

組み込みシステムにおける深層学習フレームワークによる 学習結果を用いた認識機能の自動生成の試み

松本 斗貴[†] 趙 茜[†] 中本 幸一[†]

[†]兵庫県立大学大学院 応用情報科学研究科

1 はじめに

深層学習が様々な分野で利用され始めており、組み込みシステム分野においても需要がある。たとえば、自動車の自動運転では正確な画像認識技術が要求される。そのため深層学習が利用されている。

しかし、組み込みシステムにおいては厳しいリソース制約があるため、低消費電力、省メモリかつリアルタイム性を考慮した実装が要求される [1]。しかし、GPU を利用した実装は消費電力が過大となり、また CPU だけを利用した実装は性能が不足し、リアルタイム性を求められるシステムには不向きである。そのため組み込みシステムにおけるリソース制約を満たすために FPGA が使用されることが想定される。その理由として FPGA は特定の演算に対する回路を柔軟に構築可能であり、CPU と比べて高速かつ GPU と比べて電力消費を抑えて演算することが可能だと言われているためである。

そこで本研究では厳しいリソース制約の中で深層学習による認識機能を組み込みシステムで使用される FPGA に実装するにあたって作業者の負担を軽減し、開発作業の効率化を目的とした認識機能の自動生成方法を提案する。

2 関連研究

本研究は FPGA を用いた組み込みシステム向け深層学習フレームワークの構想 [2] によって提案されたフレームワーク (図 1) の C Source Code Generator の役割を担っている。提案されたフレームワークは既存の汎用マシン向け深層学習フレームワークによる学習結果から、高位合成向けに最適化した深層学習の認識部分の C 言語ソースコードを生成する。高位合成向けの最適化については高位合成のための知識を持たない開発者でもハードウェアを意識せず容易に開発できるよう、柔軟に対応できるようにする。

以上によって作成された C 言語ソースコードから、SystemBuilder によってソフトウェア/ハードウェアに対応したモデルを生成し、高位合成ツールまたはコ

ンパイラによって FPGA 向けアクセラレータおよび実行ファイルが生成され、FPGA から深層学習が利用可能となるものである。

3 深層学習フレームワークによる学習

本研究では現在、既存の深層学習フレームワーク Caffe[3] を利用している。そこでまず Caffe による学習を行う。Caffe とはカリフォルニア大学バークレー校のコンピュータビジョンおよび機械学習に関する研究センターである BVLC が中心となって開発している Open Source Deep Learning Framework である。C++ で実装され、GPU に対応しているため高速な計算処理が可能である。

また、Caffe はニューラルネットワークの構造を記述するためにプロトテキストと呼ばれる独自のテキスト形式を使用する。プロトテキストは Google が開発したデータ構造を記述するためのテキスト形式の一種である Protocol Buffers の文法に基づいたものである。Protocol Buffers は C++, Java などいくつかのプログラミング言語をサポートしている。

学習は LeNet-5 を使用し、0~9 の 10 種類の手書き数字画像の分類を行った。LeNet-5 は畳込み層が 2 層、プーリング層が 2 層、全結合層が 2 層の畳込みニューラルネットワークである。また、手書き数字画像の分類を行うために MNIST データセットを使用した。MNIST とは 28x28 ピクセルの手書き数字の画像データセットであり、トレーニングデータが 6 万枚、テストデータが 1 万枚と合計で 7 万枚の画像が用意されているものである。

4 学習結果を用いた認識機能の生成

学習の結果、深層学習の認識において必要なパラメータが計算される。それらのパラメータおよびニューラルネットワークの構成情報から認識部の C 言語ソース

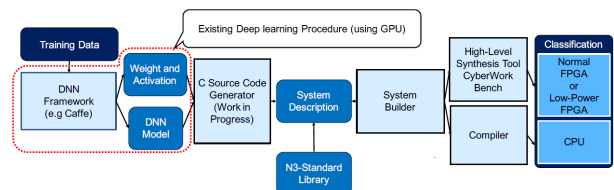


図 1: 組み込みシステムのための深層学習フレームワーク

Attempt to Automatically Generate Recognition Function Using Learning Results by Deep Learning Framework in Embedded System

[†] Toki Matsumoto, Qian Zhao, Yukikazu Nakamoto · Graduate School of Applied Informatics University of Hyogo

コードを生成する。本研究ではC言語でハードウェア・ソフトウェアを含むシステム全体を記述し、その一部をFPGAでハードウェア化する方法をとる。FPGA化するにはNEC社製のCyberWorkBench[4]を用いる。CyberWorkBenchは、NEC社製のASIC・FPGA設計向けC言語ベース高位合成ツールであり、C言語からの高位合成とC言語レベルの機能・タイミングデバッグや形式検証を実施可能である。これによりFPGAの設計をC言語ソースコードで行うことができる。

本手法では、Caffeによる学習結果を利用して認識を行うために学習により計算されたパラメータを用いた学習済み畳み込みニューラルネットワークをC言語ソースコードで生成する必要がある。学習済み畳み込みニューラルネットワークをC言語ソースコードで生成するためにprototxtファイルとcaffemodelファイルの解析を行った。

はじめに畳み込みニューラルネットワークの構造をprototxtから読み取り、認識のためのC言語ソースコードプログラムを生成する。次に学習によって計算された認識に必要なパラメータをcaffemodelファイルから抽出する。パラメータはバイナリーデータとしてcaffemodelファイルに入っているため、プロトテキスト形式に従ってテキストデータに変換を行った。最後にテキストデータに変換されたパラメータをC言語ソースコードのデータとして抽出し、抽出したデータを認識プログラムで使用することによってCaffeによる学習結果を使用した認識機能をC言語ソースコードで実現した。一連の流れを図2に示す。

5 認識機能の自動生成の試み

前項で述べた通り、認識機能を実現するには図2の工程が必要となる。そこで本研究では認識プログラムに必要なデータを学習によって得られたパラメータのテキストデータから抽出し、認識プログラム内の変数への代入の自動化を行った。

LeNet-5での手書き数字の画像分類の学習によって得られたパラメータは膨大な量であった。学習の対象が28x28ピクセルの手書き数字画像よりサイズが大きな画像や複雑な構成の画像であった場合や畳み

ニューラルネットワークの構成に変更があった場合、手書き数字画像分類の学習結果より多くのパラメータが出力されることが想定される。

このようなことから自動化の提案を行うことによって学習結果である膨大な量のパラメータを認識を行う度にテキストデータに変換した学習結果を作業者が参照するのではなく、学習結果のテキストデータを入力するだけでデータの抽出から認識プログラムの変数への代入、実行までをソフトウェア自身がを行い、自動的に認識結果が得られるようにすることで作業者の負担を軽減し、開発作業の効率化が望める。

6 おわりに

本研究は組込みシステムにおいてCaffeによる学習結果を利用して深層学習を実装する際にバイナリデータであるcaffemodelファイルをテキストデータに変換した結果から必要なデータを抽出し認識プログラムの変数へ代入し認識機能を実現する工程の自動化を行った。今後の展望としてはcaffemodelファイルをテキストデータに変換する工程を含めた全自動化を行うソフトウェアの提案、実装を行う必要がある。

7 謝辞

この成果は、国立研究開発法人新エネルギー・産業技術総合開発機構(NEDO)の委託業務の結果得られたものである。また、一部JSPS科研費17H00762の助成を受けている。

参考文献

- [1] 高田広章：組込みシステム開発技術の現状と展望，情報処理学会論文誌，Vol. 42, No. 4, pp. 930-938 (2001).
- [2] 山本 椋太, 岡本 卓也, 本田 晋也, ほか：FPGAを用いた組込みシステム向け深層学習フレームワークの構想，情報処理学会組込みシステムワークショップ(2017)
- [3] Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., Guadarrama, S. and Darrell, T: Caffe: Convolutional Architecture for Fast Feature Embedding, arXiv preprint arXiv:1408.5093 (2014).
- [4] Wakabayashi, K: CyberWorkBench: integrated design environment based on C-based behavior synthesis and verification, VLSI Design, Automation and Test, 2005. (VLSI-TSA-DAT). 2005 IEEE VLSI-TSA International Symposium, IEEE, pp. 173-176 (2005)

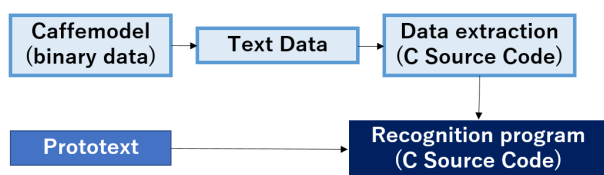


図2: 認識機能生成手順