

# 動的ビジョンセンサー搭載エッジ AI 向きスパイク駆動神経回路網モデルの検討

岡村 健勝 【 コンピュータ構成学研究室 】

## A Study on Spiking Neural Network on Edge-AI Platforms with Dynamic Vision Sensor Integration

Kensho Okamura 【 Advanced Computer Engineering Lab. 】

### 1 はじめに

近年、IoT エッジデバイス上で高度な推論機能を実現するエッジ AI 技術への期待が高まっている。一方で、AI 機能には膨大な計算量とメモリ量が必要になり、低コストなエッジデバイス上での実現には課題がある。そこで、本研究では、従来型の人工神経回路網ではなく、より生物の神経回路に近いスパイク駆動型神経回路網 SNN (Spiking Neural Network) に着目した。SNN はニューロン間でスパイク信号を伝搬することによって機能するため、原理的に、必要最小限のイベントが授受され、低消費電力であるとされている。さらに、本研究では、監視カメラ等の応用を想定して、従来型のフレームベース画像センサではなく、輝度値が変化した画素情報のみを出力する動的ビジョンセンサー DVS (Dynamic Vision Sensor) にも着目している。これは、低コストなエッジデバイスでは、入出力データ帯域も限定的であること、ならびに、SNN のイベント駆動動作とも親和性が高いため、その相乗効果で低消費電力化が期待できるためである。しかしながら、SNN では神経細胞の膜電位を微分方程式に基づいて計算するため、汎用プロセッサ上でのソフトウェアでは計算量が膨大になり、エッジデバイスでは実現が難しい。

そこで、本研究では、動的ビジョンセンサー搭載エッジ AI を FPGA や ASIC 等による専用ハードウェアで実現することを最終目標として、それに適したスパイク駆動神経回路網モデル [1, 2] の軽量化に関して検討した。その際、IBM 研究所が公開している DVS128-Gesture dataset[3] を対象とした分類精度ならびに回路コストを指標として、各種軽量化手法について比較評価した。

### 2 既存の DVS 応用向け SNN

#### 2.1 全結合型 SNN [1]

全結合型 SNN (FC-SNN: Fully Connected SNN) は、全結合層のみで構成される。回路実装する場合、シナプス重みを  $i$  Byte とすると全シナプス重みの格納に必要なメモリ容量  $S_{mem}[\text{Byte}]$  は、入力層、中間層、出力層のニューロン数をそれぞれ  $N_{input}$ ,  $N_{hidden}$ ,  $N_{output}$  とすると、式 (1) で表せる。

$$S_{mem} = \{N_{input} \times N_{hidden} + N_{hidden} \times N_{output}\} \times i \quad (1)$$

文献 [1] では  $N_{input}=1024, N_{hidden}=512, N_{output}=11$  の FC-SNN が FPGA 実装されており、入力層-中間層間のシナプス重みが全シナプス重みの約 99% と大半を占めており、その削減方法が課題である。その際に、精度低下への配慮が必要になる。

#### 2.2 畳み込み型 SNN [2]

畳み込み型 SNN (CSNN: Convolutional SNN) では、畳み込み後の特徴マップの各空間位置に対応した LIF ニューロンが備えられており、FC-SNN に比べ、ニューロン数が数十倍必要になる。ハードウェア実装を踏まえるとニューロン数の削減は必須である。ニューロン数は畳み込み後の層構成やチャンネル数によって大きく変化する。また CSNN の層の構成や畳み込み後のチャンネル数によるニューロン数の変化が精度に与える影響も考慮する必要がある。

### 3 軽量化手法

本研究の目的はネットワーク構成を軽量化した際の性能を評価するため、DVS128gesture データセット [3] を縦 32 × 横 32 の範囲に縮小したデータを入力としたモデルを対象として、軽量化を行った。

#### 3.1 FC-SNN の軽量化

文献 [1] のネットワーク構成に加えて、DVS データの極性（輝度の増減）を考慮して、入力層 2048 ニューロン、中間層 512 ニューロン、出力層 11 ニューロンの構成 (2048-512-11) を比較基準モデルとして、入力層-中間層間のシナプス結合の枝刈りと、中間層ニューロンの削減を試みた。

#### 3.2 CSNN の軽量化

(a) 畳み込みモジュールの構成： 図 1 に示すように、CSNN では畳み込み (conv2d) 後に LIF 層を経由して pooling している。しかし畳み込み後に pooling し LIF 層で処理する構成 (Conv2d-pooling-LIF) により LIF 層

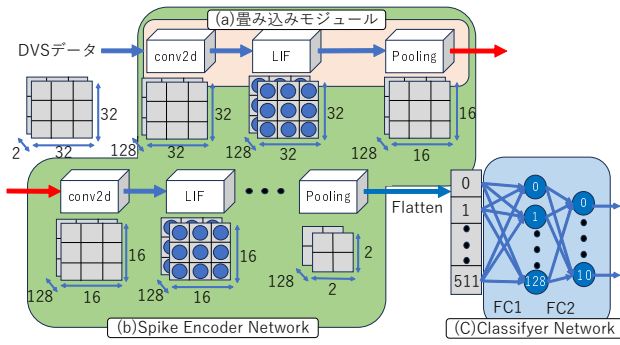


図1 CSNNの軽量化ポイント

のニューロン数を pooling した割合だけ削減できる。  
 (b) Spiking Encoder Network の構成：図1では第一層の畳み込み層で2チャンネルを128に拡張している。これに対して、第一層の畳み込み後のチャンネル数を16に削減し、以降の複数の畳み込み層でチャンネル数をそれぞれ2倍に増加させ、全4層で128チャンネルに拡張して、全体的に必要なニューロン数を削減する。  
 (c) Classifier Network の構成：全結合層2層を1層に縮退してニューロン数を削減する。

#### 4 評価

本研究では、図2における灰色のモデルについて評価を行った。学習にはDVS128 gesture datasetを使用し、学習データ1077サンプルを5分割して交差検証し、それぞれで検証結果が最も高いエポックの重みを使ってテストデータ264サンプルを推論した結果の平均をそのモデルの推論精度とした。またFC-SNNに関しては pruning 後に同一学習データを使い追加学習を行った。各モデルのリソースについてはLIFニューロン数  $N_n$ 、シナプス数  $N_s$  から要求されるメモリ使用量 Mem(Byte)とし、既存の実装結果 [1] に基づき  $Mem[Byte] = N_n \times 3 + N_s$  と定義した。また軽量化効果を定量化するために、ベースライン精度 Base を基準にして、 $(Acc.-Base)/Mem$  を算出した。

FC-SNNのうち、比較基準モデル、最高分類精度のモデル、 $(Acc.-Base)/Mem$  が最高値のモデルの評価結

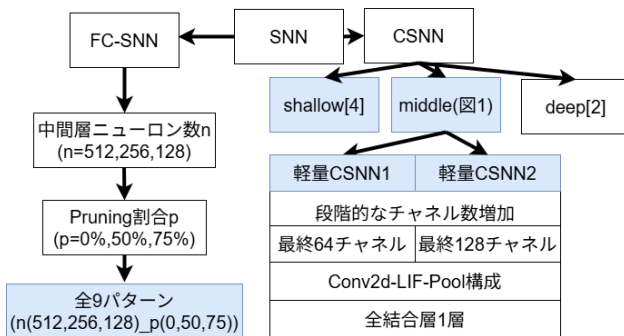


図2 DVS 応用向け軽量化 SNN モデルの派生関係

果のみ抜粋して表1に示す。ただし、Baseを最軽量構成 n32\_p75%の精度 71.32%とした。その結果、分類精度自体に大差はなく、中間層ニューロン数128で、シナプス結合を75% pruningした n128\_p75が軽量化効果が最善となった。

軽量化CSNNの評価結果を表2に示す。ただし、BaseをFC-SNNのおおよその精度80.00%とした。ShallowCSNNが最も軽量の構成だが、総合的な結果としては、軽量CSNN1が最良構成であった。

#### 5 まとめ

本研究では、動的ビジョンセンサー向きスパイク駆動神経回路網モデルに関して検討し、いくつかの軽量化手法を提案した。DVS128-Gesture datasetを対象として軽量化効果を比較検討した結果、poolingによりLIFニューロンを削減した層を4層備えたSpiking Encoderと全結合層1層の分類器を組み合わせたCSNNが最も軽量化効果が高いことが分かった。今後は、具体的な回路設計・実装を通して回路コストを精密に評価して、軽量化効果の妥当性を検証する必要がある。

#### 参考文献

- [1] X. Cheng, et al., "An FPGA-Based Event-Driven SNN Accelerator for DVS Applications With Structured Sparsity and Early-Stop," IEEE Trans. Circuit and Systems—I, Vol. 72, No. 7, 2025.
- [2] W. Fang, et al., "Incorporating Learnable Membrane Time Constant to Enhance Learning of Spiking Neural Networks," ICCV 2021.
- [3] A. Amir, et al., "A Low Power, Fully Event-Based Gesture Recognition System," CVPR 2017.
- [4] Denis Larionov, "dvs128gesture+snntorch," kaggle, <https://www.kaggle.com/code/dlarionov/dvs128gesture-snntorch> (参照 2026-01-24)

表1 軽量化FC-SNNの評価結果

#neuron _%pruning	Acc. [%]	Mem [K Byte]	(Acc.-Base) /Mem
n512_p0%	80.29	1,056	0.85E-5
n256_p50%	<b>81.25</b>	264	3.76E-5
n128_p75%	78.46	<b>67</b>	<b>10.60E-5</b>

表2 軽量化CSNNの評価結果

#neuron _%pruning	Acc. [%]	Mem [K Byte]	(Acc.-Base) /Mem
ShallowCSNN	87.5	<b>28</b>	26.32E-5
MiddleCSNN	<b>93.82</b>	1,034	1.34E-5
軽量CSNN1	90.88	39	<b>28.11E-5</b>
軽量CSNN2	91.25	126	8.95E-5