

アップスケーリング処理を含む畳み込みニューラルネットワークのための チェス盤ひずみ回避法

菅原 佑介[†] 塩田さやか[†] 貴家 仁志[†]

[†] 首都大学東京システムデザイン研究科 〒191-0065 東京都日野市旭ヶ丘 6-6

E-mail: [†]sugawara-yusuke@ed.tmu.ac.jp, ^{††}{sayaka,kiya}@tmu.ac.jp

あらまし 超解像処理に代表される、アップスケーリング処理を含む畳み込みニューラルネットワークにおいて、チェス盤ひずみが発生することが知られている。畳み込みニューラルネットワークにおけるチェス盤ひずみは、Deconvolution 層や Sub-pixel Convolution 層などの Upsampling 層により引き起こされるひずみである。本稿では、アップスケーリング処理を含む畳み込みニューラルネットワークにおけるチェス盤ひずみについて考察し、ひずみを回避するための条件を示す。さらに、Deconvolution 層や Sub-pixel Convolution 層の直後に、指定された条件を満たすフィルタを内包する畳み込み層を追加することで、チェス盤ひずみが回避できることを示す。提案法の有効性を確認するため、畳み込みニューラルネットワークを用いた最先端の超解像法に、提案するチェス盤ひずみ回避法を適用して、画質評価実験を実施する。実験では、提案法によってチェス盤ひずみが完全に回避可能であることが確認される。

キーワード 畳み込みニューラルネットワーク, チェス盤ひずみ

Avoiding checkerboard artifact in convolutional neural networks including upsampling layers

Yusuke SUGAWARA[†], Sayaka SHIOTA[†], and Hitoshi KIYA[†]

[†] Faculty of System Design, Tokyo Metropolitan University 6-6, Asahigaoka, Hino-shi, Tokyo, 191-0065 Japan

E-mail: [†]sugawara-yusuke@ed.tmu.ac.jp, ^{††}{sayaka,kiya}@tmu.ac.jp

Abstract In convolutional neural networks with upsampling layers, represented by super-resolution methods, it is known that checkerboard artifacts are generated. These artifacts are caused in upsampling layers such as deconvolution and sub-pixel convolution layers. A condition to avoid the checkerboard artifacts for convolutional neural networks is considered in this paper. Moreover, it is shown that the artifacts can be avoided by adding the convolutional layer with a filter which satisfies the condition. To confirm the effectiveness of the proposed methods, a number of experiments on state-of-the-art super-resolution methods are carried out. The experimental results demonstrate that the proposed method can completely overcome checkerboard artifacts.

Key words Convolutional Neural Network, Checkerboard Artifact

1. ま え が き

畳み込みニューラルネットワーク (Convolutional Neural Network, CNN) [1] は畳み込み層, プーリング層, 全結合層から構成される多層ニューラルネットワークの一種であり, 主にコンピュータビジョン及び画像処理の分野において用いられている。近年, 画像生成 [2], 領域分割 [3], 超解像 [4], [5] などの様々な応用においてアップスケーリング処理を含む畳み込みニューラルネットワークを用いた手法が注目され, 盛んに研究されている。例えば, 超解像では, アップスケーリング処理を含む CNN

を用いることで低解像度画像から高解像度画像を直接生成することが可能となり, アップスケーリング処理を CNN 中に含まない従来の超解像法 [4] と比較して, より高速な超解像が実現されている [6], [7]。これらの手法では, そのアップスケーリング処理を実現する Upsampling 層 (Deconvolution 層 [8] や Sub-pixel Convolution 層 [7]) を含む。

しかしながら, このようなアップスケーリング処理を含む CNN が生成する画像には, チェス盤状の周期的なひずみが生じることが知られている [9]。このひずみは, チェス盤ひずみと呼ばれ, Deconvolution 層や Sub-pixel Convolution 層など

の Upsampling 層で発生する。また、この問題を解決するための手法がいくつか提案されている [9], [10]。特に, Nearest-Neighbor の補間後に畳み込み層の演算を行う Upsampling 層 (Resize Convolution 層 [9]) がある。しかし、この手法の有効性は Resize Convolution 層に限定されており, Deconvolution 層や Sub-pixel Convolution 層などに適用できない。

一方、チェス盤ひずみの先行研究として、線形な処理システムであるマルチレートシステムに対する成果がある [11] ~ [13]。チェス盤ひずみは、マルチレートシステムを構成するインターポレータの周期的な時変性に起因して発生するひずみとして知られており、ひずみを回避するための条件や、多次元フィルタ [12]、多次元フィルタバンク [13] などへの拡張が検討されている。

本稿では、アップスケーリング処理を含む CNN におけるチェス盤ひずみを回避するための手法を考察する。CNN と先行研究でのマルチレートシステムとの大きな違いは、多層構成の学習システムであることと、非線形な処理であることである。Deconvolution 層などが複数のインターポレータから構成されるマルチレートシステムを用いて表現できることに着目し、ひずみを回避するための条件を導出する。さらに, Deconvolution 層や Sub-pixel Convolution 層の直後に、指定された条件を満たすフィルタを内包する畳み込み層を追加することで、チェス盤ひずみが回避できることを示す。超解像による画質評価実験を実施し、提案法によってチェス盤ひずみが完全に回避可能であることを確認する。

2. 準備

ここでは、アップスケーリング処理を含む CNN を用いた超解像 [6] [7] を例にして議論を展開する。また簡単のため、本稿ではグレースケールの画像を扱う。

2.1 アップスケーリング処理を含む CNN 超解像

図 1 に示すように、低解像度画像 I_{LR} は複数の畳み込み層と単一の Upsampling 層で構成された CNN に入力される。本稿では、[7] と同様に畳み込み層の層数は 2 層とする。第 1 層の特徴マップ $f^{(1)}(I_{LR})$ は、低解像度画像 I_{LR} から次式に従って求められる。

$$f_n^{(1)}(I_{LR}) = \max(0, I_{LR} * W_{c,n}^{(1)} + b_n^{(1)}) \quad (1)$$

$$c = C_1, n = 1, 2, \dots, N_1$$

ここで、演算子 $*$ は畳み込み演算を表し、 $\max(0, \cdot)$ は活性化関数 ReLU に対応する。また、 $W_{c,n}^{(1)}$ 、 $b_n^{(1)}$ はそれぞれ第 1 層のフィルタ、バイアスを表す。フィルタ $W_{c,n}^{(1)}$ のサイズは $C_1 \times K_1 \times K_1 \times N_1$ である。ただし、 C_1 は第 1 層のチャンネル数を表し $C_1 = 1$ とする。また、 K_1 は第 1 層のフィルタサイズ、 N_1 は第 1 層のフィルタ数を表す。また、インデックス c, n はチャンネル番号、フィルタ番号にそれぞれ対応している。第 2 層の特徴マップ $f^{(2)}(I_{LR})$ は、第 1 層の特徴マップ $f^{(1)}(I_{LR})$ から次式に従って求められる。

$$f_n^{(2)}(I_{LR}) = \max(0, \sum_{c=1}^{C_2} f_c^{(1)}(I_{LR}) * W_{c,n}^{(2)} + b_n^{(2)}) \quad (2)$$

$$n = 1, 2, \dots, N_2$$

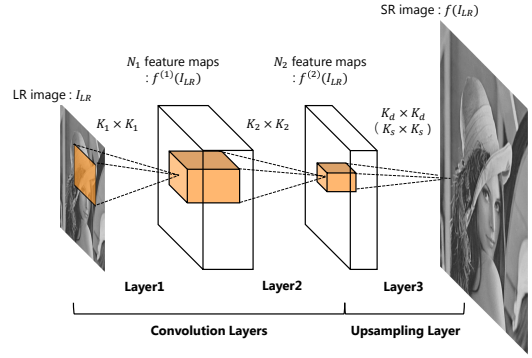


図 1 アップスケーリング処理を含む CNN を用いた超解像 [6] [7]

第 2 層のフィルタ $W_{c,n}^{(2)}$ のサイズは、 $C_2 \times K_2 \times K_2 \times N_2$ である。ここで、第 2 層のチャンネル数 C_2 は $C_2 = N_1$ が選択される。超解像画像 $f(I_{LR})$ は、第 2 層の特徴マップ $f^{(2)}(I_{LR})$ に対し Upsampling 層での拡大処理を施すことで与えられる。Upsampling 層の計算は、使用するアルゴリズムに依存しており、アルゴリズム毎に特別な名称で呼ばれることが多い。Deconvolution 層 [8] と Sub-pixel Convolution 層 [7] は、その代表的な手法である。

2.2 Upsampling 層

ここでは、2 つの代表的な Upsampling 層について要約する。

A. Deconvolution 層

Deconvolution 層 [8] によって生成される超解像画像 $f^{(d)}(I_{LR})$ は、第 2 層の特徴マップ $f^{(2)}(I_{LR})$ から次式に従って求められる。

$$f^{(d)}(I_{LR}) = \sum_{c=1}^{C_d} \uparrow U(f_c^{(2)}(I_{LR})) * W_{c,n}^{(d)} + b_n^{(d)} \quad (3)$$

$$n = N_d$$

ここで、演算子 $\uparrow U(\cdot)$ は拡大率 U でのアップサンプリング (ゼロ値挿入) を表す。Deconvolution 層のフィルタ $W_{c,n}^{(d)}$ のサイズは $C_d \times K_d \times K_d \times N_d$ であり、チャンネル数 C_d は $C_d = N_2$ が選択される。フィルタ数 N_d は $N_d = 1$ とする。また本稿では議論を簡潔にするため、畳み込み層と Deconvolution 層における相関と畳み込みの演算の違いは区別せず、畳み込み演算 $*$ で統一する。

B. Sub-pixel Convolution 層

Sub-pixel Convolution 層 [7] は 2 つのステップからなる。まず畳み込み層の演算により、特徴マップ $\hat{f}^{(s)}(I_{LR})$ を第 2 層の特徴マップ $f^{(2)}(I_{LR})$ から次式に従って求める。

$$\hat{f}_n^{(s)}(I_{LR}) = \sum_{c=1}^{C_s} f_c^{(2)}(I_{LR}) * W_{c,n}^{(s)} + b_n^{(s)} \quad (4)$$

$$n = 1, 2, \dots, U^2$$

フィルタ $W_{c,n}^{(s)}$ のサイズは $C_s \times K_s \times K_s \times N_s$ であり、チャンネル数 C_s は $C_s = N_2$ が選択される。Deconvolution 層とは異なり、フィルタ数 N_s は $N_s = U^2$ とする。次に、この特徴マップ $\hat{f}^{(s)}(I_{LR})$ から、超解像画像 $f^{(s)}(I_{LR})$ を次式に従って計算する。

$$f^{(s)}(I_{LR}) = PS(\hat{f}_1^{(s)}(I_{LR}), \hat{f}_2^{(s)}(I_{LR}), \dots, \hat{f}_{U^2}^{(s)}(I_{LR})) \quad (5)$$

ここで演算子 $PS(\cdot)$ は、 U^2 チャンネルの特徴マップを 1 チャンネルのものに並び替える操作である [7].

C. Deconvolution 層と Sub-pixel Convolution 層の関係

ここでは、Deconvolution 層と Sub-pixel Convolution 層の入出力関係が等価であることを説明する。まず Sub-pixel Convolution 層のフィルタ $W_{c,n}^{(s)}$ 、バイアス $b_n^{(s)}$ から次式により与えられるフィルタ $W_{c,n}$ とバイアス b_n を考える。

$$\begin{aligned} W_{c,n} &= PS(W_{c,1}^{(s)}, W_{c,2}^{(s)}, \dots, W_{c,U^2}^{(s)}) \\ b_n &= PS(b_1^{(s)}, b_2^{(s)}, \dots, b_{U^2}^{(s)}) \quad c = 1, 2, \dots, C_s, n = 1 \end{aligned} \quad (6)$$

このとき、フィルタ $W_{c,n}$ のサイズは $C_s \times UK_s \times UK_s \times 1$ となる。さらに、Sub-pixel Convolution 層より生成される超解像画像 $f^{(s)}(I_{LR})$ は、式 (3) において $W_{c,n}^{(d)}, b_n^{(d)}$ に $W_{c,n}, b_n$ をそれぞれ代入した値との間に、次式が成り立つ [7].

$$f^{(s)}(I_{LR}) = f^{(d)}(I_{LR}) \quad (7)$$

以上から、Sub-pixel Convolution 層は、その入出力関係が等価な Deconvolution 層が存在する。すなわち、Deconvolution 層または Sub-pixel Convolution 層において、その学習済みのフィルタ及びバイアスが与えられたとき、入出力関係が等価な他方の表現がお互いに可能である。ただし、式 (6) のバイアス b_n は、Deconvolution 層のバイアス $b_n^{(d)}$ と異なり一定値になるとは限らないことに注意する。

2.3 チェス盤ひずみと Upsampling 層の関係

式 (4) が示す通り、特徴マップ $\hat{f}^{(s)}(I_{LR})$ は n によって異なる入出力特性を持つ。換言すると、このことは、 $\hat{f}^{(s)}(I_{LR})$ が一般に周期的な応答を持つことになる。このことが、Deconvolution 層及び Sub-pixel Convolution 層において、式 (5) より与えられる超解像画像に周期的なひずみ (チェス盤ひずみ) を引き起こす。一方、畳み込み層ではこのように入出力特性が周期的に変化しない。したがって、アップスケーリング処理を含む CNN においてチェス盤ひずみが発生する計算箇所は、Deconvolution 層や Sub-pixel Convolution 層などの Upsampling 層に限定される。

3. 提案手法

アップスケーリング処理を含む CNN のためのチェス盤ひずみ回避法を提案する。

3.1 チェス盤ひずみの発生メカニズム

Deconvolution 層におけるチェス盤ひずみの問題 [9] は、その入出力関係が等価な Sub-pixel Convolution 層を考えることにより明らかにできる。ここでは、Upsampling 層においてチェス盤ひずみが発生する理由を図 2 のステップ応答を用いて説明する。

図 1 に示した CNN において、その入力が 2 次元単位ステップ信号 I_{step} である場合を考える (図 2(a) 参照)。このとき、第 2 層の特徴マップ $f^{(2)}(I_{step})$ は、すべてのチャンネルにおいて 2 次元ステップ信号となり、係数 $A_c \geq 0$ を用いて $A_c I_{step}$ と表される。ここで、次の層が Deconvolution 層であるとすると、その入出力関係が等価な Sub-pixel Convolution 層を考えることに

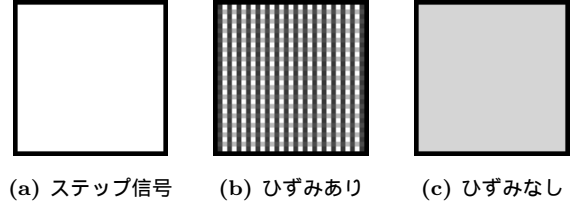


図 2 ステップ信号とそのステップ応答

より、式 (4) で与えられる特徴マップ $\hat{f}^{(s)}(I_{step})$ は、画像端付近の画素を除いて次式で表される。

$$\hat{f}_n^{(s)}(I_{step}) = \sum_{c=1}^{C_s} A_c \bar{W}_{c,n}^{(s)} + b_n^{(s)} \quad n = 1, 2, \dots, U^2 \quad (8)$$

ただし、 $\bar{W}_{c,n}^{(s)}$ における座標 (i', j') での値 $\bar{W}_{c,n}^{(s)}(i', j')$ は、次式のように 2 次元フィルタの係数 $W_{c,n}^{(s)}$ の値の総和となる。

$$\bar{W}_{c,n}^{(s)}(i', j') = \sum_{i=1}^{K_s} \sum_{j=1}^{K_s} W_{c,n}^{(s)}(i, j) \quad (9)$$

式 (8) より、2 次元ステップ信号 I_{step} に対する特徴マップ $\hat{f}^{(s)}(I_{step})$ は一定値をとるが、一般にその値は n によって異なる。そのため、式 (5) より与えられる超解像画像は、各軸周期 U で異なる値をとり、任意のチャンネル c で

$$\bar{W}_{c,1}^{(s)} = \bar{W}_{c,2}^{(s)} = \dots = \bar{W}_{c,U^2}^{(s)} \quad (10)$$

が成立しない限り、一定値とはならない (図 2(b) 参照)。このことに起因して、チェス盤ひずみが発生する。

一方、式 (10) が成立する (式 (8) の右辺第 1 項がすべての n で等しい) とき、2 次元単位ステップ信号 I_{step} を入力として与えた場合の超解像画像は一定値をとり、チェス盤ひずみは回避される (図 2(c) 参照)。ここで、Deconvolution 層のバイアス $b_n^{(d)}$ は一定値をとるため、式 (8) の右辺第 2 項はすべての n で常に等しいことに注意する。以降では、この事実に基づくチェス盤ひずみの回避条件について考察する。

これまで、Deconvolution 層におけるチェス盤ひずみの問題について考えてきたが、Sub-pixel Convolution 層に対しても同様の結論を得ることができる。ただし、Sub-pixel Convolution 層のバイアス $b_n^{(s)}$ は、すべての n で等しくなるとは限らない。したがって、Sub-pixel Convolution 層では、式 (8) の右辺第 1 項の条件 (式 (10)) に加え、第 2 項がすべての n で等しくなる必要がある。

3.2 チェス盤ひずみの回避条件

前述の通り、チェス盤ひずみの問題を解決するためには、Upsampling 層のフィルタ及びバイアスに何らかの制約が必要となる。ここでは、Deconvolution 層及び Sub-pixel Convolution 層においてチェス盤ひずみを回避するための条件を導出する。

Deconvolution 層におけるチェス盤ひずみの回避条件について考察する。図 3 に示すように、Deconvolution 層は、複数のインターポレータから構成されるマルチレートシステムを用いて表現できる。図中のインターポレータ (i) に対応する 2 次元フィルタの係数 $W_{1,1}^{(d)}$ が

$$W_{1,1}^{(d)} = P * NN \quad (11)$$

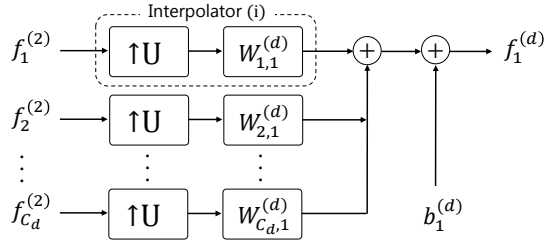


図 3 Deconvolution 層のシステム構成図

と表されるとき, 以下の関係が成り立つことが知られている [13].

$$\overline{W}_{1,1}^{(s)} = \overline{W}_{1,2}^{(s)} = \dots = \overline{W}_{1,U^2}^{(s)} \quad (12)$$

ここで, P は任意の 2 次元信号, NN はすべての値が等しい $U \times U$ の 2 次元信号である.

さらに, Deconvolution 層におけるすべてのインターポレータが式 (11) の条件を満たす場合を考えると, 任意のチャンネル c において, 次式が成り立つ.

$$\overline{W}_{c,1}^{(s)} = \overline{W}_{c,2}^{(s)} = \dots = \overline{W}_{c,U^2}^{(s)} \quad (13)$$

このとき, 式 (8) の右辺第 1 項はすべての n で等しくなり, ひずみは発生しない. 以上から, Deconvolution 層におけるチェス盤ひずみは, 次の条件 1 を満たすことで回避される.

[チェス盤ひずみの回避条件 1]

Deconvolution 層における任意の 2 次元フィルタの係数 $W_{c,n}^{(d)}$ が 2 次元信号 $P_{c,n}$, すべての値が等しい $U \times U$ の 2 次元信号 NN を用いて, 次式のように表される.

$$W_{c,n}^{(d)} = P_{c,n} * NN \quad (14)$$

一方, Sub-pixel Convolution 層では, 式 (8) の右辺第 2 項がすべての n で等しいとは限らない. したがって, Sub-pixel Convolution 層におけるチェス盤ひずみは, 上記の条件 1 に加えて次の条件 2 を満たすことで回避される.

[チェス盤ひずみの回避条件 2]

Sub-pixel Convolution 層における任意のバイアス $b_n^{(s)}$ について以下の関係が成り立つ.

$$b_1^{(s)} = b_2^{(s)} = \dots = b_{U^2}^{(s)} \quad (15)$$

3.3 チェス盤ひずみ回避法

ここでは, 上述のチェス盤ひずみの回避条件を考慮した Deconvolution 層及び Sub-pixel Convolution 層について考察する. Deconvolution 層より生成される超解像画像 $f^{(d)}(I_{LR})$ に対し, 2 次元ステップ信号 NN を畳み込むことで与えられる次式を考えると,

$$\begin{aligned} & NN * f^{(d)}(I_{LR}) \\ &= \sum_{c=1}^{C_d} \uparrow U(f_c^{(2)}(I_{LR})) * NN * W_{c,n}^{(d)} + NN * b_n^{(d)} \end{aligned} \quad (16)$$

となる. このとき, 上式は明らかにチェス盤ひずみの回避条件 1 を満たし, ひずみは発生しない.

また同様に, Sub-pixel Convolution 層より生成される超解像

画像 $f^{(s)}(I_{LR})$ に対し, 2 次元ステップ信号 NN を畳み込むことで与えられる次式を考えると,

$$\begin{aligned} & NN * f^{(s)}(I_{LR}) \\ &= NN * f^{(d)}(I_{LR}) \\ &= \sum_{c=1}^{C_d} \uparrow U(f_c^{(2)}(I_{LR})) * NN * W_{c,n} + NN * b_n \end{aligned} \quad (17)$$

となる. このとき, 上式の右辺第 2 項は, Deconvolution 層のバイアスと同様に一定値をとる. したがって, 上式はチェス盤ひずみの回避条件 1, 2 を満たし, ひずみは回避される.

結論として, 任意の Deconvolution 層及び Sub-pixel Convolution 層におけるチェス盤ひずみは, その出力に 2 次元ステップ信号 NN を畳み込むことで回避することができる. また, このことは Deconvolution 層及び Sub-pixel Convolution 層の直後に, 2 次元ステップ信号 NN を内包する畳み込み層を追加することと等価である. 以降では, これらをチェス盤ひずみ回避法として Deconvolution 層及び Sub-pixel Convolution 層に適用する. さらに, 提案するチェス盤ひずみ回避法は, 以下の 2 つの場合に分けられる.

A. チェス盤ひずみ回避法 (No Training)

提案するチェス盤ひずみ回避法では, その学習の有無に関わらず, ひずみは回避される. したがって, 既に学習済みの CNN が利用可能な場合, 提案法を適用することで再び学習を行うことなく, ひずみを回避することができる. ただし, このとき 2 次元ステップ信号 NN の値は, その直流利得が 1 になるように $\frac{1}{U}$ とする必要がある. しかしながら, このとき使用されるフィルタ及びバイアスは, 提案法で追加される畳み込み層の影響を考慮していない. そのため, この手法では, 最適な出力を得ることはできないと考えられる.

B. チェス盤ひずみ回避法 (Training)

提案するチェス盤ひずみ回避法を適用した CNN に対して, 学習を行うことを考える. このとき, 提案法により追加される畳み込み層の影響を考慮したフィルタ及びバイアスを学習することができる. そのため, この手法では, 前述のチェス盤ひずみ回避法 (No Training) と異なり, 最適な出力を得られることが期待される. また, 2 次元ステップ信号 NN は任意の値を選択可能である. 簡単のため, 本稿では値を $\frac{1}{U}$ として実験を実施する.

4. 実験

提案するチェス盤ひずみ回避法を Deconvolution 層及び Sub-pixel Convolution 層に適用することで, チェス盤ひずみが回避可能であることを確認するため, 超解像による画質評価実験を実施した.

4.1 実験条件

A. 学習・評価データセット

実験では, 91 枚の自然画像で構成されているデータセット 91-image [14] を学習用に準備した. この 91-image に対して縮小・回転によるデータ拡張を施し, 1820 枚からなるデータセット 91-image-Aug を作成した. ここでデータ拡張は [6] と同様

に、縮小率 0.9, 0.8, 0.7, 0.6 の縮小と 90° , 180° , 270° の回転を使用した。また、評価データセットとして Set5 [15], Set14 [16] の 2 つを準備した。

学習データセットを準備するため、91-image-Aug の全ての画像に対して所望の拡大率 2, 3, 4 倍に応じた Bicubic による縮小を行い、低解像度画像を生成する。ただし、データセット中の画像は YCbCr 色空間に変換されその輝度成分 (Y) のみが用いられる。低解像度画像は、拡大率 2, 3, 4 倍に応じて、それぞれ 18×18 , 12×12 , 9×9 画素からなるパッチに分割され、それらが学習データセットの入力データとなる。正解ラベルは、すべての拡大率において原画像から 36×36 画素のパッチとして抽出される。また、総パッチ数は 32,000 とした。

B. ネットワークとハイパーパラメータ

実験では、図 1 に示した 2 層の畳み込み層と 1 層の Upsampling 層から構成されるネットワークに基づいて、その Upsampling 層が、Deconvolution 層または Sub-pixel Convolution 層である場合 (Deconv, Sub-pixel), Deconvolution 層, Sub-pixel Convolution 層それぞれに対して提案するチェス盤ひずみ回避法を適用した場合 (Deconv+NN, Sub-pixel+NN), Resize Convolution 層 [9] である場合 (ResizeConv) の 5 種類のネットワークを評価した。また [7] と同様に、各畳み込み層のフィルタサイズは $K_1 = 5, K_2 = 3$, フィルタ数は $N_1 = 64, N_2 = 32$ とした。Deconvolution 層, Sub-pixel Convolution 層及び Resize Convolution 層のフィルタサイズは、それぞれ $K_d = 6, K_s = 3, K_r = 6$ とした。このとき、すべてのネットワークにおいて総パラメータ数は約 21,184 となる。最適化の対象となる目的関数は、次式で定義される超解像画像 $f(I_{LR})$ と原画像 I_{HR} の平均二乗誤差 $\ell(\theta)$ である。

$$\ell(\theta) = \frac{1}{N} \sum_{i=1}^N \|f(I_{LR}^{(i)}; \theta) - I_{HR}^{(i)}\|^2 \quad (18)$$

ここで、 N は学習に用いられるデータ数、 θ は学習によって最適化される各層のフィルタとバイアスである。

実験で使用したハイパーパラメータについて説明する。学習ではバッチサイズ 128, 学習率 0.0001 として Adam [17] を使用した。ここで、Adam の一次モーメント β_1 の値は 0.9, 二次モーメント β_2 の値は 0.999 である。また、各畳み込み層のフィルタの初期値として He の初期値 [18] を使用した。学習回数は 100 万回である。実装は、Deep Learning のフレームワークの 1 つである Tensorflow を使用した。

C. 評価用フレームワーク

評価方法は、Huang らの公開されているフレームワーク [19] に従う。このフレームワークでは、原画像を Bicubic で縮小し、得られた低解像度画像の輝度成分 (Y) にのみ超解像を適用する。画質評価は、原画像と超解像画像の輝度成分 (Y) のみを用いて PSNR (Peak Signal-to-Noise Ratio), SSIM (Structural SIMilarity) の値が使用される。

4.2 実験結果

図 4 は、解像度が異なる複数の画像に対し、拡大率 2 倍における各 Upsampling 層の処理をそれぞれ 10,000 回繰り返した際の

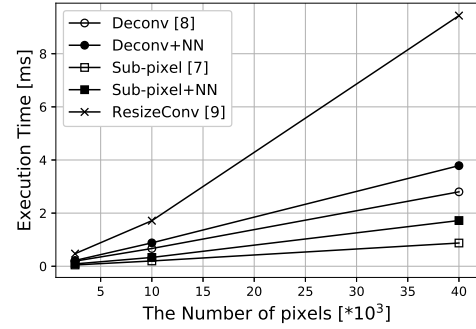


図 4 Execution Time of Upsampling Layers ($U = 2$)

平均実行時間 (ms) を表したグラフである。図より、提案法である Deconv+NN, Sub-pixel+NN は、ResizeConv と比較してより短い時間で実行可能であることがわかる。また、Sub-pixel+NN は、提案法を適用したにもかかわらず、Deconv よりも高速に実行できる。ただし、これらの実行時間は 3.3GHz のプロセッサと 16GB のメインメモリを搭載するコンピュータ上で、Caffe (C++ 実装) により測定された値である。

図 5 は、同図 (a) の "ppt3" に対し、拡大率 3 倍の超解像を施すことで生成される超解像画像を示している。ここで、図中の Deconv+NN, Sub-pixel+NN は、Deconv, Sub-pixel それぞれに対し、提案するチェス盤ひずみ回避法を適用したものである。3.3 で述べたように、提案法の適用は、2 種類 (No Training), (Training) がある。ただし、学習回数は 10 万回である。同図 (b), (f) で生じていたチェス盤ひずみが、同図 (c), (d), (g), (h) では発生していない。

表 1 に、各手法に対する評価データセット Set5, Set14 の PSNR (dB), SSIM の平均値を示す。表から、提案法である Deconv+NN, Sub-pixel+NN は、従来法である Deconv, Sub-pixel が生成する超解像画像の品質を維持したまま、チェス盤ひずみを回避可能であることがわかる。また、提案法は ResizeConv より高速に実行可能であるにもかかわらず、ほぼ同品質の超解像画像を生成することが示された。

5. まとめ

本稿では、アップスケーリング処理を含む CNN において発生するチェス盤ひずみを回避するための手法を提案した。実験では、提案法を Deconvolution 層, Sub-pixel Convolution 層に適用することで、そのチェス盤ひずみが完全に回避可能であることを確認した。また、提案法は、Resize Convolution 層より少ない計算量で実行可能であることが示された。

文 献

- [1] A. Krizhevsky, I. Sutskever and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.
- [2] A. Radford, L. Metz and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," *arXiv preprint arXiv:1511.06434*, 2015.
- [3] J. Long, E. Shelhamer and T. Darrell, "Fully convolutional networks for semantic segmentation," in *IEEE Conf. Comput. Vis. Pattern. Recognit.*, 2015, pp. 3431–3440.
- [4] C. Dong, C. C. Loy, K. He and X. Tang, "Learning a Deep Convolutional Network for Image Super-Resolution,"

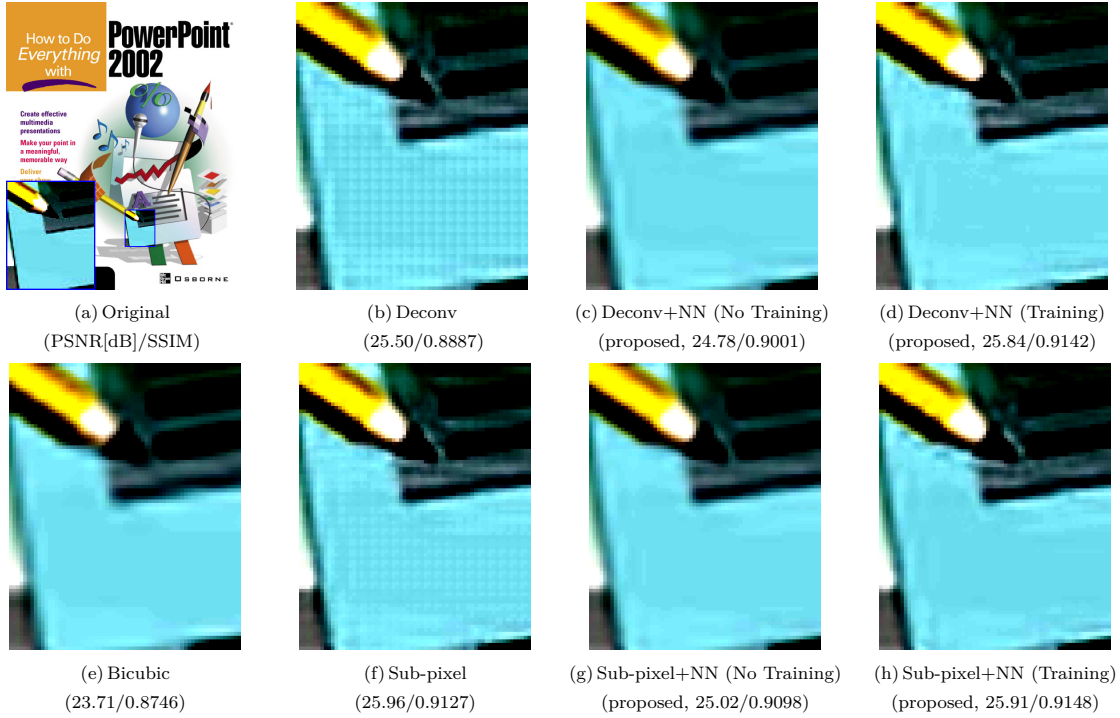


图 5 Examples of Super-Resolved images ($U = 3$)

表 1 Experimental Results (PSNR[dB], SSIM)

test dataset	upsampling factor	Bicubic PSNR/SSIM	Deconv PSNR/SSIM	Deconv+NN (Training) PSNR/SSIM	Sub-pixel PSNR/SSIM	Sub-pixel+NN (Training) PSNR/SSIM	ResizeConv PSNR/SSIM
Set5	×2	33.66/0.9299	36.76/0.9549	36.81/0.9552	36.79/0.9550	36.81/0.9550	36.83/0.9551
	×3	30.39/0.8682	32.57/0.9065	32.63/0.9065	32.73/0.9084	32.67/0.9075	32.67/0.9075
	×4	28.42/0.8104	30.29/0.8575	30.23/0.8520	30.40/0.8589	30.32/0.8556	30.24/0.8525
Set14	×2	30.23/0.8687	32.41/0.9068	32.46/0.9069	32.46/0.9071	32.44/0.9072	32.45/0.9070
	×3	27.54/0.7736	29.12/0.8174	29.16/0.8185	29.22/0.8192	29.16/0.8184	29.17/0.8185
	×4	26.00/0.7019	27.36/0.7479	27.30/0.7444	27.40/0.7486	27.36/0.7473	27.29/0.7450

in *Proc. Eur. Conf. Comput. Vis.*, Springer, 2014, pp. 184–199.

- [5] Y. Sugawara, S. Shiota, and H. Kiya, “A parallel computation algorithm for super-resolution methods using convolutional neural networks,” in *Proc. APSIPA Annual Summit and Conference*, 2017.
- [6] C. Dong, C. C. Loy and X. Tang, “Accelerating the super-resolution convolutional neural network,” in *Proc. Eur. Conf. Comput. Vis.*, Springer, 2016, pp. 391–407.
- [7] 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,” in *IEEE Conf. Comput. Vis. Pattern. Recognit.*, 2016, pp. 1874–1883.
- [8] M. D. Zeiler, G. W. Taylor and R. Fergus, “Adaptive deconvolutional networks for mid and high level feature learning,” in *Proc. IEEE Inter. Conf. Comput. Vis.*, 2011, pp. 2018–2025.
- [9] A. Odena, V. Dumoulin and C. Olah, “Deconvolution and checkerboard artifacts,” *Distill*, 2016. [Online]. Available: <http://distill.pub/2016/deconv-checkerboard>
- [10] A. P. 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 preprint arXiv:1707.02937*, 2017.
- [11] I. W. Selesnick, R. G. Baraniuk and N. C. Kingsbury, “The dual-tree complex wavelet transform,” *IEEE Signal Processing Magazine*, vol. 22, no. 6, pp. 123–151, 2005.
- [12] T. Tamura, M. Kato, T. Yoshida and A. Nishihara, “Design of checkerboard-distortion-free multidimensional multirate filters,” *IEICE Trans. Fundamentals*, vol. E81-A, no. 8, pp. 1598–1606, 1998.
- [13] Y. Harada, S. Muramatu and H. Kiya, “Multidimensional multirate filter and filter bank without checkerboard effect,” *IEICE Trans. Fundamentals*, vol. E81-A, no. 8, pp. 1607–1615, 1998.
- [14] J. Yang, J. Wright, T. S. Huang and Y. Ma, “Image super-resolution via sparse representation,” *IEEE Trans. Image Processing*, vol. 19, no. 11, pp. 2861–2873, 2010.
- [15] M. Bevilacqua, A. Roumy, C. Guillemot and M. L. Alberi-Morel, “Low-complexity single-image super-resolution based on nonnegative neighbor embedding,” in *Proc. Brit. Mach. Vis. Conf.*, 2012.
- [16] Zeyde, Roman and Elad, Michael and Protter, Matan, “On single image scale-up using sparse-representations,” in *Proc. Curves and Surfaces*, 2010, pp. 711–730.
- [17] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” in *Proc. Inter. Conf. on Learning Representations*, 2015.
- [18] K. He, X. Zhang, S. Ren and J. Sun, “Delving deep into rectifiers: Surpassing human-level performance on imagenet classification,” in *Proc. IEEE Inter. Conf. Comput. Vis.*, 2015, pp. 1026–1034.
- [19] J.-B. Huang, A. Singh and N. Ahuja, “Single image super-resolution from transformed self-exemplars,” in *IEEE Conf. Comput. Vis. Pattern. Recognit.*, 2015, pp. 5197–5206.