

CondConv を用いた GhostNet

日大生産工 ○鈴木 勇輝 日大生産工 山内 ゆかり

1. まえがき

スマートフォンや自動車など、組み込み機器への畳み込みニューラルネットワーク (CNN) の導入は、メモリと計算リソースが限られているため難しい。そこでHanらは、ResNet-50[1]などが持つ特徴マップの冗長性に注目し、少数の特徴マップに単純な線形変換を施し特徴マップを生成することで、より多くの特徴マップを生成する新しい手法であるGhost moduleおよびGhostNet[2]を提案した。このGhost moduleを導入した既存モデルは、そのモデルの精度を維持しつつも2倍以上の速度で推論させることに成功しており、精度と速度のトレードオフを高い位置で成立させた。

本研究では、Conditionally Parameterized Convolutions[3]を導入しGhostNetの畳み込み部分に改変を行うことで、通常のGhostNetの精度の向上を目指す。

2. 従来研究

2.1 CNN

CNN(畳み込みニューラルネットワーク)とは、畳み込み層、プーリング層、全結合層で構成される、主に画像認識の分野において用いられるディープラーニングアルゴリズムである。畳み込み層で局所的な特徴量の抽出を行い、プーリング層であり重要ではない情報を削ぎ落とすことで画像データを小さく纏め、全結合層では畳み込み層とプーリング層で得られた特徴量から最終的な予測や分類の結果を出力するという流れで成り立っている。

2.2 GhostNet

GhostNetは、Ghost moduleを用いることで既存のCNNモデルよりも効率的に特徴マップを生成する、非常に軽量のモデルである。Ghost moduleでは、通常の畳み込み層を2段階に分けて行う。このGhost module単体を導入するだけでも、精度を維持しつつもモデルの大幅な軽量化と高速化が期待できるというメリットがある。具体的に図1を用いて説明を行う。

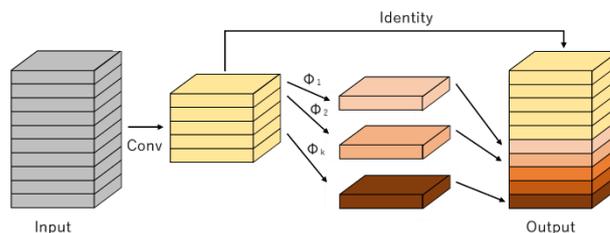


図 1. Ghost module

始めに通常の畳み込みを行い、 m 個の特徴マップが生成される。次にそこから得られた m 個の特徴マップに単純な線形演算(Φ)を行うことで、 s 個の特徴マップが生成される。得られた2つの特徴マップを合わせ、最終的に $n(=s \cdot m)$ 個の特徴マップが得られる。このGhost moduleを用いたGhostNetは従来のCNNモデルと比べて、高速化率と圧縮率を s 倍にする効果がある。(1)に理論的な高速化率、(2)に圧縮率を示す。

$$r_s = \frac{c \cdot k \cdot k}{\frac{1}{s} \cdot c \cdot k \cdot k + \frac{s-1}{s} \cdot d \cdot d} = \frac{s \cdot c}{s + c - 1} \approx s \quad (1)$$

$$r_c = \frac{n \cdot c \cdot k \cdot k}{\frac{n}{s} \cdot c \cdot k \cdot k + (s-1) \cdot d \cdot d} = \frac{s \cdot c}{s + c - 1} \approx s \quad (2)$$

Ghost moduleの利点を生かすために設計されたGhost bottleneck (G-bneck)について、図2を用いて概要を説明する。

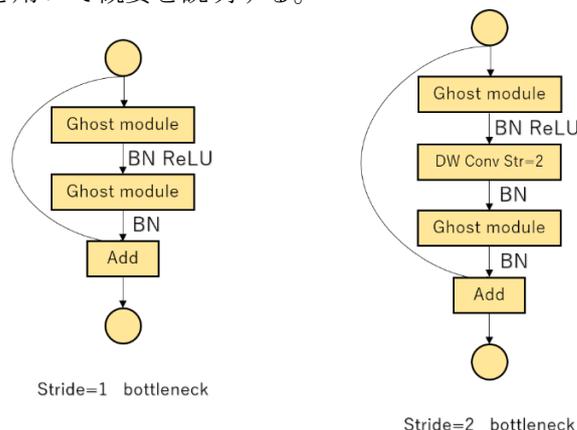


図 2. Ghost bottleneck

図2が示すように、Ghost bottleneckはResNetの残差ブロックに似ており、複数の畳み

込み層(Ghost module)とショートカットが統合されている。Ghost bottleneckは、主に2つの積み重ねられたGhost moduleから構成され、最初のGhost moduleはチャンネル数を増やす拡張層として機能する。2番目のGhost moduleは、ショートカットの経路に合わせてチャンネル数を減らす機能を持つ。そして、ショートカットはこれら2つのGhost moduleの入力と出力の間に接続される。Stride=2のGhost bottleneckでは、Ghost moduleの間にDepthwise convolutionを挿入する。バッチ正規化 (BN) とReLU関数は各層の後に適用されるが、2番目のGhost moduleの後にReLU関数は使用されない。

GhostNet全体の構成は、基本的にはMobileNetV3[4]のアーキテクチャを踏襲しており、MobileNetV3のボトルネックブロックをGhost bottleneckに置き換える形で構築される。

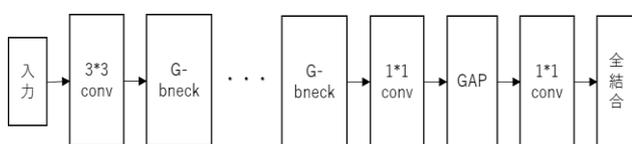


図 2. GhostNet

最初のレイヤーは16のフィルターを持つ標準的な畳み込みレイヤーで、その後、徐々にチャンネルを増やした一連のGhost bottleneckが続く。基本的にGhost bottleneckは図2のStride=1のものを用いるが、各ステージ最後のGhost bottleneckではStride=2のmoduleを使う。最後に、特徴マップを特徴ベクトルに変換するためにGlobal Average Pooling(GAP)と畳み込み層が利用される。

2.3 Cond Conv

Conditionally Parameterized Convolutions (Cond Conv)は、保持させた複数の畳み込みのパラメータWに、入力に応じて変化する混合係数 α を求め、それらから新たなパラメータWを求めるといった流れで行われる。

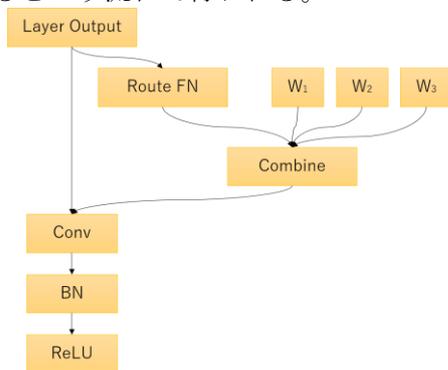


図 3. CondConv

単一ネットワークよりも汎用的で高度な推論ができるようにする、畳み込みの手法である。

僅かな計算量の増加で、精度の向上が確認されている。

3. 提案手法

GhostNetは非常に軽量のモデルであり、Ghost moduleも高速化とモデルの圧縮率は向上されるものの、精度の向上はそこまで見込めない。そこで、GhostNetで用いられていたpointwise convolutionの代わりに、Cond Convを導入し、計算量の削減と精度の向上の両立を目指す。

4. まとめ

Cond Convを使用するとわずかに計算量が増加してしまうものの、GhostNetの高速化、軽量化の性能は既存のCNNモデルを大きく上回っているため、計算量の削減と精度の向上の両立を図ることができるのではないと思われる。

参考文献

- 1) Kaiming He, Xiangyu Zhang, Shaoqing Ren and Jian Sun, "Deep Residual Learning for Image Recognition", arXiv preprint arXiv:1512.03385, (2015)
- 2) Kai Han, Yunhe Wang, Qi Tian, Jianyuan Guo, Chunjing Xu, Chang Xu, "GhostNet: More Features from Cheap Operations", arXiv preprint arXiv:1911.11907, (2019)
- 3) Brandon Yang, Gabriel Bender, Quoc V. Le, Jiquan Ngiam, "CondConv: Conditionally Parameterized Convolutions for Efficient Inference", arXiv preprint arXiv:1904.04971v1, (2019)
- 4) Andrew Howard, Mark Sandler, Grace Chu, Liang-Chieh Chen, Bo Chen, Mingxing Tan, Weijun Wang, Yukun Zhu, Ruoming Pang, Vijay Vasudevan, Quoc V. Le, Hartwig Adam, "Searching for mobilenetv3.", arXiv preprint arXiv:1905.02244, (2019)