4月号特集で使ったJetson Orinの活用術

Jetson大実験4…
モデル圧縮手法 プルーニングを試す

土井 伸洋

Jetson AGX Orinシリーズ(エヌビディア，**写真1**)のGPUは，Ampere世代のアーキテクチャを採用しています．このAmpere世代のGPUには，第3世代のTensorCore(行列演算ユニット)が搭載されており，スパース(値のほとんどがゼロ)なニューラル・ネットワークへの対応機能が新たに追加されています．これをうまく利用すれば推論速度/メモリ消費量を改善できるのでチャレンジしてみます．

本稿は2023年4月号 特集1(第9章，第10章)を前提に話を進めますので，必要に応じて当該記事を参照ください．

量子化だけじゃない…
モデルを圧縮する手法「プルーニング」

深層学習モデルを圧縮/軽量化し，高速に実行する手段として，これまで量子化(quantization)を実行してきました．それ以外にも手段はあり，代表的なものの1つがプルーニング(pruning)です．

● 畳み込み演算を例に圧縮手法の土台を知る

ここでは3×3の畳み込み演算を行う場合を考えます．また，縦方向のエッジ検出を処理例として挙げると，畳み込み演算の重みは3×3のテンソルとなります(**図1**)．

このとき，**図2**(**a**)のように3×3の重みをまるごと保持した場合(Denseと呼ばれる表現方法)，32ビット×9＝288ビットの領域が必要であり，畳み込み演算には9回の乗算と1回の総和演算が必要です．

重みの中身を見ると，出力に関与しないもの(0)が複数あります．そこで，出力に関与するものに限って値を保持(Sparseと呼ばれる表現方法)すると，**図2**(**b**)のように32ビット×6のインデックスのみで済みます(インデックスは，各有効重みがDenseテンソル

写真1　Jetson AGX Orinで高速道路の白線を認識中

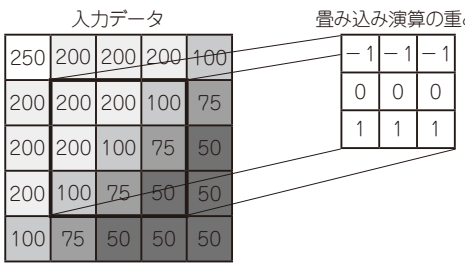

図1　画像の縦方向に対するエッジ処理
データと重み(float型で32ビット)の畳み込み演算をすればよい

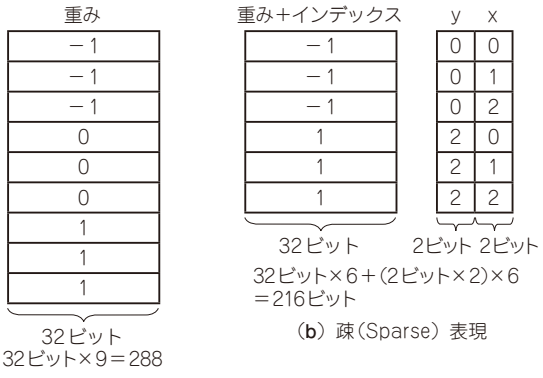

図2　出力に関与する重みだけを保持するようにすればデータ量/演算量を削減できる