning sample reweighting for adversarial robustness", OpenReview, 2021.

[107] C. Finn, P. Abbeel, and S. Levine, "Model-agnostic meta-learning for fast adaptation of deep networks", International Conference on Machine Learning, vol.70, pp.1126-1135, 2017.

[108] J. Uesato, J. B. Alayrac, P. S. Huang, R. Stanforth, A. Fawzi, and P. Kohli, "Are labels required for improving adversarial robustness?", Advances in Neural Information Processing Systems, pp.12214–12223, 2019.

[109] Y. Carmon, A. Raghunathan, L. Schmidt, P. Liang, and J. C. Duchi, "Unlabeled data improves adversarial robustness", Advances in Neural Information Processing Systems, pp.11192–11203, 2019.

[110] A. Shafahi, M. Najibi, A. Ghiasi, Z. Xu, J. Dickerson, C. Studer, L. S. Davis, G. Taylor, and T. Goldstein, "Adversarial training for free!", Advances in Neural Information Processing Systems, vol.32, pp.3358-3369, 2019.

[111] E. Wong, L. Rice, and J. Z. Kolter, “Fast is better than free: Revisiting adversarial training”, International Conference on Learning Representations, pp.1-17, 2020.

[112] C. Qin, J. Martens, S. Gowal, D. Krishnan, K. Dvijotham, A. Fawzi, S. De, R. Stanforth, and P. Kohli, “Adversarial robustness through local linearization”, Advances in Neural Information Processing Systems, pp.13842–13853, 2019.

[113] H. Kim, W. Lee, and J. Lee, “Understanding catastrophic overfitting in single-step adversarial training”, AAAI Conference on Artificial Intelligence, vol.35, no.9, pp.8119-8127, 2021.

[114] Y. Fu, Q. Yu, Y. Zhang, S. Wu, X. Ouyang, D. D. Cox, and Y. Lin, “Drawing robust scratch tickets: Subnetworks with inborn robustness are found within randomly initialized networks”, Advances in Neural Information Processing Systems, pp.13059-13072, 2021.

[115] T. Li, Y. Wu, S. Chen, K. Fang, and X. Huang, “Subspace adversarial training”, IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp.13409-13418, 2022.

[116] J. Frankle, and M. Carbin, “The lottery ticket hypothesis: Finding sparse, trainable neural networks”, International Conference on Learning Representations, pp.1-42, 2019.

[117] D. P. Kingma, and P. Dhariwal, “Glow: Generative flow with invertible 1x1 convolutions”, Advances in Neural Information Processing Systems, pp.10215-10224, 2018.

[118] A. van den Oord, N. Kalchbrenner, O. Vinyals, L. Espeholt, A. Graves, and K. Kavukcuoglu, “Conditional image generation with pixelcnn decoders”, Advances in Neural Information Processing Systems, pp.4797–4805, 2016.

[119] L. Wan, J. Wan, Y. Jin, Z. Tan, and S. Z. Li, “Fine-grained multi-attribute adversarial learning for face generation of age, gender and ethnicity”, International Conference on Biometrics, pp.98–103, 2018.

[120] N. Bodla, G. Hua, and R. Chellappa, “Semi-supervised fusedgan for conditional image generation”, The European Conference on Computer Vision, pp.669–683, 2018.

[121] Z. Yuan, J. Zhang, S. Shan, and X. Chen, “Attributes aware face generation with generative adversarial networks”, International Conference on Pattern Recognition, pp.1657–1664, 2020.

[122] Z. Liu, P. Luo, X. Wang, and X. Tang, “Deep learning face attributes in the wild”, IEEE International Conference on Computer Vision, pp.3730–3738, 2015.

[123] Z. Zhong, L. Zheng, G. Kang, S. Li, and Y. Yang, “Random erasing data augmentation”, AAAI Conference on Artificial Intelligence, vol.34, no.07, pp.13001–13008, 2020.

[124] J. Y. Zhu, T. Park, P. Isola, and A. A. Efros, “Unpaired image-to-image translation using cycle-consistent adversarial networks”, IEEE International Conference on Computer Vision, pp.2223–2232, 2017.

[125] T. R. Shaham, T. Dekel, and T. Michaeli, “Singan: Learning a generative model from a single natural image”, IEEE/CVF International Conference on Computer Vision, pp.4570–4580, 2019.

[126] Y. Choi, Y. Uh, J. Yoo, and J. W. Ha, “Stargan v2: Diverse image synthesis for multiple domains”, IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp.8188–8197, 2020.

[127] C. Ledig, L. Theis, F. Huszar, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, and W. Shi, “Photo-realistic single image super-resolution using a generative adversarial network”, IEEE Conference on Computer Vision and Pattern Recognition, pp.4681–4690, 2017.

[128] A. Antoniou, A. Storkey, and H. Edwards, “Data augmentation generative adversarial networks”, arXiv preprint arXiv: 1711.04340, 2017.

[129] C. Han, L. Rundo, R. A. Y. Nagano, Y. Furukawa, G. Mauri, H. Nakayama, and H. Hayashi, “Combining noise-to-image and image-to-image gans: Brain mr image augmentation for tumor detection”, IEEE Access, vol.7, pp.156966–156977, 2019.

[130] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, “Learning deep features for discriminative localization”, IEEE Conference on Computer Vision and Pattern Recognition, pp.2921–2929, 2016.

[131] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, “Grad-cam: Visual explanations from deep networks via gradient-based localization”, IEEE International Conference on Computer Vision, pp.618–626, 2017.

[132] H. Fukui, T. Hirakawa, T. Yamashita, and H. Fujiyoshi, “Attention branch network: Learning of attention mechanism for visual explanation”, IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp.10705–10714, 2019.

[133] S. Yun, D. Han, S. J. Oh, S. Chun, J. Choe, and Y. Yoo, “Cutmix: Regularization strategy to train strong classifiers with localizable features”, the IEEE/CVF International Conference on Computer Vision, pp.6023–6032, 2019.

[134] J. Qin, J. Fang, Q. Zhang, W. Liu, X. Wang, and Xinggang, “Resizemix: Mixing data with preserved object information and true labels”, arXiv preprint arXiv: 2012.11101, 2020.

[135] X. Huang, M. Y. Liu, S. Belongie, and J. Kautz, “Multimodal unsupervised image-to-image translation”, European Conference on Computer Vision, ed. 172-189, 2018.

[136] A. Shrivastava, T. Pfister, O. Tuzel, J. Susskind, W. Wang, and R. Webb, “Learning from simulated and unsupervised images through adversarial training”, IEEE Conference on Computer Vision and Pattern Recognition, pp.2107–2116, 2017.

[137] M. Lin, Q. Chen, and S. Yan, “Network in network”, International Conference on Learning Representations, pp.1–10, 2014.

[138] Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, and A. Y. Ng, “Reading digits in natural images with unsupervised feature learning”, NIPS Workshop on Deep Learning and Unsupervised Feature Learning, 2011.

[139] K. Shmelkov, C. Schmid, and K. Alahari, “How good is my gan?”, European Conference on Computer Vision, pp.218–234, 2018.

[140] S. Zhao, Z. Liu, J. Lin, J. Y. Zhu, and S. Han, “Differentiable augmentation for data-efficient gan training”, Advances in Neural Information Processing Systems, pp.7559–7570, 2020.

[141] K. Yamanaka, R. Matsumoto, K. Takahashi, and T. Fujii, “Adversarial patch attacks on monocular depth estimation networks”, IEEE Access, vol.8, pp.179094–179104, 2020.

[142] C. Xie, J. Wang, Z. Zhang, Y. Zhou, L. Xie, and A. Yuille, “Adversarial examples for semantic segmentation and object detection”, IEEE International Conference on Computer Vision, pp.1369–1378, 2017.

[143] D. Meng, and H. Chen, “Magnet: a two-pronged defense against adversarial examples”, ACM Conference on Computer and Communications Security, pp.135—147, 2017.

[144] D. Yin, R. G. Lopes, J. Shlens, E. D. Cubuk, and J. Gilmer, “A fourier perspective on model robustness in computer vision”, Advances in Neural Information Processing Systems, no.1189, pp.13276–13286, 2019.

[145] D. Tsipras, S. Santurkar, L. Engstrom, A. Turner, and A. Madry, “Robustness may be at odds with accuracy”, International Conference on Learning Representations, pp.1-23, 2019.

[146] H. Guo, Y. Mao, and R. Zhang, “Mixup as locally linear out-of-manifold regularization”, AAAI Conference on Artificial Intelligence, vol.33, no.01, pp.3714–3722, 2019.

[147] C. Fu, H. Chen, N. Ruan, and W. Jia, “Label smoothing and adversarial robustness”, arXiv preprint arXiv:2009.08233, 2020.

[148] S. Zagoruyko, and N. Komodakis, “Wide residual networks”, British Machine Vision Conference, pp.87.1-87.12, 2016.

[149] L. Rice, E. Wong, and Z. Kolter, “Overfitting in adversarially robust deep learning”, International Conference on Machine Learning, vol.119, pp.8093-8104, 2020.

[150] A. Krizhevsky, and G. Hinton, “Learning multiple layers of features from tiny images”, Technical report, University of Toronto, 2009.

[151] T. Pang, X. Yang, Y. Dong, H. Su, and J. Zhu, “Bag of tricks for adversarial training”, International Conference on Learning Representations, pp.1-21, 2021.

[152] F. Croce, and M. Hein, “Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks”, International Conference on Machine Learning, vol.119, pp.2206-2216, 2020.

[153] P. Y. Simard, Y. A. L. Cun, J. S. Denker, and B. Victorri, Transformation invariance in pattern recognition - tangent distance and tangent propagation, , pp.235–269Springer Berlin Heidelberg, 2012.

[154] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based learning applied to document recognition”, Proceedings of the IEEE, vol.86, no.11, pp.2278-2324, 1998.

[155] K. Simonyan, and A. Zisserman, “Very deep convolutional networks for large-scale image recognition”, International Conference on Learning Representations, 2015.

[156] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, “Dropout: A simple way to prevent neural networks from overfitting”, Journal of Machine Learning Research, vol.15, no.56, pp.1929–1958, 2014.

[157] S. Hanson, and L. Pratt, “Comparing biases for minimal network construction with back-propagation”, Advances in Neural Information Processing Systems, pp.177–185, 1988.

[158] I. Loshchilov, and F. Hutter, “Decoupled weight decay regularization”, International Conference on Learning Representations, pp.1-18, 2019.

[159] C. Xie, M. Tan, B. Gong, J. Wang, A. L. Yuille, and Q. V. Le, “Adversarial examples improve image recognition”, IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp.819-828, 2020.

[160] T. DeVries, and G. W. Taylor, “Improved regularization of convolutional neural networks with cutout”, arXiv preprint arXiv: 1708.04552, 2017.

[161] Y. Tokozume, Y. Ushiku, and T. Harada, “Between-class learning for image classification”, IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp.5486-5494, 2018.

[162] V. Verma, A. Lamb, C. Beckham, A. Najafi, I. Mitliagkas, D. Lopez-Paz, and Y. Bengio, "Manifold mixup: Better representations by interpolating hidden states", International Conference on Machine Learning, pp.6438-6447, 2019.

[163] E. D. Cubuk, B. Zoph, D. Mane, V. Vasudevan, and Q. V. Le, "Autoaugment: Learning augmentation strategies from data", IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp.113-123, 2019.

[164] E. D. Cubuk, B. Zoph, J. Shlens, and Q. V. Le, "Randaugment: Practical automated data augmentation with a reduced search space", IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, pp.702-703, 2020.

[165] J. yong Sohn, L. Shang, H. Chen, J. Moon, D. S. Papailiopoulos, and K. Lee, "Genlabel: Mixup relabeling using generative models", International Conference on Machine Learning, pp.20278-20313, 2022.

[166] R. Baena, L. Drumetz, and V. Gripon, "Preventing manifold intrusion with locality: Local mixup", arXiv preprint arXiv: 2201.04368, 2022.

[167] Z. Mai, G. Hu, D. Chen, F. Shen, and H. T. Shen, "Metamixup: Learning adaptive interpolation policy of mixup with metalearning", IEEE Transactions on Neural Networks and Learning Systems, vol.33, no.7, pp.3050-3064, July 2022.

[168] J. H. Lee, M. Z. Zaheer, M. Astrid, and S. I. Lee, "Smoothmix: a simple yet effective data augmentation to train robust classifiers", IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, pp.3264-3274, 2020.

[169] J. H. Kim, W. Choo, H. Jeong, and H. O. Song, "Co-mixup: Saliency guided joint mixup with supermodular diversity", International Conference on Learning Representations, pp.1-21, 2021.

[170] J. H. Kim, W. Choo, and H. O. Song, "Puzzle mix: Exploiting saliency and local statistics for optimal mixup", International Conference on Machine Learning, pp.5275-5285, 2020.

[171] H. P. Chou, S. C. Chang, J. Y. Pan, W. Wei, and D. C. Juan, "Remix: Rebalanced mixup", European Conference on Computer Vision Workshops, pp.95-110, 2020.

[172] D. Walawalkar, Z. Liu, M. Savvides, and Z. Shen, "Attentive cutmix: An enhanced data augmentation approach for deep learning based image classification", IEEE International Conference on Acoustics, Speech and Signal Processing, pp.3642-3646, 2020.

[173] C. Gong, D. Wang, M. Li, V. Chandra, and Q. Liu, "Keepaugment: A simple information-preserving data augmentation approach", IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp.1055-1064, 2021.

[174] S. Huang, X. Wang, and D. Tao, "Snapmix: Semantically proportional mixing for augmenting fine-grained data", AAAI Conference on Artificial Intelligence, pp.1628-1636, 2021.

[175] D. Hendrycks, N. Mu, E. D. Cubuk, B. Zoph, J. Gilmer, and B. Lakshminarayanan, "Augmix: A simple data processing method to improve robustness and uncertainty", International Conference on Learning Representations, pp.1-15, 2020.

[176] G. Liu, Y. Mao, H. Huang, W. Gao, and X. Li, "Adversarial mixing policy for relaxing locally linear constraints in mixup", Conference on Empirical Methods in Natural Laguage Processing, pp.2998-3008, 2021.

[177] K. He, X. Zhang, S. Ren, and J. Sun, "Identity mappings in deep residual networks", European Conference on Computer Vision, pp.630–645, 2016.

[178] H. Li, Z. Xu, G. Taylor, C. Studer, and T. Goldstein, "Visualizing the loss landscape of neural nets", Advances in Neural Information Processing Systems, pp.6391-6401, 2018.

研究業績一覧

学術論文

[1] 足立 浩規, 平川 翼, 山下 隆義, 藤吉 弘亘, “重み付き条件を用いた Generative Adversarial Networks による有効な顔画像生成”, 電子情報通信学会論文誌, vol. J105-D, no. 04, pp. 271–282, 2022.

[2] 足立 浩規, 平川 翼, 山下 隆義, 藤吉 弘亘, “注視領域を考慮した GAN による識別に効果的なデータ増幅”, 電子情報通信学会論文誌, vol. J105-D, no. 07, pp. 470–479, 2022.

国際会議発表論文 (査読あり)

[1] Hiroki Adachi, Tsubasa Hirakawa, Takayoshi Yamashita, Hironobu Fujiyoshi, Yasunori Ishii, Kazuki Kozuka, “Masking and Mixing Adversarial Training”, 18th International Conference on Computer Vision Theory and Applications, 2023.

[2] Takaaki Iwayoshi, Hiroki Adachi, Tsubasa Hirakawa, Takayoshi Yamashita, Hironobu Fujiyoshi, “Complement Objective Mining Branch for Optimizing Attention Map”, 18th International Conference on Computer Vision Theory and Applications, 2023.

[3] Hiroki Adachi, Hiroshi Fukui, Takayoshi Yamashita, Hironobu Fujiyoshi, “Facial Image Generation by Generative Adversarial Networks using Weighted Conditions”, 14th International Conference on Computer Vision Theory and Applications, pp. 139–145, 2019.

学会口頭発表 (査読あり)

[1] 土松千紗, 足立浩規, 平川翼, 山下隆義, 藤吉弘亘, “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”, 画像の認識・理解シンポジウム, 2022.

[2] 足立 浩規, 平川 翼, 山下 隆義, 藤吉 弘亘, “注視領域を考慮した GAN による識別に効果的なデータ増幅”, 画像の認識・理解シンポジウム, 2020.

[3] 高田 雅之, 足立 浩規, 平川 翼, 山下 隆義, 藤吉 弘亘, “Attention Pairwise Ranking によるスキル優劣判定における視覚的説明と高精度化”, 画像の認識・理解シンポジウム, 2020.

[4] 今枝 航, 足立 浩規, 平川 翼, 山下 隆義, 藤吉 弘亘, “Attention 機構を導入した CycleGAN による識別に有効なスタイル変換”, 画像の認識・理解シンポジウム, 2019.

[5] 足立浩規，福井宏，山下隆義，藤吉弘亘, “重みを導入した Conditional Generative Adversarial Network による顔画像生成の高品質化”, 画像の認識・理解シンポジウム, 2018.

学会口頭発表 (査読なし)

[1] 足立 浩規, 平川 翼, 山下 隆義, 藤吉 弘亘, “[サーベイ論文] Adversarial Training”, パターン認識・メディア理解研究会, 2022.

[2] 足立 浩規, 平川 翼, 山下 隆義, 藤吉 弘亘, 石井 育規, 石坂 隼, 小塚 和紀, “精度と頑健性のトレードオフ緩和を目的とした敵対学習”, 画像の認識・理解シンポジウム, 2021.

[3] 佐々木一磨, 足立浩規, Yifei Huang, 石川裕地, 菊池康太郎, 藤森和希, Li Zhenqiang, “【招待ショートサーベイ】実ロボットの知識獲得のためのシミュレーションを用いた転移学習における環境差異の解決”, パターン認識・メディア理解研究会, 2018.

[4] 足立浩規，福井宏，山下隆義，藤吉弘亘, “重みを導入した Conditional Generative Adversarial Network による段階的な顔画像生成”, 日本ロボット学会学術講演会, 2018.

学術表彰

[1] 2020 年　MIRU 学生奨励賞
題目: 注視領域を考慮した GAN による識別に効果的なデータ増幅

[2] 2021 年　2021 年度 PRMU 研究奨励賞
題目: [サーベイ論文] Adversarial Training

著書

[1] 足立浩規 (1 章担当): “コンピュータビジョン最前線 Winter 2022”, 共立出版, ISBN:9784320125469, 2022.