

チェス盤歪みを考慮した固定畳み込み層を含むCNNとその効果 CNN with Fixed Convolutional Layers Considering Checkerboard Artifacts and Its Effectiveness

木下裕磨[†] 貴家仁志[†]
[†] 首都大学東京大学院システムデザイン研究科

Yuma KINOSHITA[†] Hitoshi KIYA[†]
[†] Department of Computer Science, Tokyo Metropolitan University

アブストラクト 本稿では、畳み込みニューラルネットワーク (Convolutional neural network, CNN) におけるチェス盤歪みの発生を回避するため、新しい固定畳み込み層とそれらを含むCNNを提案する。さらに、敵対的生成ネットワーク (Generative adversarial network, GAN) に対して固定畳み込み層を適用し、GANにおけるチェス盤歪みとその回避のためのフィルタカーネルの滑らかさがGANの性能に与える影響をはじめて調査する。4種のCNNを用いた画像分類実験の結果、従来のCNNは、チェス盤歪みによりその性能が制限されていたことが確認される。さらに、これらCNNに提案する固定畳み込み層を適用することで、画像分類の精度向上が可能であると示される。GANを用いた画像生成実験でも同様に、提案する固定畳み込み層の利用によるチェス盤歪み回避が、生成される画像の品質を高めることが示される。加えて、滑らかな固定畳み込み層のGANへの適用が、生成画像のさらなる品質向上に寄与することが明らかとなる。

1 はじめに

深層ニューラルネットワーク (Deep neural network, DNN) の一種である畳み込みニューラルネットワーク (Convolutional neural network, CNN) [1,2] は、その卓越した性能から、画像処理、自然言語処理、音声・音響処理などの分野で広く利用されている。これまでに、VGG [3]、ResNet [4]、U-Net [5] など多くのCNNアーキテクチャが提案されている。これらのCNNアーキテクチャでは、アップサンプリング層およびダウンサンプリング層を用いた特徴マップのサンプリングレート変換処理が一般に用いられるアップサンプリング層としては転置畳み込み層が、ダウンサンプリング層としてはストライドありの畳み込み層がそれぞれ代表的である [6]。しかしながら、超解像などの画像変換処理に転置畳み込み層を利用することは、出力画像に対してチェス盤歪みを引き起こすことが広く知られている [7]。加えて、ストライドありの畳み込み層を含むCNNの学習では、誤差逆伝搬時に転置畳み込み処理が実行されるため、計算される勾配にチェス盤歪みが発生する [7]。したがって、ResNetなどの転置畳み込み層を含まないCNNでさえ、チェス盤歪みの影響を受けているといえる。

アップサンプリング層に起因するチェス盤歪みの解決のため、滑らかさの制約 [8] や後処理による歪みの除去 [9]、初期化法の開発 [10]、および他のアップサンプリング層の設計 [7, 11–13] など、様々な取り組みがこれまでに行われている。線形インターポレータがチェス盤歪み回避のために満たすべき条件は既に明らかとされているが [14–17]、このチェス盤歪みの回避条件はアップサンプリング層に対して直接適用することができない。そこで菅原らは、線形インターポレータに対するチェス盤歪み回避条件を拡張し、アップサンプリング層におけるチェス盤歪みの回避には二つの方式があることを示した [13]。これらの方式に基づく菅原らの手法は、アップサンプリング層にゼロ次ホールドカーネルを挿入することでチェス盤歪みの完全回避を実現する。しかしながら、チェス盤歪みを回避することがCNNの性能に与える影響は、超解像と画像分類の二つの分野でのみ解析されている状況にあり、GANを用いた画像生成などの分野については未だ調査されていない。加えて、チェス盤歪み回避のため挿入するフィルタカーネルには滑らかさの選択に自由度があるが、その滑らかさがCNNに与える影響についての議論はこれまでに行われていない。

このような背景から、本稿ではまず、CNNにおけるチェス盤歪みの発生を回避するための、新しい固定畳み込み層とそれらを含むCNNを提案する。提案する固定畳み込み層のフィルタカーネルは、ゼロ次ホールドカーネルを複数回畳み込むことで計算され、その畳み込む回数によってフィルタの滑らかさを制御することが可能である。提案する固定畳み込み層を、CNN中のアップサンプリング層あるいはダウンサンプリング層の前後に挿入することで、CNNの順伝搬および逆伝搬処理を実行する際のチェス盤歪みの発生を完全に回避できる。ここで、提案する固定畳み込み層は、チェス盤歪み回避の二つの方式どちらにも適用できる。さらに、本稿では、敵対的生成ネットワーク (Generative adversarial network, GAN) に対して固定畳み込み層を適用し、チェス盤歪みとその回避のためのフィルタカーネルの滑らかさがGANの性能に与える影響をはじめて調査する。

提案する固定畳み込み層の有効性を評価するため画像分類実験および画像生成実験を実施した。実験では、提案

表 1: Notation

Symbol	Definition
a	A scalar
\mathbf{a}	A vector
\mathbf{a}_i	Element i of vector \mathbf{a} , with indexing starting at 1
\mathbf{A}	A matrix
$A_{i,j}$	Element i, j of matrix \mathbf{A}
$A_{i,:}$	Row i of matrix \mathbf{A}
$A_{:,i}$	Column i of matrix \mathbf{A}
\mathbf{A}	A 3- or higher-order tensor
$\mathbf{A}_{i,j,k}$	Element i, j, k of 3-D tensor \mathbf{A}
$\mathbf{A}_{:::,i}$	2-D slice of 3-D tensor \mathbf{A}
\mathbf{V}	A 3-D tensor with a size of $channel \times height \times width$, which denotes an input feature map for a layer.
\mathbf{K}	A 4-D tensor with a size of $output\ channel \times input\ channel \times height \times width$, which denotes a filter kernel (weights) of a convolutional layer.
\mathbf{b}	A vector with a size of $output\ channel$ which denotes a bias of a convolutional layer.
\mathbf{Z}	A 3-D tensor with a size of $output\ channel \times height \times width$, which denotes an output feature map of a layer.
I_i	A single-channel image which is a 2-D slice $\mathbf{V}_{i,:,:}$ of \mathbf{V} .
$h_{i,j}$	A 2-D filter which is a 2-D slice $\mathbf{K}_{i,j,:,:}$ of filter kernel \mathbf{K} .
\mathcal{L}	A loss function
\mathbb{Z}^+	The set of non-negative integers
$\mathbf{A} * \mathbf{B}$	The convolution on two matrixes \mathbf{A} and \mathbf{B}

する固定畳み込み層を ResNet や GAN などの CNN に対して適用し、その性能を評価した。4 種の CNN を用いた画像分類実験より、提案する固定畳み込み層によりチェス盤歪みを回避した CNN の性能が、チェス盤歪みを回避しないものと比べて高い認識精度を持つことを示す。GAN を用いた画像生成実験でも同様に、提案する固定畳み込み層の利用によるチェス盤歪み回避が、生成される画像の品質を高めることが示される。加えて、滑らかな固定畳み込み層を GAN へ適用することにより、生成画像のさらなる品質向上が可能であることが明らかとなる。

2 準備

本章では、CNN とそのチェス盤歪みについて簡単に要約する。本稿では、簡単のため畳み込み層を 2 次元として議論する。また、本稿を通して、表 1 に示す表記法を用いる。

2.1 畳み込みニューラルネットワーク

CNN は、ネットワーク内に 1 層以上の畳み込み層を含む DNN である。畳み込み層の使用は、一般の DNN で用いられる全結合層と比較して、大幅なパラメータの減少に加え可変長入力への適用を可能とするという特長がある。畳み込み層における計算は、以下のように定義される。

$$\mathbf{Z}_{i,j,k} = \sum_{l,m,n} \mathbf{V}_{l,j+m-1,k+n-1} \mathbf{K}_{i,l,m,n} + \mathbf{b}_i \quad (1)$$

ここで、 \mathbf{V} は畳み込み層へ入力される特徴マップであり、 \mathbf{K} と \mathbf{b} はそれぞれ畳み込み層の持つフィルタカーネル (重み)、およびバイアスである。出力特徴マップの i チャンネル目に注目すると、式 (1) の計算は 2 次元フィルタ $h_{i,j} = \mathbf{K}_{i,j,:,:}$ 、および単チャンネル画像 $I_j = \mathbf{V}_{j,:,:}$ を用いて、図 1(a) に示すブロック線図として表せる。

CNN 内では、特徴マップのサンプリングレート変換処理 (解像度変換処理) が一般に用いられる。この処理は、

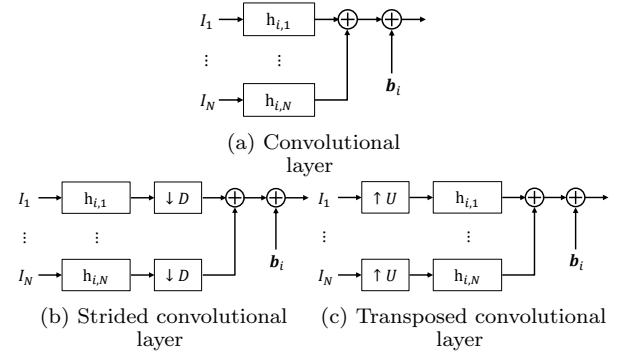


図 1: Block diagrams for convolutional layers

ダウンサンプリング層とアップサンプリング層によって実現される。

代表的なダウンサンプリング層として、ストライドありの畳み込み層や平均プーリング層、最大プーリング層がある [6]。ここで、平均プーリング層は、ストライドありの畳み込み層の特殊な例として考えることができる。また、最大プーリング層をストライドありの畳み込み層で置き換えた場合でも、CNN の性能にはほとんど影響を与えないことが文献 [18] により示されている。これらの理由から、本研究ではダウンサンプリング層としてストライドありの畳み込み層のみに注目する。ストライドありの畳み込み層は、次式により定義される。

$$\begin{aligned} \mathbf{Z}_{i,j,k} &= c(\mathbf{K}, \mathbf{V}, s)_{i,j,k} + \mathbf{b}_i \\ &= \sum_{l,m,n} \mathbf{V}_{l,(j-1)s+m,(k-1)s+n} \mathbf{K}_{i,l,m,n} + \mathbf{b}_i \quad (2) \end{aligned}$$

ここで、 $s \in \mathbb{Z}^+$ はストライドと呼ばれるパラメータである。出力特徴マップの i チャンネル目に注目すると、ストライドありの畳み込み層は、入力特徴マップ I_j とフィルタ $h_{i,j}$ との畳み込みを計算した後に、ダウンサンプリングレート $D = s$ で信号をダウンサンプリングする処理に相当する (図 1(b) 参照)。

一方、アップサンプリング層には、転置畳み込み層やサブサンプリング層 [19], resize convolution layer [7] などがある。中でも、最も代表的な転置畳み込み層は、次式により与えられる。

$$\begin{aligned} \mathbf{Z}_{i,j,k} &= t(\mathbf{K}, \mathbf{V}, s)_{i,j,k} + \mathbf{b}_i \\ &= \sum_{\substack{l,m \\ \text{s.t.} \\ (l-1)s+m=j}} \sum_{\substack{n,p \\ \text{s.t.} \\ (n-1)s+p=k}} \sum_q \mathbf{V}_{q,l,n} \mathbf{K}_{q,i,m,p} + \mathbf{b}_i \quad (3) \end{aligned}$$

ストライドありの畳み込み層とは逆に、転置畳み込み層における出力特徴マップの i チャンネル目の計算は、アップサンプリングレート $U = s$ で入力特徴マップ I_j をアップサンプリングして得られる信号を、フィルタ $h_{i,j}$ で畳み込む処理に相当する (図 1(c) 参照)。

2.2 アップサンプリングとチェス盤歪み

チェス盤歪みは、線形システムにおけるアップサンプリングに起因して発生する周期的な歪みとして研究されてきた [14–17]。

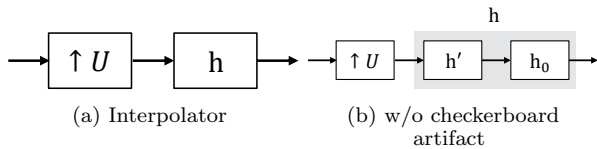


図 2: Linear interpolator and avoiding checkerboard artifact

図 2(a) に示すような、アップサンブラと線形時不変フィルタからなる線形システム (インターポレータ) を考える。チェス盤歪みを発生させるインターポレータは、ある一定の周期で振動する単位ステップ応答を持ち、その周期はアップサンプリングレート U と一致することが知られている。チェス盤歪みを回避するためには、この振動が発生せず単位ステップ応答がある一定値に収束するフィルタ h を用いばよい。そのようなフィルタ h はすべて、カーネルサイズ $U \times U$ のゼロ次ホールドカーネル h_0 を用いて、次式の制約を満たす (必要十分条件)。

$$h = h_0 * h' \quad (4)$$

これは、 h の分解によって生成される U 個のポリフェーズフィルタの直流利得がすべて等しいことと同値である。

図 2(b) におけるフィルタ h' の単位ステップ応答の定常値は、一般に周期 U の周期性を持つ。この応答に対し、サイズ U のゼロ次ホールドカーネル h_0 を畳み込むことは、その入力における各サンプルについて連続的に U 点平均を計算することに相当する。よって、 h_0 の出力は、その入力が周期的であった場合にも一定値となる。したがって、フィルタ $h = h_0 * h'$ の単位ステップ応答はある一定値に収束するため (単位ステップ応答の定常値が周期性を持たず)、このインターポレータはチェス盤歪みを発生させない。

一方、転置畳み込み層にあるような、インターポレータ (図 3(a)) では、出力にバイアス b を加算するため非線形システムとなる。このとき、チェス盤歪みの発生を回避するためには、二つの方式が考えられる。

方式 1 フィルタ h' の直後にゼロ次ホールドカーネル h_0 を挿入する (図 3(b))。これは、線形システムのと看同様、 h に式 (4) の制約を与えることに相当する。

方式 2 フィルタ h には制約をかけずに、バイアス b の加算後に h_0 を挿入する (図 3(c))。

ここで、方式 1 および方式 2 は、互いに等価変換できないことに注意する。

2.3 畳み込みニューラルネットワークにおけるチェス盤歪み

A. 順伝搬と逆伝搬

CNN におけるチェス盤歪みには、順伝搬時に発生するものと逆伝搬時に発生するものの 2 種類がある。CNN における信号の順伝搬とは、畳み込み層などによって入力特徴マップ V から新しい特徴マップ Z を計算し、次の層へ伝搬させる一連の処理を指す。順伝搬時に発生するチェ

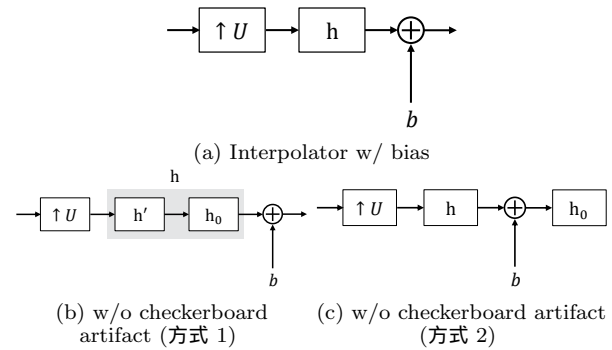


図 3: Non-linear interpolator with bias and avoiding checkerboard artifact

ス盤歪みは、転置畳み込み層に起因する。図 1(c) に示したように、転置畳み込み層はインターポレータを含む。理想的には、畳み込み層におけるフィルタカーネルは式 (4) の制約を満たすよう学習されるべきであるが、実際には多くの場合そのように学習されない。そのため、式 (4) に示す条件を満たさないインターポレータが CNN の出力特徴マップに対してチェス盤歪みを発生させる。

一方、CNN における信号の逆伝搬とは、CNN 中の各パラメータに関する損失関数 \mathcal{L} の勾配を、出力側の層から入力側の層へ順に計算する一連の処理である (誤差逆伝搬)。逆伝搬時に発生するチェス盤歪みは、ストライドありの畳み込み層に起因する。誤差逆伝搬において、ストライドありの畳み込み層は、出力側の層から与えられる 3 階の勾配テンソル G とフィルタカーネル K を用いて、入力特徴マップ V に関する損失関数 \mathcal{L} の勾配を次式により計算する。

$$\frac{\partial}{\partial V_{i,j,k}} \mathcal{L}(V, K) = t(K, G, s)_{i,j,k} \quad (5)$$

式 (5) からわかる通り、ストライドありの畳み込み層における損失関数の勾配の計算は、転置畳み込み処理を用いて行われる。この理由から、順伝搬時と同様、計算される勾配にはチェス盤歪みが含まれる。したがって、この勾配を用いて学習されるフィルタカーネル K および b は、チェス盤歪みの影響を受ける。

B. 信号とモデルへの影響

CNN の多くは、アップサンプリング層のみを含むもの、ダウンサンプリング層のみを含むもの、およびそれら両方を含むものの 3 つに分けることができる。

アップサンプリング層のみを含む CNN は、主に画像超解像に用いられる。このような CNN では、アップサンプリング層に起因する順伝搬時のチェス盤歪みが、各アップサンプリング層の出力特徴マップおよび最終的な出力画像に現れる。さらに、このチェス盤歪みは CNN モデルの学習にも影響を与える。画像超解像における CNN の学習は、CNN の出力画像を用いて計算される損失関数に基づく。この出力画像にはチェス盤歪みが含まれるため、計算される損失関数とその勾配はチェス盤歪みの影響を受けたものになる。

ダウンサンプリング層のみを含む CNN は、主に画像分類などのクラス分類問題に用いられる。このような CNN

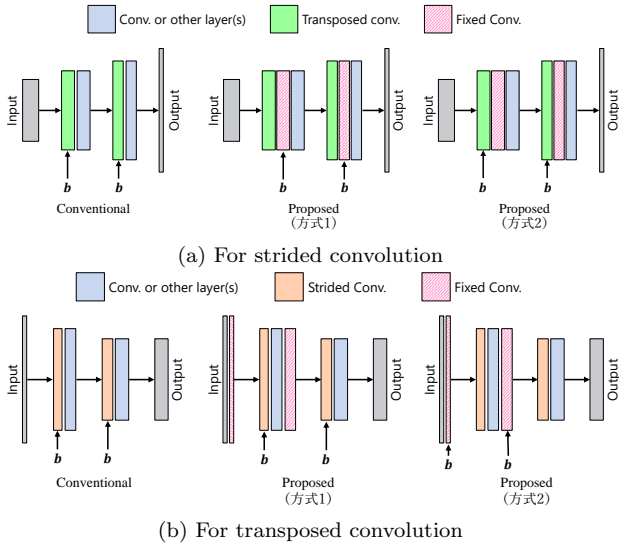


図 4: Usage of proposed fixed convolution layer

では、順伝搬時にチェス盤歪みが発生せず、出力結果もクラスラベル等の単一の値であるため、チェス盤歪みの影響はないように見える。しかしながら、画像分類のための CNN では、ダウンサンプリング層に起因して逆伝搬時に発生する勾配へのチェス盤歪みが、フィルタカーネル K および b の学習に影響を与える。

アップサンプリング層およびダウンサンプリング層の両方を含む CNN には、画像生成に用いられる GAN [20] や画像変換に用いられる U-Net [5] などがある。これらの CNN は、上記のチェス盤歪みの影響をすべて受けることになる。

C. 固定畳み込み層の滑らかさ

菅原らは、ゼロ次ホールドカーネルを固定層として利用することで、チェス盤歪みの完全回避を実現した [13]。ゼロ次ホールドカーネルを畳み込むことで得られるすべてのフィルタは、チェス盤歪みの回避を可能とする。換言すれば、チェス盤歪みの回避に用いるフィルタの特性 (滑らかさの選択) には自由度がある。しかしながら、その滑らかさが CNN の性能に与える影響はこれまでに議論されていない。

本稿では、チェス盤歪みの回避のための、ゼロ次ホールドカーネルの畳み込みに基づく新しい固定畳み込み層を提案する。加えて、固定畳み込み層におけるフィルタカーネルの滑らかさが CNN の性能に与える影響を実験的に調査する。

3 提案法

図 4 に、提案する固定畳み込み層の利用例を示す。提案する固定畳み込み層は、チェス盤歪みの回避条件を満たす固定のカーネル K を持つ畳み込み層である。このカーネル K は、 h_0 を複数回畳み込むことによって生成される。 h_0 を畳み込む回数によって、 K の滑らかさを制御できる。

3.1 固定畳み込み層

入力特徴マップの i チャネル目に注目したとき、固定畳み込み層のカーネル $K^{(d)}$ は、次式により計算する。

$$K_{i,i,\dots}^{(d)} = \begin{cases} h_0 & (d = 0) \\ K_{i,i,\dots}^{(d-1)} * h_0 & (d > 1) \end{cases} \quad (6)$$

ここで、 h_0 のカーネルサイズは、適用するアップサンプリング層におけるアップサンプリングレート U を用いて、 $U \times U$ として与える。ダウンサンプリング層に適用する場合には、ダウンサンプリングレート D を U の代わりに用いて、 $D \times D$ としてカーネルサイズを与える。固定畳み込み層の出力特徴マップ Z は、カーネル $K^{(d)}$ と学習可能なバイアス b を用いて次式により計算される。

$$Z_{i,j,k} = \sum_{m,n} V_{i,j+m-1,k+n-1} K_{i,i,m,n}^{(d)} + b_i \quad (7)$$

3.2 チェス盤歪みの固定畳み込み層による回避

転置畳み込み層におけるチェス盤歪みは、2.2 節で議論した通り、方式 1 および 2 の二つの方法を用いて回避することができる。

転置畳み込み層のチェス盤歪みを、固定畳み込み層を用いて回避する方法を以下に示す (図 4(a) 参照)。

方式 1 転置畳み込み層のバイアス b をすべて 0 で固定し、それら転置畳み込み層の直後に、学習可能なバイアス b を持つ固定畳み込み層をそれぞれ挿入する。

方式 2 学習可能なバイアス b を持つ転置畳み込み層の直後に、バイアス b を 0 で固定した固定畳み込み層をそれぞれ挿入する。

方式 1 を用いた場合には、ポリフェーズ分解によってインターポレータの冗長性を取り除くことが可能であり、効率的な実装による計算コストの削減が期待できる。しかし、畳み込み層の後には Batch normalization (BN) [21] を適用することが一般的であり、BN の効果によってバイアス b の効果が打ち消されてしまう。方式 2 を用いた場合には、バイアス b が BN の影響を受けない。

ストライドありの畳み込み層におけるチェス盤歪みは誤差逆伝搬時に発生することから、転置畳み込み層の場合とは逆に、ストライドありの畳み込み層の直前に固定畳み込み層を挿入することでチェス盤歪みを回避できる (図 4(b) 参照)。このとき、転置畳み込み層のときと同様に、バイアスを加算する位置によって方式 1 および 2 の二通りを考えることができる。

4 実験

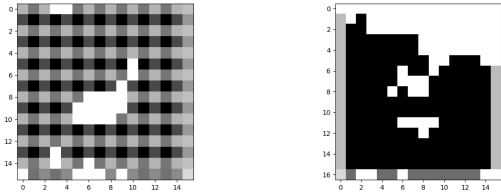
提案する固定畳み込み層によるチェス盤歪みの回避が CNN の性能に与える影響を評価するため、画像分類および画像生成の二つのタスクに対する実験を行った。

4.1 画像分類

誤差逆伝搬時のチェス盤歪みを回避することの有効性を確かめるため、提案する固定畳み込み層を四つの CNN に対して適用し、それらによる画像分類の精度を評価した。ここで、固定畳み込み層のカーネルはすべて $K^{(0)}$ とした。評価のため使用した四つの CNN を以下に示す。

表 2: Architecture of Simple CNN. ch denotes the number of output channels.

Layer	Stride s	Kernel size	ch
Conv. + BN + ReLU	2	3×3	64
Conv. + BN + ReLU	2	3×3	128
Conv. + BN + ReLU	2	3×3	256
Conv. + BN + ReLU	2	3×3	512
Conv. + BN + ReLU	1	2×2	10



(a) w/o fixed conv. layer (b) with fixed conv. layer

図 5: Unit step response of second convolutional layer in Simple CNN during backpropagation

表 3: Image-classification accuracy for CIFER10 (train/test)

Network	Conv.	Prop. (方式 1)	Prop. (方式 2)
Simple CNN	0.999/0.585	0.93/0.645	0.976/ 0.671
VGG8	1.000/0.795	1.000/ 0.840	1.0000/0.839
ResNet-18	1.000/0.812	1.000/ 0.862	1.000/ 0.862
ResNet-101	1.000/0.852	1.000/ 0.863	1.000/ 0.863

1. 表 2 に示す構成を持つ単純な CNN (Simple CNN). ここで, ReLU は Rectified linear unit activation function を指す [22].
2. VGG8
3. ResNet-18
4. ResNet-101

本実験では, 学習および評価のためのデータセットとして, CIFER10 [23] のトレーニングセット 5 万枚およびテストセット 1 万枚をそれぞれ利用した. ここで, 学習においてデータ拡張は実施しなかった. 各 CNN は, Adam 最適化法 [24] を用いて 300 エポックの学習を行った. Adam のパラメータには, 初期学習率 0.1, $\beta_1 = 0.9$, $\beta_2 = 0.999$ を利用した. ただし, VGG8 にのみ初期学習率 0.01 を用いた. また学習率は, エポック数が 150 および 225 に到達した時点でそれぞれ 1/10 倍した.

図 5 に, 提案する固定畳み込み層を用いた場合および用いなかった場合それぞれについて, Simple CNN の第 2 層目の逆伝搬における単位ステップ応答を示す. 図 5(a) より, 計算された勾配にチェス盤歪みが発生していることがわかる. このチェス盤歪みは, 提案する固定畳み込み層を用いることで完全に回避できる (図 5(a) 参照).

表 3 に, 画像分類の精度を示す. 表 3 より, 実験で用いたすべての CNN において, 提案する固定畳み込み層を用いた場合にテストデータに対する分類精度が向上した. この分類精度の向上は, チェス盤歪みを回避するための二つの方式のどちらでも確認できた. したがって, 従来の画像分類のための CNN は, チェス盤歪みの影響を受けていたといえる.

表 4: Generator architecture. ch denotes the number of output channels.

Layer	Stride s	Kernel size	ch
Trans. conv. + BN + Leaky ReLU	2	4×4	512
Trans. conv. + BN + Leaky ReLU	2	4×4	512
Trans. conv. + BN + Leaky ReLU	2	4×4	512
Trans. conv. + BN + Leaky ReLU	2	4×4	256
Trans. conv. + BN + Leaky ReLU	2	4×4	128
Trans. conv. + BN + Leaky ReLU	2	4×4	64
Conv. + Tanh	1	3×3	3

表 5: Discriminator Architecture. ch denotes the number of output channels.

Layer	Stride s	Kernel size	ch
Conv. + BN + Leaky ReLU	2	4×4	8
Conv. + BN + Leaky ReLU	2	4×4	32
Conv. + BN + Leaky ReLU	2	4×4	64
Conv. + BN + Leaky ReLU	2	4×4	64
Conv.	1	4×4	1

4.2 画像生成

画像生成実験では, 敵対的生成ネットワーク (Generative adversarial network, GAN) を用いた [20, 25]. GAN は一般に, Generator と Discriminator という二つのネットワークから構成される. ここで, Generator は転置畳み込み層を含み, Discriminator はストライドありの畳み込み層を含むことが一般的である. したがって, GAN の学習においては, 順伝搬と逆伝搬の双方においてチェス盤歪みが発生する. このチェス盤歪みを回避するため, 提案する固定畳み込み層を GAN に適用した. 固定畳み込み層のカーネルは $K^{(0)}$, $K^{(1)}$, $K^{(2)}$ の三種類を用いた.

本実験で用いた Generator および Discriminator の構成を, 表 4 および 5 にそれぞれ示す. それらネットワーク中の活性化関数には, Leaky ReLU [26] を用いた. 学習のためのデータセットには CelebA [27] を用い, 正規分布に従うランダムな 100 次元ベクトルから顔画像を生成するように学習した. 学習では, Generator と Discriminator 共に, 学習率 0.08, $\beta_1 = 0.5$, $\beta_2 = 0.9$ とした Adam 最適化法 [24] を用いて 10 epoch の学習を行った.

図 6 に, 学習された GAN を用いてランダムなノイズ 64 個から生成された 64 枚の顔画像を示す. 図より, 従来の GAN によって生成された画像にはチェス盤歪みを含む一方, 提案する固定畳み込み層を適用した GAN はすべてチェス盤歪みが発生していないことがわかる. また, 提案する固定畳み込み層を用いることで, GAN により生成される画像品質が向上していることが確認できる. 図 6(b), (c), および (c) を比較すると, フィルタカーネルが $K^{(1)}$ および $K^{(2)}$ の場合により明瞭な顔画像を生成できていることがわかる. このことから, 滑らかなフィルタカーネルを持つ固定畳み込み層を用いることで, GAN のさらなる性能向上が可能となるといえる.

5 おわりに

本稿では, CNN におけるチェス盤歪み回避のための新しい固定畳み込み層と, それら固定畳み込み層を含む新しい CNN を提案した. 提案した固定畳み込み層は, 特徴マップのサンプリングレート変換に起因するチェス盤歪みの発生を完全に防ぐことができる. 提案した固定畳み

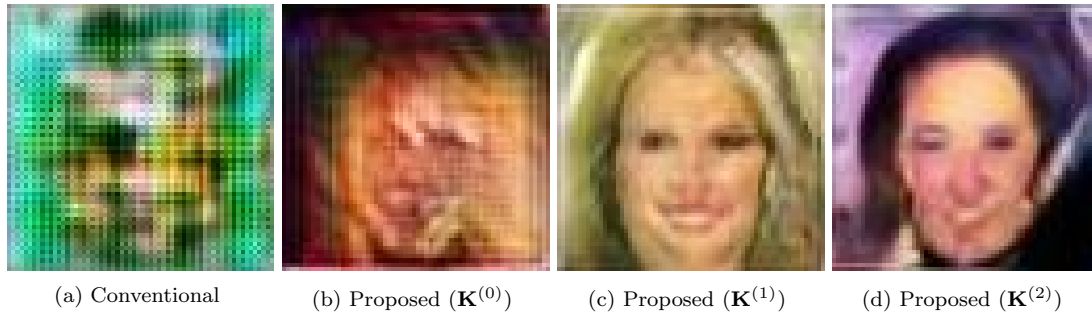


図 6: Generated images by using GAN

込み層におけるフィルタカーネルの滑らかさは、フィルタカーネル計算時のゼロ次ホールドカーネルを畳み込む回数により制御できる。画像分類実験および画像生成実験により、提案する固定畳み込み層がチェス盤歪みの発生を回避できることが示された。また、画像分類実験の結果、提案する固定畳み込み層の利用は、実験に用いた4種すべてのCNNの性能を向上させることが示された。したがって、従来の画像分類のためのCNNは、チェス盤歪みの影響を受けていたといえる。加えて、GANを用いた画像生成実験の結果、固定畳み込み層のフィルタカーネルの滑らかさは、GANによって生成される画像の品質に大きな影響を与えることが示された。

参考文献

- [1] Y. LeCun, "Generalization and Network Design Strategies," Technical report, Department of Computer Science, University of Toronto, 1989.
- [2] A. Krizhevsky, I. Sutskever, and G.E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," Proceedings of International Conference on Neural Information Processing Systems, pp.1097–1105, 2012.
- [3] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," arXiv:1409.1556, Sep. 2014. <http://arxiv.org/abs/1409.1556>
- [4] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp.770–778, Jun. 2016.
- [5] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation," Medical Image Computing and Computer-Assisted Intervention (MICCAI), vol.9351, pp.234–241, LNCS, Springer, Nov. 2015.
- [6] I. Goodfellow, Y. Bengio, and A. Courville, Deep learning, MIT Press, 2016.
- [7] A. Odena, V. Dumoulin, and C. Olah, "Deconvolution and Checkerboard Artifacts," 2016. <https://distill.pub/2016/deconv-checkerboard/>
- [8] A. Dosovitskiy, P. Fischer, E. Ilg, P. Hausser, C. Hazirbas, V. Golkov, P. van derSmagt, D. Cremers, and T. Brox, "FlowNet: Learning Optical Flow with Convolutional Networks," Proceedings of IEEE International Conference on Computer Vision, pp.2758–2766, Dec. 2015.
- [9] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual Losses for Real-Time Style Transfer and Super-Resolution," Proceedings of European Conference on Computer Vision, pp.694–711, 2016.
- [10] A. Aitken, C. Ledig, L. Theis, J. Caballero, Z. Wang, and W. Shi, "Checkerboard artifact free sub-pixel convolution: A note on sub-pixel convolution, resize convolution and convolution resize," arXiv:1707.02937, Jul. 2017. <http://arxiv.org/abs/1707.02937>
- [11] H. Gao, H. Yuan, Z. Wang, and S. Ji, "Pixel Deconvolutional Networks," arXiv:1705.06820, May 2017. <http://arxiv.org/abs/1705.06820>
- [12] Z. Wojna, V. Ferrari, S. Guadarrama, N. Silberman, L.-C. Chen, A. Fathi, and J. Uijlings, "The Devil is in the Decoder: Classification, Regression and GANs," arXiv:1707.05847, Jul. 2017. <http://arxiv.org/abs/1707.05847>
- [13] Y. Sugawara, S. Shiota, and H. Kiya, "Checkerboard artifacts free convolutional neural networks," APSIPA Transactions on Signal and Information Processing, vol.8, no.e9, Feb. 2019.
- [14] Y. Harada, S. Muramatsu, and H. Kiya, "Multidimensional Multirate Filter without Checkerboard Effects," Proceedings of European Signal Processing Conference, pp.1881–1884, 1998.
- [15] T. Tamura, M. Kato, T. Yoshida, and A. Nishihara, "Design of Checkerboard-Distortion-Free Multidimensional Multirate Filters," IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences, vol.E81-A, no.8, pp.1598–1606, Aug. 1998.
- [16] Y. Harada, S. Muramatsu, and H. Kiya, "Multidimensional Multirate Filter and Filter Bank without Checkerboard Effect," IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences, vol.E81-A, no.8, pp.1607–1615, Aug. 1998.
- [17] H. IWAI, M. IWAHASHI, and H. KIYA, "Methods for Avoiding the Checkerboard Distortion Caused by Finite Word Length Error in Multirate System," IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences, vol.E93-A, no.3, pp.631–635, 2010.
- [18] J.T. Springenberg, A. Dosovitskiy, T. Brox, and M. Riedmiller, "Striving for Simplicity: The All Convolutional Net," arXiv:1412.6806, Dec. 2014. <http://arxiv.org/abs/1412.6806>
- [19] W. Shi, J. Caballero, F. Huszar, J. Totz, A.P. Aitken, R. Bishop, D. Rueckert, and Z. Wang, "Real-Time Single Image and Video Super-Resolution Using an Efficient Sub-Pixel Convolutional Neural Network," Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp.1874–1883, Jun. 2016.
- [20] 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, eds. by Z. Ghahramani, M. Welling, C. Cortes, N.D. Lawrence, and K.Q. Weinberger, pp.2672–2680, 2014.
- [21] S. Ioffe and C. Szegedy, "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift," arXiv:1502.03167, Feb. 2015. <http://arxiv.org/abs/1502.03167>
- [22] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier neural networks," Proceedings of the International Conference on Artificial Intelligence and Statistics, pp.315–323, Apr. 2011.
- [23] A. Krizhevsky and G.E. Hinton, "Learning multiple layers of features from tiny images," Master's thesis, University of Toronto, 2009.
- [24] D.P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," arXiv:1412.6980, Dec. 2014. <http://arxiv.org/abs/1412.6980>
- [25] A. Radford, L. Metz, and S. Chintala, "Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks," arXiv:1511.06434, Nov. 2015. <http://arxiv.org/abs/1511.06434>
- [26] B. Xu, N. Wang, T. Chen, and M. Li, "Empirical Evaluation of Rectified Activations in Convolutional Network," arXiv:1505.00853, May 2015. <http://arxiv.org/abs/1505.00853>
- [27] Z. Liu, P. Luo, X. Wang, and X. Tang, "Deep Learning Face Attributes in the Wild," Proceedings of International Conference on Computer Vision, Dec. 2015.