FPGA+ Chainerでディープ・ラーニング

Altera SDK for OpenCLで電力効率に優れたアクセラレータを実現できる

Chainerを使ったディープ・ニューラル・ネットワークをFPGAで動かしてみよう

中原 啓貴 Hiroki Nakahara

前号（FPGAマガジンNo.15）は，Altera SDK for OpenCLを使って多層ニューラル・ネットワークによる手書きの数字を認識させました．今回は，日本発のディープ・ラーニング・フレームワークであるChainerを使って，畳み込みディープ・ニューラル・ネットワーク（CNN）を設計し，学習したモデルからOpenCL向けのC++を生成して，FPGA評価ボード上に実装してみます．

● HDLでニューラル・ネットワークの設計は困難

FPGAは，消費電力当たりの性能効率がGPUよりも優れていることから，ディープ・ラーニング・アクセラレータとして注目されています．しかしFPGA上に任意の回路を設計するには，従来はVerilog HDL/VHDLによる（Register Transfer Level）レベルの設計が必要であり，敷居が高く時間がかかることが問題でした．ディープ・ラーニング・アクセラレータ回路を実現するにはディープ・ニューラル・ネットワーク自体の設計も必要であり，実現するための敷居はますます高くなっています．

● Altera SDK for OpenCLの登場

FPGAの設計問題を解決するため，Altera社（現在はIntel社のFPGA部門）はAltera SDK for OpenCL（現在はIntel SDK for OpenCLだが，ここでは以前の名称で呼ぶ）をリリースしました．環境が制限されるものの，OpenCLに準拠したC++コードを書くだけでFPGAにアクセラレータを実現でき，ホストから簡単に制御できます．HDLを書く必要がないため，開発期間を短縮することができ，かつ，ソフトウェア技術者も参入しやすい利点があります．

ディープ・ニューラル・ネットワークに関しては，各社からフレームワークが提供されており，多くはPythonコードで記述してGPUを使って高速に学習ができます．従って，ディープ・ラーニング・フレームワークからAltera SDK for OpenCLにつなぐことができれば，電力効率に優れたディープ・ラーニング・アクセラレータを容易に実現できます．

1 畳み込みディープ・ニューラル・ネットワーク（CNN）の特徴

● 画像認識に最適

ニューラル・ネットワークを多層に並べたものをディープ・ニューラル・ネットワークといいます．ディープ・ニューラル・ネットワークを深くすればするほど認識精度が上がることは分かっていましたが，過学習や適切なニューロンの値を探索する範囲が広すぎたことから学習することは困難でした．

近年になって，自己符号化器や畳み込みディープ・ニューラル・ネットワークが提案され，ディープ・ニューラル・ネットワークの学習ができるようになり，多くの分野で応用されています．ここでは画像認識に適している畳み込みディープ・ニューラル・ネットワークについて説明します．

● 手書き文字認識をさせる

図1に手書き文字認識タスクの一種である，MNISTデータベースを認識する畳み込みディープ・ニューラル・ネットワーク（CNN：Convolutional Neural Network）を示します．CNNは，

- 畳み込み層
- プーリング層
- フル結合層

の3層で構成されています．今回は**図1**に示した構成で実装します．

ニューロンを2次元に並べたものを，特徴マップといいます．CNNは特徴マップを各層ごとに複数並べ，次層の特徴マップを畳み込み，プーリング演算，フル結合演算で計算していきます．

● 畳み込み層の動作

まず，畳み込み層の説明をします．**図2**に畳み込み演算を示します．N個の入力特徴マップとM個の出力特徴マップがあり，$K \times K$サイズのカーネルという計算単位で畳み込み演算を行います．例えば，ある出力特徴マップのニューロンを計算するには，N個の入力特徴マップに対してそれぞれ$K \times K$サイズの重みを掛けたあと，総和をとる積和演算を行います．これをM個の出力マップの全てのニューロンに対して行います．ただし，畳み込み演算では1つの出力特徴マップに対して同じ重みを用います．つまり，重みを頻繁に読み出すことはありません．従って，畳み込み層では積和演算がボトルネックになることが知られています．

● プーリング層の動作

図3にプーリング演算の説明をします．入力画像の