C-009

SoCFPGA による深層学習モデル「RegNet」高速化手法の検討
Investigation of Acceleration Method for Deep Learning Model "RegNet" Using SoCFPGA

田嶋 夏己† 中西 知嘉子‡
Natsuki Tajima Chikako Nakanishi

1. はじめに

近年，AI 技術が注目されている．現在はクラウド上の高性能端末にデータを送信し，推論処理を行うクラウド AI が主流となっている．しかし，クラウド AI は膨大なデータ通信が必要な点，それに伴うセキュリティへの懸念など通信面での問題点がある．これに対し，エッジ端末上で完結した推論を行うエッジ AI は，セキュリティ面への懸念も少なく，通信環境に左右されない安定した推論が可能である．しかし，エッジ端末は性能面で大きく劣っており，膨大な演算が必要である AI を高速に動作させる事は難しいという問題点がある．

本研究では，エッジ端末として SoC FPGA ボードである Ultra96-V2[1]を用いる．SoC FPGA の特徴である回路部とソフト部の協調動作によって精度を維持したまま推論処理の高速化を行う手法を提案する．

2. AI の分析

本研究では，本研究では，Ultra96-v2 上に実装する画像認識 AI モデルとして，RegNet[2]を使用する．

RegNet は 2020 年に発表されたシンプルなネットワーク構造を持つ高速かつ高精度な深層学習モデルである．今回は最も小さいモデルである RegNetX-200MF を採用した．

2.1 回路化処理の決定

FPGA 上に実装する処理を決定するために RegNetX-200MF を Ultra96-V2 上の CPU のみで実行した場合の各層の処理時間を計測した．計測には Ceras[3]を用いた．Ceras(C++ edge rapid ai simulator)とは C++言語を用いて，Keras，ONNX の学習済みモデルの推論を行うことが出来るライブラリである．計測結果を表 1 に示す．

表 1　層ごとの処理時間と割合

層の種類	処理時間	割合
Conv2D	11494.2ms	95.0%
GroupConv2D	500.5ms	4.1%
その他	109.9ms	0.9%

表 1 より，畳み込み演算を行う層である，Conv2D 層，GroupConv2D 層が処理時間の約 99%を占めている．

Conv2D 層ではカーネルサイズ 1 のストライド 1 と 2，カーネルサイズ 3 のストライド 2 の 3 種類の畳み込み演算が行われている．また，GroupConv2D 層ではカーネルサイズ 3 のストライド 1 と 2 が行われている．そのため，この 4 つの畳み込み演算を回路化することとした．

3. 回路設計

本研究では，C++言語を用いてアルゴリズムを設計し，高位合成によって回路設計を行った．高位合成を行うツールとして Vivado HLS 2019.2，高位合成から得られた RTL を回路に組み込むツールとして Vivado 2019.2 を用いた．

3.1 ベース回路

本研究では，ベース回路として，RegNet に存在するカーネルサイズ 1 のストライド 2 以外の 3 種類の畳み込み演算に対応した，先行研究[4]で作成された汎用畳み込み演算回路を使用した．

先行研究[4]の回路は乗算を 9 並列に行い，乗算結果を加算する処理を 8 並列で行う回路である．そのため，カーネルサイズ 3 は 8 並列，カーネルサイズ 1 は 72 並列で行える．

RegNetX-200MF は 32bit の float 型で処理が行われているため，CPU と回路間のデータパス幅は 32bit とした．また，PL 部に供給するクロック周波数は 300MHz とした．回路部へのデータ転送のためのバッファとして共有メモリを，データ転送には DMA 転送を使用する．

この回路では，FPGA のメモリリソース量から処理できる畳み込み演算のサイズに上限を設定している．上限サイズを表 2 に示す．

表 2　回路で処理できる上限サイズ

	上限サイズ
入力データサイズ	66x66
カーネルデータ	152
カーネルデータ，入力データのチャネル	64(カーネルサイズ 1 の場合は 576)

先行研究[4]の結果から，共有メモリの読み書き時間が推論処理全体の約 26%を占めており，出力データの読み込み時間，入力データの書き込み時間の削減の余地がある．その削減手法のための設計を 5 章，6 章で行う．

4. ソフトウェア設計

ソフトウェア部 AI 処理には Ceras[3]を用いた． Ceras は C++言語の標準ライブラリのみで動作可能であり，環境への追従性が高い．また，AI を C++言語で推論させることでソフト部と回路部の協調動作を容易にした．

Conv2D 層，GroupConv2D 層の畳み込み演算を行う際には，必要なデータを共有メモリに書き込む処理，回路部の演算結果を取り出し，整形する処理を行う．

4.1 データ分割・整形

Conv2D 層，GroupConv2D 層のサイズによっては，回路が対応可能なサイズを超える層がある．そのため，サイズを超えた Conv2D 層を処理する場合はデータを分割して回路で処理を行う．GroupConv2D 層の場合は Conv2D 層の処理をグループ回繰り返すことで対応した．

カーネルサイズ 1 のストライド 2 に関しては，入力データの内，必要データのみを共有メモリに書き込み回路部で

カーネルサイズ 1 のストライド 1 として処理することで対応した．

4.2 データ転送のオフセット化

カーネルやバイアスなどの変化しないデータをモデルの読み込み時に共有メモリへ書き込み，入力データは層の処理時にすべて共有メモリに書き込むようにした．データ転送の際には共有メモリのアドレスをオフセット値によって指定し，転送する．

5. GroupConv2D 層の処理方法の変更

これまでは 1 グループずつ回路を使用して処理を行っていた．RegNet では 1 グループ 8 カーネル 8 チャネルなので回路で処理できるチャネル上限を 8 で割ったグループ数を 1 度に処理できるように変更した．

複数のグループを回路で処理するために，1 つの縦横座標内でグループを変更して処理を行うようにした．これにより出力が縦横座標ごとにすべてのカーネルの順で出力される．図 1 に出力順のイメージを示す．

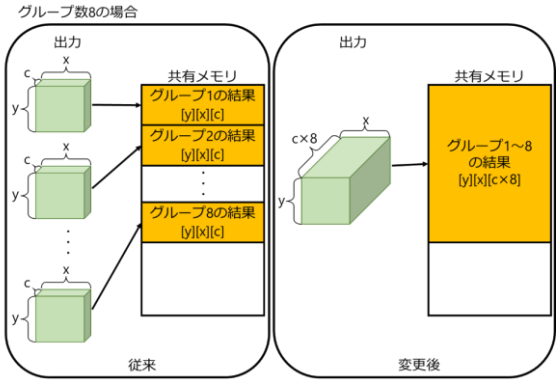

図 1 変更前と変更後の出力順

この変更により，処理時間が増加する．また，タイミング制約の関係から畳み込み演算の並列度を 8 から 4 に変更した．これらの変更は今後適用する出力データの再利用による共有メモリの読み書き時間の削減手法のためであり，この手法による削減時間の方が大きいと考えたため変更した．

6. 回路で処理できる畳み込み演算上限の拡張

回路内のメモリで保持するパラメータ量の上限を拡張することで処理できる畳み込み演算のサイズ上限の拡張を行った．拡張後の回路で処理できる畳み込み演算サイズを表 3 に示す．この回路ではサイクリック分割によってカーネル毎に BRAM が割り振られている．BRAM1 つの容量は 36kbit である．そのため，カーネル 1 チャネルにつき 32bit のデータが 9 つ，計 125 チャネルが BRAM1 つに入る上限となる．また，RegNet のカーネル数は 8 の倍数のため 8 の倍数とした．上限値と 8 の倍数でどこまでチャネルが 1 つの BRAM に入るか検討した結果，112 となったので 112 に設定した．また，RegNet は 224x224 の入力データのモデルであることからパディングサイズを含めた 226x226 を処理できるように変更した．これにより入力データの縦横サイズの分割がなくなる．カーネル数上限に関しては今後適応する出力データの再利用による共有メモリの読み書き時間の削減手法のためにカーネル数とチャネル数を同数にする必要があったため，112 とした．

表 3　変更後の回路で処理できる上限サイズ

	上限サイズ
入力データサイズ	226x226
カーネルデータ	112
カーネルデータ，入力データのチャネル	112(カーネルサイズ 1 の場合は 1008)

7. 評価方法と検証

評価は Ultra96-V2 上で行い，CPU のみで推論を行った場合と回路を使用して推論した場合の処理時間の比較で行った．OS は Ultra96-V2 向け Debian GNU/Linux を用いた．

Ultra96-V2 上で RegNetX-200MF を CPU のみ，ベース回路と CPU，5 章，6 章の変更内容を加えた回路と CPU を使用してそれぞれ推論した場合の処理時間を表 4 に示す．

表 4　処理時間

	CPU のみ	ベース回路	変更後
推論	12104.6ms	587.0ms	618.7ms
Conv2D	11494.2ms	296.9ms	327.8ms
GroupConv2D	500.5ms	174.5ms	185.0ms

表 4 より，推論処理全体を約 19.5 倍，Conv2D 層を約 35.1 倍，GroupConv2D 層を約 2.7 倍の高速化となった．しかしベース回路と比べて推論処理が 31.7ms，Conv2D 層が 30.9ms，GroupConv2D 層が 10.5ms 処理時間が長くなった．

8. おわりに

本研究では，エッジ端末である Ultra96-V2 上で FPGA による回路と CPU による協調動作によって RegNet-200MF の推論の高速化を行った．CNN の中から処理時間が長く，繰り返す回数が多い処理を回路化し，共有メモリへの書き込み時間などの削減を行ったことで推論時間を 19.5 倍高速にすることが出来た．

GroupConv 層の処理を変更することで，ベース回路と比べて 31.7ms 秒処理時間が長くなったが，これについては今後出力データの再利用による共有メモリの読み書き時間の削減手法を適用すること推論処理の約 26%を占める共有メモリ読み書き時間を 8 割ほど減少させることができると考えられる．その結果，ベース回路と比べても高速に動作すると思われる．

参考文献

[1] https://japan.xilinx.com/products/boards-and-kits/1-vad4rl.html
[2] Ilija Radosavovic, Raj Prateek Kosaraju, Ross Girshick, Kaiming He, Piotr Dollár,” Designing Network Design Spaces” arXiv preprint arXiv:2003.13678(2020).
[3] 西岡駿, 中西知嘉子, "機械学習ライブラリの C 言語化の実現", 電子情報通信学会ソサイエティ大会(2021).
[4] 田嶋夏己, 中西知嘉子, " SoC FPGA による「RegNet」高速化手法の検討",リコンフィギャラブルシステム研究会(2023).

† 大阪工業大学 報科学研究科 報科学専攻 Graduate School of Information Science and Technology Osaka Institute of Technology
‡ 大阪工業大学 情報科学部 情報知能学科 Department of Information and Computer Science Osaka Institute of Technology