

# 表題 The Evaluation of DCNN on Vector-SIMD DSP

著者 C. Yang, S. Chen, Y. Wang, J. Zhang from  
所属 National University of Defense Technology, Changsha, China  
出典 IEEE Access, vol. 7, pp. 22301-22309, 2019

立命館大学 理工学部電子情報工学科4回生

渥美 柁彦

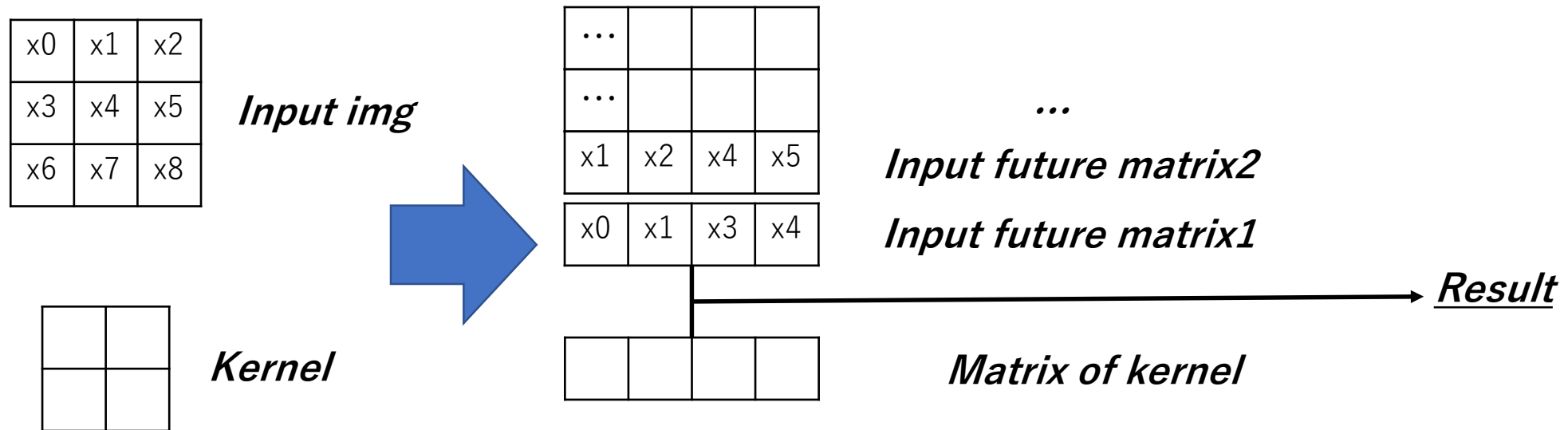
発表日 2020年6月10日

# 概要

- DCNN (Deep Convolutional neural network) を実行の際、プロセッサ上で、ベクターマッピング、SIMDを利用することで、無駄なデータパスの削減、不必要なクロックの削減する
- 性能評価を行い、現行のGPUとの性能を比較を行う。

# 重要な点

- CPU上でのDCNNの実行の際、入力値をベクトル演算により処理をする。
- 一般的なDCNNにおける、演算処理に関する概要図。



# 重要な点-ベクトル演算 (SIMD) とは

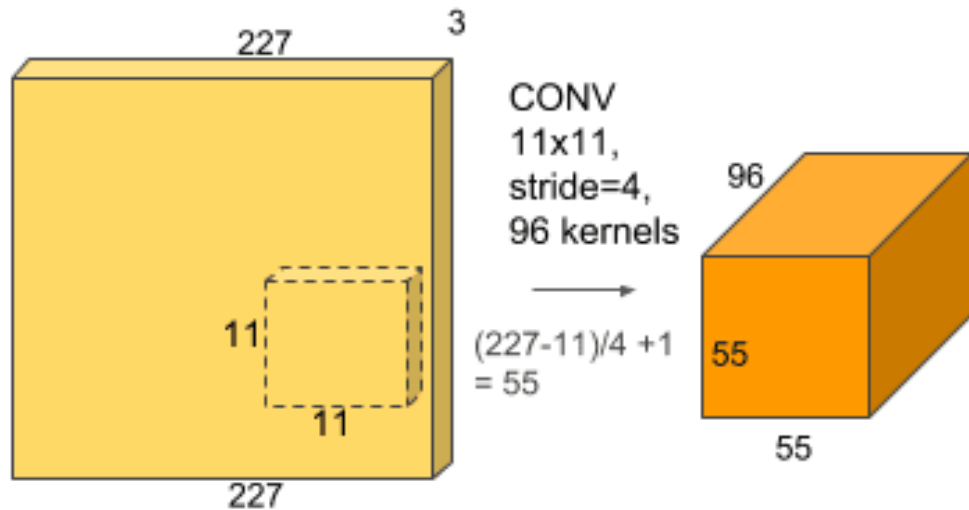
- 本論文で利用されている手法
- 下記の式を一命令で並列処理できる演算

## ***Vector-SIMD***

$$\left\{ \begin{array}{l} a1 \leq b1 \times c1 \\ a2 \leq b2 \times c2 \\ a3 \leq b3 \times c3 \end{array} \right.$$

- 仮に、各乗算が一命令32ビットで行われるとき、
- 32ビットのレジスタ幅であれば逐次実行であるが、
- 96ビットのレジスタ幅であれば一命令で実行することができる。

# 提案手法の例を考える(準備)



画像: AlexNet第一層畳み込み演算

cite:

<https://neurohive.io/en/popular-networks/alexnet-imagenet-classification-with-deep-convolutional-neural-networks/>

<ハードウェア上の設定>

- 1CORE当たりの演算ユニット数U: 16

<畳み込み演算の設定>

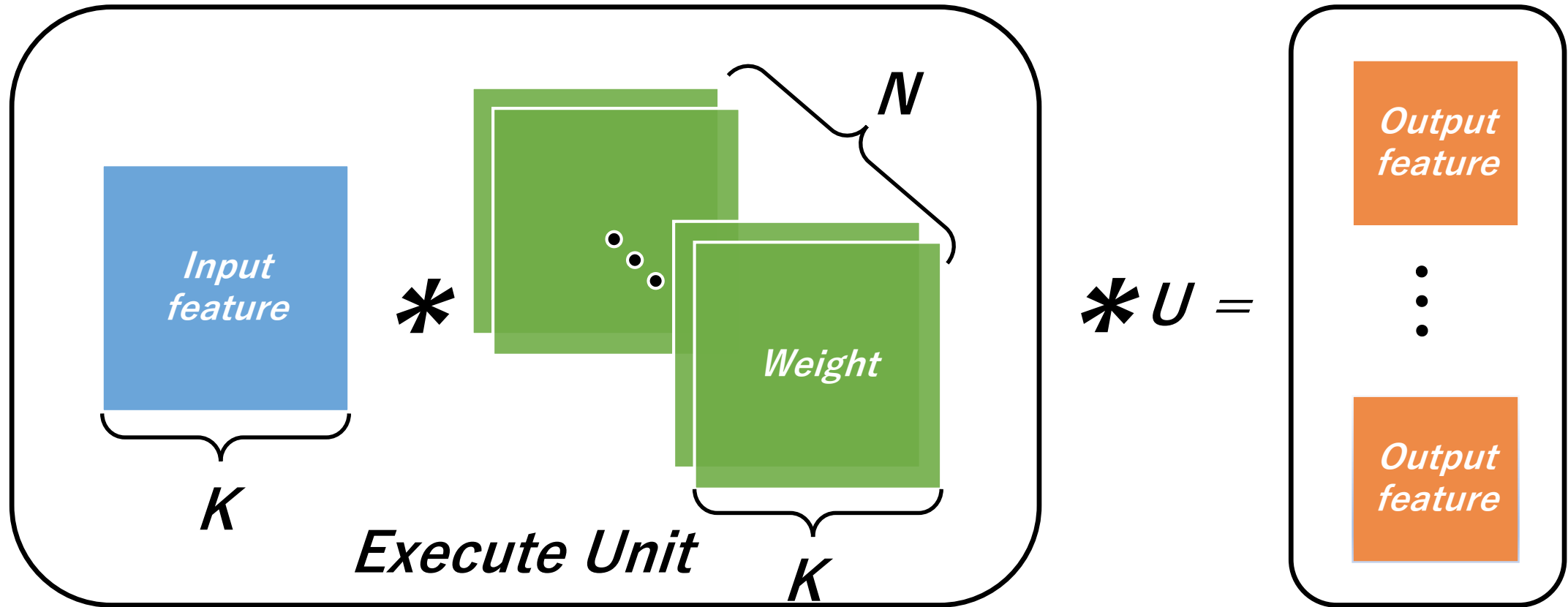
- カーネルサイズSK: 11
- カーネル数K: 96
- 入力サイズIS: 227
- 出力サイズOS: 55

<特別な引数>

- カーネル数K / 演算ユニットU = 6(N)

# 提案手法の例を考える(マッピング、演算)

- 各演算ユニットでの複数演算の実行



# 効率化実現の理由

- DRAMとCORE間のデータ転送時間を効率化によりエネルギーの節約を実現。（ハードウェア資源の節約）

→複数のデータを一度に作業領域まで転送するため。  
（ハードウェア資源の節約）

→しかし、実際のハードウェアには資源の制限が存在するため、制限内での効率性を目指す必要がある。

# 実験結果

- CuDNNが搭載されている、GTX1080 ,GTX1080Tiと計算効率、エネルギー消費の効率性について比較。
- 提案手法においては、NVIDIAのGPUに比べ、20%から30%計算効率が大きかった。
- エネルギー効率においては、NVIDIAのGPUに比べ、9%から30%エネルギー消費が効率的であった。
- また既存の演算処理に比べ、計算効率は62~75%向上、エネルギー効率は56%向上した。