C-011

量子化推論におけるスケール整合を支援するハードウェア向け補助ツールの開発

Development of a Hardware-Assisted Support Tool for Scale Alignment in Quantized Inference

石見 蒼汰[†]
Sota Ishimi

中西知嘉子[‡]
Chikako Nakanishi

1. はじめに

1.1 研究背景と目的

近年、エッジ AI において、限られたリソース環境下での高速かつ低消費電力な推論処理が求められている。その一つとして、SoC FPGA を活用したハードウェアアクセラレーションが注目されている。中でも、演算量とメモリ使用量を削減可能な量子化手法は、ハードウェアでの実装において、非常に有効である。しかし、量子化されたモデルでは各層でスケール値が異なることが多く、スケールの不一致によって演算誤差や精度劣化が生じるという課題がある。また、量子化・逆量子化のタイミングやビット幅の選定も推論精度や処理効率に大きな影響を与える。

本研究では、ネットワーク全体の構造を解析し、各層間でスケール値が異なる場合に、前の層の出力スケール値と次の層の入力スケール値の差分に基づいて、前層の出力値を調整する処理を導入することで、整数演算による安定した推論処理を実現することを目的とする。あわせて、量子化パラメータの自動調整やメモリ使用の効率化を支援する補助ツールの開発に取り組む。

1.2 課題

従来の推論モデルはすべて float32(浮動小数点型)によって構成されており、演算精度の面では優れている一方で、エッジデバイス上での実行には演算負荷やメモリ使用量が大きいという課題がある。

2. 仕様モデル

2.1 Ceras[1]

Ceras(C++ Edge Rapid AI Simulator)は、本研究室で開発された C++ベースの AI推論ライブラリである。学習済みの深層学習モデルを独自の中間形式(Ceras 形式)に変換することで、標準的な C++ライブラリのみを用いた軽量な推論処理が可能となる。これにより、組み込み機器やリソース制約のあるエッジデバイス上でも、効率的かつ移植性の高い推論環境を実現できる。

本研究では、Ceras を用い、モデル構造の明示的な制御と、FPGA との高い親和性を活かして、エッジデバイス上での最適な推論処理を構築している。

2.2 ONNX

ONNX(Open Neural Network Exchange）は、機械学習モデルを共通の形式で表現するために策定されたオープンフォーマットである。学習済みのモデルは、構築時のフレームワークやライブラリに依存することが多いが、ONNX 形式に変換することで、これらの依存性を吸収し、モデルの再利用性と互換性を高めることができる。本研究では、学習済みの物体検出モデルを一旦 ONNX 形式に変換し、その後 Ceras 形式へと変換して、エッジデバイス上での推論実行に適した形式へと変換している。

2.3 YOLOX

YOLOX は物体検出 AI、YOLO シリーズより、2021 年に発表されたモデルである。YOLO シリーズは、他の物体検出モデルと比較し、推論速度が高速なことが特徴として挙げられる。本研究では、リソースの限られたエッジデバイス上での推論実行を前提とし、量子化モデルに適した YOLOX-Nano の deploy 版を採用した。

2.4 Vitis AI

Vitis AI は、Xilinx 社が提供するエッジ AI 向けの開発プラットフォームであり、FPGA や SoC 上での効率的な AI 推論を支援するツールを備えている。量子化ツール「AI Quantizer」や最適化済モデルを集めた「Model Zoo」により、ハードウェアの知識がなくても高速かつ省電力な AI アプリケーションの開発が可能となり、開発期間の短縮が期待されている。

3. 提案手法

本研究では、エッジAI向けの高効率な推論処理の実現を目的として、以下の 3 点に基づく手法を提案する。 これらは、学習済みモデルの Ceras 形式への変換時に行う。

・モデルの量子化による演算効率の向上
・層間スケールの整合による出力調整
・層の生存期間に基づく共有メモリの最適割り当て

3.1 モデルの量子化による演算効率の向上

深層学習モデルの演算を int8 に量子化することで、演算コストとメモリ使用量を削減し、効率的な推論を可能にする。

本手法では、主に Conv2D 層 に対して量子化を適用し、推論処理の高速化とメモリ使用量の削減を実現する。具体的には、学習済みモデルの重み（weight）を、それぞれのスケール値に基づいて int8 に変換し、効率的な整数演算を可能とする。これらの 量子化スケール値(scale)およびゼロ点(zero-point）は、Vitis-AI の AI-Quantizer を用いて、実際の入力データに対するキャリブレーションを通じて取得している。この手法により、量子化に伴う精度劣化を抑えつつ、各層に最適なスケールパラメータを導出している。

また、バイアス項（bias）は、Conv2D 演算における積和演算の結果に加算される前の段階で扱われるため、量子化誤差の蓄積を防ぐ必要がある。本手法では、演算精度の劣化を最小限に抑えるため、これを 32 ビット整数（int32）で量子化している。量子化後のモデルは、Ceras ライブラリ上で実行される。Ceras は、8 ビット量子化された入力や重みを用いた演算をすべて整数ベースで処理できるように設計されており、エッジデバイス上での効率的な推論を支

える。さらに、層の種類や特性に応じて、精度劣化の影響が大きい一部の層は浮動小数点演算(float)で実行するように指定できる。どの層を浮動小数点で処理するかはモデル情報として事前に埋め込む仕様になっている。次の層が浮動小数点の場合は、前の層の出力に対して逆量子化(dequantization）を行う必要があり、その処理に必要な情報もモデルファイル内に付加されている。

このように、Vitis-AI による量子化と、Ceras による最適化された整数演算実行環境を組み合わせることで、演算精度と実行効率の両立を図った量子化推論が可能となる。

3.2 層間スケールの整合による出力調整

量子化モデルにおいては、各演算層が独立したスケール値（scale）を持つため、層間でスケールが一致しない場合が発生する。このため、各層の出力を次の層で適切に扱えるようにスケールを調整する処理が必要になる。一般的には、各演算の前後で再量子化やスケール変換が行われるが、これらは実行時の演算負荷や処理の非効率性を招く要因となる。 本研究では、モデル構造をあらかじめ静的に解析し、各層の出力スケールを次の層の期待するスケールに揃えるための情報をモデルに付加する。この処理はモデルの Ceras 形式への変換時に行い、各層の出力スケールと次の層の入力スケールの差分から適切な変換係数（主にビットシフト量）を計算し、情報として埋め込む。

3.3 層の生存期間に基づく共有メモリの最適割り当て

本手法では、Ceras ライブラリによる推論実行において、FPGA 上のアクセラレータを活用する構成を想定している。アクセラレータとのデータ授受には、SoC FPGA 内部に配置された高速な共有メモリが使用される。この共有メモリは、アクセラレータと CPU 間での中間データの受け渡しに用いられる重要なリソースである。限られたメモリ資源を効率的に利用するため、本手法ではこの共有メモリを複数のブロックに分割し、各演算層が使用するブロックをあらかじめモデル情報として明示的に指定する。ブロックの割り当ては、モデル変換時(ONNX→Ceras) に行われ、各層の生存期間(データの使用開始から不要になるまでの期間)を考慮して、必要な期間中はデータが失われることなく保持されるように、メモリブロックが安全に共有できるよう設計されている。

このメモリ管理戦略は、特に FPGA のようなオンチップリソースに制約がある環境において、高速処理と省メモリ動作の両立に貢献する。

4. 結果

4.1 量子化前後のモデルサイズの比較

表 1 量子化前後の比較

種類	ファイルサイズ[B]	層の数
量子化前	10861822	252
量子化後	2678012	142

表 1 は、YOLOX-Nano deploy 版の量子化処理の前後における Ceras 形式モデルのファイルサイズと層の数の変化を示している。提案手法を適用することで、量子化前後でモデルのファイルサイズおよび層数に顕著な差が見られた。本モデルでは、Conv2D 層とアクティベーション層（ReLU など）を統合する処理を行っており、Conv2D の出力に対してアクティベーションを内部的に実行する構成としている。この結果、明示的なアクティベーション層を削除することが可能となり、層数を削減している。なお、この融合処理の詳細な手法については本稿では説明を省略する。

その結果、
・ファイルサイズは約 1/4 に圧縮
・全体の層数は約 40%削減
という効果が得られており、モデルの実行効率やリソース消費の面で大きな利点があることが確認された。

4.2 精度結果の比較

表 2 推論精度の比較

モデル	実行環境	mAP
Pytorch(浮動小数型32)	GPU	0.285
VitisAI(整数型8)	DPU	0.186
Ceras(整数型8)	CPU	0.196

本研究では、量子化による精度変化を評価するため、YOLOX-Nano deploy 版を用いて各実行環境における精度の比較を行った。 精度指標には、COCO 評価指標に基づく mAP@0.5（mean Average Precision at IoU=0.5） を採用した。 表 2 に結果を示す。

その結果、Ceras 上で実行した int8 量子化モデルは、Vitis-AI の DPU 上で実行した量子化モデルを上回る精度を達成した。これは、Ceras におけるスケール整合処理や逆量子化制御など、提案手法における量子化設計の工夫によるものである。 この結果より、提案手法が精度・効率の両面において有効であることを示している。

5. まとめ・今後の展望

本研究では、**エッジ AI 向けの推論実行を目的として、**
1.Vitis-AI を用いた int8 量子化の導入
2.層間スケール整合によるスムーズな整数演算処理
3.共有メモリのブロック分割によるメモリ効率の最適化
の 3 点を柱とする手法を提案した。

ファイルサイズや層数の削減に加え、推論精度も DPU を上回る結果を得た。これにより、本手法がリソース効率と推論精度を両立した量子化実行環境として有効であることが確認された。

今後は以下の方向で本手法の拡張・発展を検討している：
・より大規模なモデル(YOLOX-S など)への適用と評価
・浮動小数点／整数演算のハイブリッド実行時制御
・実機上での性能評価

これらにより、Ceras と量子化モデルの連携を強化し、実環境で動作可能な高効率・高精度なエッジ AI 推論システムの構築を目指す。

参考文献

[1] 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)

† 大阪工業大学 情報科学研究科 情報科学専攻
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