C-001

Incremental Network Quantization の適用による CNN モデル MobileOne の軽量化
Lightweight CNN Model Based on MobileOne by Applying Incremental Network Quantization

横田 俊介† 黒木 修隆† 沼 昌宏†
Shunsuke Yokota Nobutaka Kuroki Masahiro Numa

1. はじめに

近年，画像認識分野において畳み込みニューラルネットワーク（CNN：Convolutional Neural Network）による深層学習が注目を集めている。CNN の処理には膨大な計算量が必要であり，その処理を高速化するアクセラレータとして GPU（Graphics Processing Unit）が一般的に用いられてきたが，消費電力が大きいという問題がある。そこで，汎用性に優れながら GPU より低い消費電力で動作可能な FPGA（Field-Programmable Gate Array）上に CNN 専用の回路を実装することにより，低消費電力化を実現する研究が注目されている。しかし，FPGA の回路規模と利用可能なリソース数には制限があるため，FPGA に実装することを考慮した演算手法や回路構成が必要となる。

CNN モデル MobileOne [1] は MobileNet-V1 をベースとしたモデルであり，空間方向の畳み込み（Depthwise Convolution）とチャネル方向の畳み込み（Pointwise Convolution）を分割した Separable Convolution を用いることによる軽量化を行っている。また，学習時と推論時の構造が異なり，学習時には複数ブロックによる畳み込み演算によって，CNN モデルの表現力を向上させることができる。推論時には，学習時に獲得したパラメータを Reparameterize によって等価変換し，軽量なモデルで推論を行うことができる。そのため，この MobileOne に軽量化を施すことによって，認識精度の低下を小さくする手法が考えられる。

そこで本稿では，CNN モデルとして MobileOne を導入するとともに，インクリメンタル量子化手法 INQ（Incremental Network Quantization） [2] を適用することによって軽量化を施す手法を提案する。

2. 提案手法

2.1 INQ

インクリメンタル量子化手法 INQ とは，32 bit 浮動小数点数で表現された学習済み CNN の重みを，指定したビット幅の 2 のべき乗に量子化する手法である。重みをすべて 2 のべき乗に変換することによって，ハードウェア上の乗算をシフト演算に置き換えることが可能となり，FPGA 上のリソース数を削減できる。図 1 に INQ 量子化手法の概要を示す。まず，全体の 50% の重みを 2 のべき乗に量子化して固定する。この時，絶対値の大きい重みから量子化を行う。その後，再度学習を行うことによって，量子化していない重みを更新し，残りの 50% の重みを量子化する。以上の処理を，すべての重みが量子化されるまで繰り返す。量子化の間に学習をはさむことによって，精度低下を抑えつつ，軽量化を実現することができる。

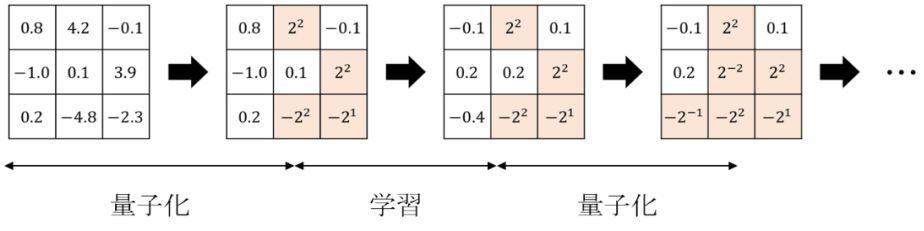

図 1 INQ 量子化手法の概要

2.2 提案ネットワーク

図 2 に提案するネットワークの構造を示す。CNN モデルとして MobileOne を導入して，精度向上を図るとともに，演算に必要なリソース数の削減を目的として，INQ を適用

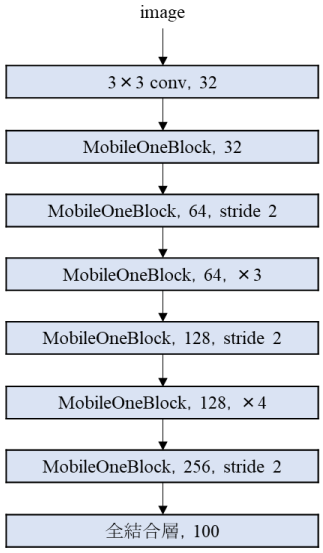

図 2 提案ネットワークの構造

† 神戸大学，Kobe University

する。軽量化と精度低下の兼ね合いを考慮して，INQ による量子化後のビット幅を 5 bit とした。

ハードウェア上では，INQ の適用により乗算をシフト演算に置き換えることによって，演算に要するリソース数を削減した。また，さらなるリソース数の削減を図るため，演算に用いる数値表現を浮動小数点数から固定小数点数に変更した。固定小数点数のビット幅については，オーバーフローを考慮して整数部 16 bit，小数部 16 bit とした。

3. 実験と評価

3.1 ソフトウェアでの CNN 学習

以下の 4 つのモデルを用いて，CIFAR-100 データセットに対する認識精度とパラメータ容量の比較を行った。

a) MobileNet-V1
b) MobileOne
c) MobileNet-V1 に INQ を適用したモデル
d) MobileOne に INQ を適用したモデル（提案手法）

これら 4 つのモデルの推論時の構造はすべて同一である。

表 1 に認識精度とパラメータ容量の比較結果を示す。a) と c) の比較では，INQ 適用によって精度が 4 pt 以上低下した一方で，パラメータ容量が約 84% 削減された。また，d) は b) と比べると精度が 1.13 pt 低下するものの，a) よりは 0.27 pt 高い精度を維持しつつ，パラメータ容量を約 84% 削減できることを確認した。

また，MobileOne に INQ を適用したモデルと比べて，MobileNet-V1 に INQ を適用したモデルで認識精度の低下が顕著である要因としては，MobileNet-V1 は MobileOne と比較して学習時の演算量が少ないことが挙げられる。学習時の演算量が少ない CNN に対する軽量化は，推論に必要な部分を削ることになる結果，大幅な精度低下を引き起こす場合がある。そのため，MobileNet-V1 に INQ を適用した際に大幅な精度低下が生じたと考えられる。

3.2 ハードウェア上でのリソースの比較

図 3 にハードウェア上に実装した回路の構造を示す。INQ を適用していない従来手法と提案手法で採用した INQ を用いた手法の 2 つの手法で，入出力 32 チャネルの Separable Convolution 層を，評価キットである ZCU102 が搭載する FPGA 上にマッピングして，リソース数の比較を行った。従来手法は，32 bit 浮動小数点数で演算を行い，乗算を一般的な乗算回路で実現した。一方で提案手法については，整数部と，小数部ともに 16 bit の固定小数点での演算を行い，乗算をシフト演算で実現した。

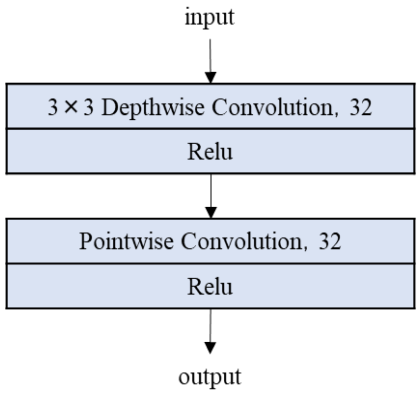

図 3　実装した回路の構造

表 1　CIFAR100 の認識精度

CNN の構成	認識精度	パラメータ容量 [bit]
a) MobileNet-V1	62.92%	5,095,552
b) MobileOne	64.32%	5,095,552
c) MobileNet-V1 + INQ	58.54%	796,180
d) 提案手法	63.19%	796,180

表 2　Separable Convolution 層のマッピング結果

リソース	従来手法（利用率 [%]）	提案手法（利用率 [%]）
LUT	362,930（132.41）	218,964（78.89）
LUTRAM	43,648　（30.31）	0　（0.00）
FF	433,702　（79.12）	137,951（25.17）
DSP	2,520（100.00）	0　（0.00）
BRAM	172　（18.86）	172（18.86）

表 2 にマッピング結果を示す。乗算にシフト演算を用いた場合，従来と比較して LUT を約 40%，FF を約 68% 削減し，LUTRAM と DSP を用いることなく実装できることを確認した。特に LUT について，従来手法では利用可能数を超えていたのに対して，提案手法では FPGA で実現可能なリソース数に抑えられ，図 3 の回路を FPGA 上に実装できることを確認した。しかし，このままの構造で図 2 のネットワーク全体の実装を行うためには，およそ 10 数倍のリソースが必要となるため，ハードウェア化に適したアーキテクチャに関する，さらなる検討が必要である。

4. おわりに

本稿では，FPGA 実装に向けた CNN モデルの軽量化と精度維持を目的として，高精度モデル MobileOne を導入し，畳み込み演算に用いられる重みを 2 のべき乗に量子化するインクリメンタル量子化手法 INQ を適用したモデルを提案した。INQ の適用によって乗算をシフト演算に置き換えることでハードウェア上での演算に用いられるリソース数を削減する一方で，量子化を行った際に生じる精度低下抑制ために，Reparameterize によって表現力を向上させる MobileOne を導入した。

評価実験の結果，INQ を適用していない MobileNet-V1 と比較して，提案手法は認識精度を維持しつつパラメータ容量を約 84% 削減できることを確認した。また，MobileNet-V1 に INQ を適用したモデルと提案手法との比較から，INQ を適用した場合でも MobileOne の Reparameterize は有用であることを確認した。

今後の課題として，MobileOne 全体の実装や，他のネットワークへの MobileOne の適用が挙げられる。

参考文献

[1] P. K. A. Vasu, J. Gabriel, J. Zhu, O. Tuzel, and A. Ranjan, “MobileOne: An improved one millisecond mobile backbone,” arXiv preprint arXiv; 2206.04040v2, 2023.

[2] A. Zhou, A. Yao, Y. Guo, L. Xu, and Y. Chen, “Incremental network quantization: Towards lossless CNNs with low-precision weights,” arXiv preprint arXiv: 1702.03044v2, 2017.