BLINK: Bit-Sparse LSTM Inference Kernel Enabling Efficient Calcium Trace Extraction for Neurofeedback Devices

Zhe Chen
University of California, Los Angeles
Los Angeles, California
zhechen@ucla.edu

Garrett J. Blair
University of California, Los Angeles
Los Angeles, California
gblair@ucla.edu

Hugh T. Blair
University of California, Los Angeles
Los Angeles, California
tadblair@ucla.edu

Jason Cong
University of California, Los Angeles
Los Angeles, California
cong@cs.ucla.edu

ABSTRACT

Miniaturized fluorescent calcium imaging microscopes are widely used for monitoring the activity of a large population of neurons in freely behaving animals *in vivo*. Conventional calcium image analyses extract calcium traces by iterative and bulk image processing and they are hard to meet the power and latency requirements for neurofeedback devices. In this paper, we propose the calcium image processing pipeline based on a bit-sparse long short-term memory (LSTM) inference kernel (BLINK) for efficient calcium trace extraction. It largely reduces the power and latency while remaining the trace extraction accuracy. We implemented the customized pipeline on the Ultra96 platform. It can extract calcium traces from up to 1024 cells with sub-ms latency on a single FPGA device. We designed the BLINK circuits in a 28-nm technology. Evaluation shows that the proposed bit-sparse representation can reduce the circuit area by 38.7% and save the power consumption by 38.4% without accuracy loss. The BLINK circuits achieve 410 pJ/inference, which has 6293x and 52.4x gains in energy efficiency compared to the evaluation on the high performance CPU and GPU, respectively.

CCS CONCEPTS

• **Hardware** → **Reconfigurable logic and FPGAs**; **Logic circuits**; • **Computing methodologies** → *Machine learning*.

KEYWORDS

Calcium image processing, energy efficient, long short-term memory (LSTM), neurofeedback

ACM Reference Format:
Zhe Chen, Garrett J. Blair, Hugh T. Blair, and Jason Cong. 2020. BLINK: Bit-Sparse LSTM Inference Kernel Enabling Efficient Calcium Trace Extraction for Neurofeedback Devices. In *ACM/IEEE International Symposium on Low Power Electronics and Design (ISLPED '20), August 10–12, 2020, Boston, MA, USA.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3370748.3406552

ISLPED '20, August 10–12, 2020, Boston, MA, USA

ACM ISBN 978-1-4503-7053-0/20/08.
https://doi.org/10.1145/3370748.3406552

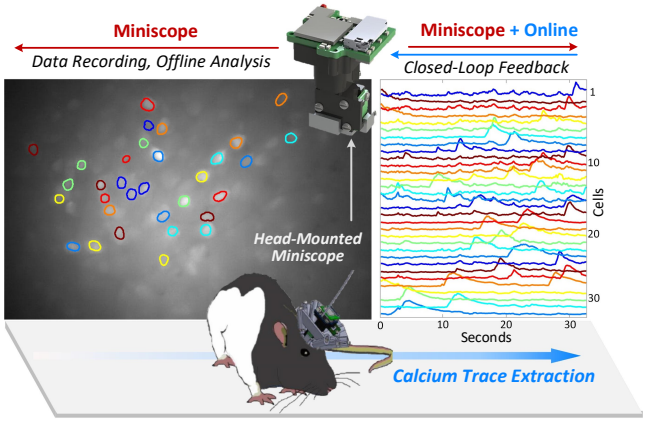

Figure 1: Calcium trace extraction from images recorded by the miniscope for closed-loop feedback applications.

1 INTRODUCTION

In-vivo calcium imaging is an emerging technique for monitoring the activity from a large population of neurons in the brain of a freely behaving animal, such as mouse or rat [1]. Such technique is usually realized by mounting a miniaturized fluorescence microscope ("miniscope") onto the animal's head to obtain video recordings of calcium activity, as Fig. 1 shows. In such recordings, a punctate flash of fluorescence is observed at a specific location when a particular neuron becomes active. The miniscope can record the activity from hundreds of neurons simultaneously over weeks or months as the animal engages in various behavioral tasks.

The cell trace extraction from calcium images is typically done offline. Such calcium image analyses [12, 18] are computationally intensive, and they often run slower than the imaging even deployed on high-end CPUs or GPUs. Given the computation complexity, it is hard to meet the power and latency requirements for neurofeedback devices, which requires a $2^{\circ}C$ temperature increase limit as neural implants [14]. As the temporal and spatial resolution of the miniscope increase [10], the gap between the desirable and achievable energy efficiency and latency escalates.

A high energy-efficiency and short-latency cell trace extraction with accuracy close to the offline analysis method is in demand, because it will enable the online decoding of the neural activity

and support a wide range of neurofeedback applications with the closed-loop feedback capability.

In this paper, we propose a bit-sparse long short-term memory (LSTM) inference kernel (BLINK) based calcium image processing pipeline. First, we introduce the proposed trace extraction method based on the LSTM inference. Then we introduce the bit-sparse representation of the LSTM inference and evaluate the energy efficiency gain. Finally, we design circuits for the BLINK and make comparison on the energy efficiency and performance against evaluations on multi-core CPU and GPU platforms. The contributions of the paper are summarized below:

- To our best knowledge, we are the first to propose an LSTM-based calcium image processing pipeline that has the potential to enable the closed-loop neurofeedback at the spike-timing resolution. It removes the acausal delay while remaining the trace extraction accuracy by using the LSTM inference to approximate the offline method.
- We propose the bit-sparse representation that improves the energy efficiency of the LSTM inference by replacing the dense matrix multiplication with the bit shift operation.
- We demonstrate the customized pipeline for the 1024-cell trace extraction on the Ultra96. BLINK circuits designed in a 28-nm process achieves 410 pJ/inference, which has 6293x and 52.4x gains in energy efficiency against the evaluations on the E5-2680 CPU and the V100 GPU, respectively.

2 BACKGROUND

2.1 Calcium Image Processing

Several pipelines for the calcium image processing have been proposed [12, 18]. These methods use the iterative constrained nonnegative matrix factorization (CNMF) to extract the spatial footprints and temporal traces of cells simultaneously from the bulk of images. Although the implementation of the CNMF algorithm on high-end CPU and GPU platforms can meet the real-time throughput [5], two reasons prevent it from being used for closed-loop neurofeedback devices. First, the power consumption is in the order of tens of watts, which is not affordable for a head-mounted neurofeedback device powered on a single battery. The accumulated heat caused by the power consumption can easily exceed the limit of $2°C$ temperature increase for neural implants [14]. Secondly, the bulk image processing induces long and non-deterministic processing latency, which is not desired for generating the neurofeedback stimulation at the spike-timing resolution of 1 ms [10].

2.2 Online Trace Extraction

Fig. 2(a) shows an online calcium image processing pipeline composed of three consecutive steps: the motion correction, the image enhancement and the calcium trace extraction. The motion correction removes the motion artifact caused by the brain tissue movement during the recording. The enhancement gets rid of the background estimated from the current frame of image by the morphological algorithm [12]. Fig. 2(b) shows an example of the enhanced image, from which cell templates extracted by the CNMF algorithm offline can be used to extract the calcium traces online. Each cell template is an $N_C \times N_C$ binary mask, as Fig. 2(c) illustrates. Within the mask, the "1" (shaded) labels the cell footprint and the

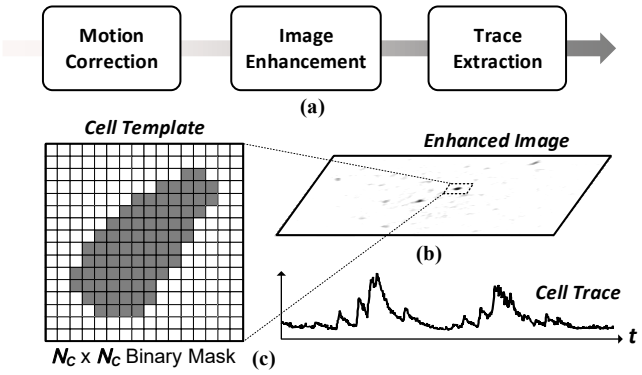

Figure 2: (a) Online calcium image processing pipeline. (b) Calcium image enhancement. (c) Template-based calcium trace extraction.

"0" (unshaded) labels the background. Pixel values corresponding to the cell footprint are accumulated as the fluorescence intensity. A time series of fluorescence intensities is extracted as the calcium trace for each cell. Fig. 2(c) shows an example of the cell template and its corresponding calcium trace extracted for the cell. There are typically hundreds of cells in the calcium image video recorded by the miniscope.

Such template-based calcium trace extraction is sensitive to the noise and it usually cannot match the CNMF method in accuracy, especially for the cells whose fluorescence has a relatively low signal-to-noise ratio (SNR). Such accuracy degradation can lead to errors in the neural signal decoding and the neurofeedback stimulation. An approach that can remain high calcium trace extraction accuracy while achieving high energy efficiency and short latency is in demand.

2.3 LSTM Approach

The LSTM [8] is a type of recurrent neural network and it has been successfully used in many time-series prediction applications such as the speech recognition and the text generation. Fig. 3 illustrates the single-layer LSTM architecture. It consists of the feedforward path and the recurrent path. The feedforward path is composed of 4 types of gates: the input gate I, the cell gate G, the forget gate F, and the output gate O. The recurrent path is composed of 2 types of nodes: the cell (C) nodes and the hidden (H) nodes. Two types of non-linear operators are used in the LSTM inference: the *sigmoid* for updating the I, the F, and the O, and the *tanh* for updating the G and the C. Finally, the LSTM output is calculated based on the weighted sum of the H nodes.

Suppose there are N_H hidden nodes in the LSTM model, the computation complexity measured in the number of operations is:

$$C(N_H) = 8N_H^2 + 14N_H. \quad (1)$$

The number of operations on the feedforward path is:

$$C_{forward}(N_H) = 8N_H^2 + 8N_H. \quad (2)$$

Considering $N_H = 5$, the operations executed on the feedforward path occupies 89% of the total operations.

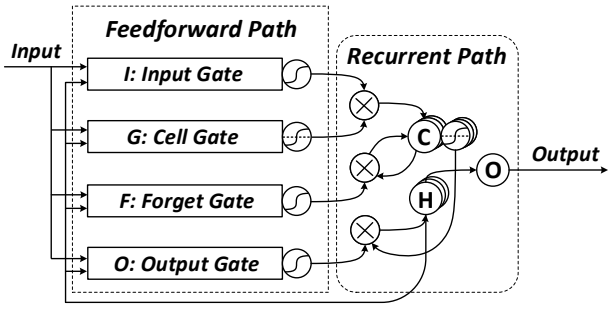

Figure 3: Architecture of the LSTM model.

3 PROPOSED METHODS

3.1 LSTM-based Calcium Image Processing

We propose an LSTM-based calcium image processing flow for efficient online trace extraction, as shown in Fig. 4. The processing flow consists of the motion correction, the image enhancement, the fluorescence tracing and the LSTM inference steps. The motion correction is based on the template matching in our prior work [2], and we concentrate on the rest of steps in this section.

The image enhancement is realized by subtracting the background from the denoised image, leaving only the dynamic foreground that reflects the calcium fluorescence generated by the active neurons. We use a 3×3 denoise filter to eliminate the black and pepper noise, and use the morphological opening [12] to estimate the background of the image.

The fluorescence tracing relies on cell templates extracted by the CNMF algorithm [18]. We first downsample the enhanced images, and then apply the cell templates on the subsampled images to extract the calcium traces according to the Section 2.2.

The LSTM inference is a processing step we proposed to improve the trace extraction accuracy. During the offline training, the online traces are used as the input and the traces extracted from the CNMF algorithm are treated as the training target. We carried out the LSTM training for each cell using the Caffe [9].After the LSTMs are well trained, they are deployed for the online trace extraction. The LSTM inference provides results that approximate to the outcome of the offline CNMF algorithm in accuracy, while significantly improving the energy efficiency and reducing the latency for the neurofeedback applications as shown below.

3.2 Bit-Sparse Data Representation

In order to improve the energy efficiency of the LSTM inference, we propose a bit-sparse data representation for the quantization of the LSTM model. In the bit-sparse data representation, we define a data type in which only a limited number of bits can be set to 1. Considering an n/M bit-sparse data, only n bits can be set to 1 whereas the n is much smaller than the M. For example, "00010000_00010000" and "00100100_00000000" are two 2/16 bit-sparse data examples. The difference between the fixed-point and the bit-sparse data representations is that the former one allows all of the bits to be able to represent 0 or 1 whereas the latter one allows only a few sparsely distributed bits to own such capability in the representation. Compared to the n-bit fixed-point, the n/M bit-sparse data type can represent values with a much wider dynamic range. We carried out the LSTM training by regulating the weights to be 1-8 bits and in 1-8/16 bit-sparse representation, respectively. It turns out that the trained weights with bit-sparse representation has 24-36 dB wider dynamic range compared to the fixed-point representation.

We perform the bit-sparse quantization of the LSTM model through retraining, a fine-tuning process commonly used for the fixed-point quantization [13]. We quantize all of the weights for updating the LSTM gates to the bit-sparse data type and keep the rest of weights as the fixed point. Fig. 5 shows the pseudocode of the bit-sparse quantization. For each weight to be quantized, we first extract the sign bit and the amplitude from the weight, and then we quantize the amplitude of the weight into the M-bit fixed-point value $wint$. From the most significant bit (MSB) to the least significant bit (LSB) of the $wint$, we count the number of 1s. When the number reaches n, we round up the value by getting rid of the rest of bits that have never been counted, and attach the sign bit back to the data representation. It can be proved that the rounding process does not increase the number of 1s, so the quantized result complies with the n/M bit-sparse format.

3.3 Algorithm Evaluation

We evaluated the proposed method with 1000-frame calcium image data recording from the mice. We first employed the CNMF algorithm to analyze the data recording. We extracted 566 cells and their corresponding calcium traces as the baseline. Then we generated the binary masks of the cells and extracted the online traces based on the binary masks using the method introduced in Section 2.2. After that, we trained a compact LSTM network (N_H = 5) for each cell by taking the online traces as the input and the CNMF extracted traces as the training target. The first half of the recording segment is used as the training set, whereas the whole segment are used for testing. We adopted the cross correlation as a measurement to evaluate the similarity between the online and offline traces. Evaluation results show that 90.3% cell traces have increases in the cross correlation after employing the LSTM inference, which indicates a pervasive improvement in the calcium trace extraction accuracy. We identified all 36 cells with a cross correlation gain over 0.15 and evaluated the accuracy performance of the bit-sparse data representation on those cells.

We performed the LSTM training under various data representations: the floating point, 16-bit fixed point and different bit-sparse data types. For each data type, the LSTM was trained independently for 5 times, and then we carried out the LSTM inference and calculated the cross correlation between the inference results and the CNMF extracted traces. We recorded the highest cross correlation score among the 5 trials to avoid occasional non-convergence of the LSTM training. In addition, we used the sum of the absolute difference (SAD) as another accuracy measurement. The SAD is calculated between the CNMF extracted and the LSTM inferenced traces obtained by the model that has the highest cross correlation score. Fig. 6 shows the averaged cross correlation and SAD among all identified cells under various data representations. According to the evaluation results, the traces extracted without the

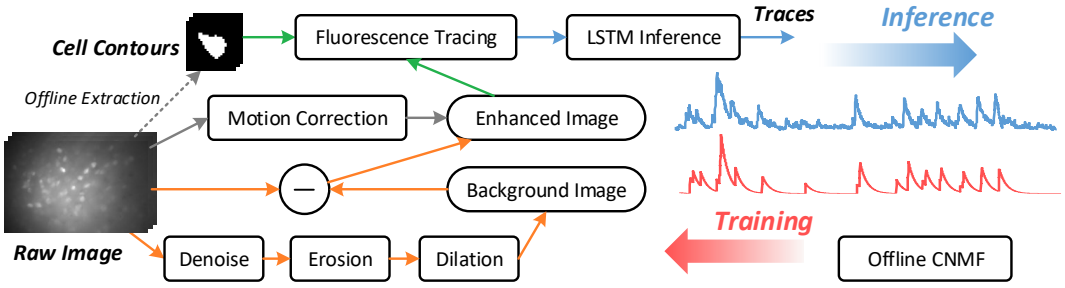

Figure 4: Proposed LSTM-based calcium image processing flow.

Algorithm 1 Bit-Sparse Quantization

```
procedure
    weight ← original weight
    n ← number of sparse bits
    M ← number of quantization bits
top:
    sign ← sign bit of weight
    wint ← quantize absolute of weight to M-bit
    i ← M − 1
    count ← 0
loop:
    if (wint ≫ i) % 2 = 1 then
        count ← count + 1.
        if count = n then
            wint = ((wint + (1 ≪ (i − 1))) ≫ i) ≪ i.
            goto end;
    i ← i − 1.
    if i = 0 then
        goto end;
    goto loop.
end:
    wint ← add sign bit to wint
    return wint
```

Figure 5: Pseudocode for the bit-sparse quantization.

LSTM inference have relatively low cross correlation scores and high SAD values, which indicate lower trace extraction accuracy. The accuracy can be improved by the LSTM inference. In addition, the accuracy improvement remains as we perform the bit-sparse quantization for the LSTM inference by adopting the 8/16 down to the 1/16 bit-sparse data type.

4 CIRCUITS AND ACCELERATORS DESIGN

4.1 BLINK Circuits

Fig. 7 shows the BLINK circuits design. The circuits feature a 1D array of bit shift operators with an array size (N_H+1), which corresponds to the 1 input and N_H hidden nodes. The bit shift operators play the same role as conventional multipliers on the feedforward path since the LSTM weights have been quantized to the 1/16 bit-sparse data type according to Section 3.3. It reduces the circuit area and improves the energy efficiency for the LSTM inference.

The recurrent part of the BLINK circuits includes N_H cell state registers CReg, N_H hidden state registers HRegs, a temporal register TReg, a cell state accumulator CAcc, an output accumulator OAcc, a shared multiplier, multiplexers and demultiplexers, and dedicated control logic for updating the cell state and the LSTM gates. A

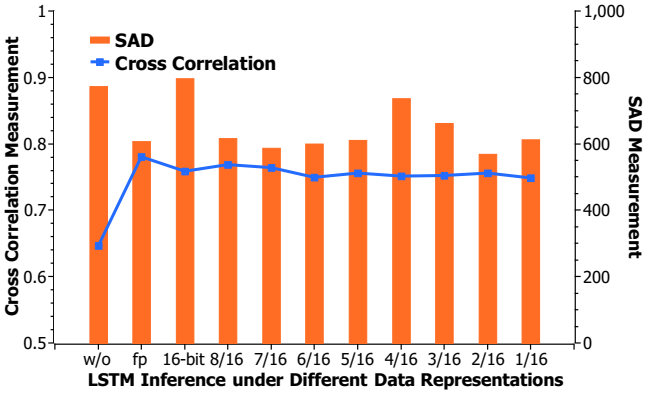

Figure 6: Accuracy evaluation for the proposed bit-sparse LSTM inference method.

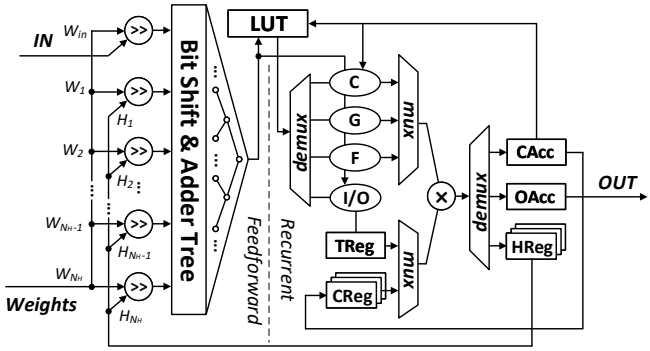

Figure 7: The BLINK circuits design.

512-entry look-up table (LUT) is used for the non-linear operations. As the circuits operate, the feedforward and recurrent parts can be fully pipelined [3], and the consumed number of clock cycles for each inference can be derived by:

$$Cycle = 5N_H + 2, \tag{3}$$

where 5 is the cycle count required to update each hidden node, and 2 indicates the initiation latency of the hidden node update.

We synthesized the circuits with the Synopsys Design Compiler using a TSMC 28 nm CMOS technology under 1 GHz. The estimated

Table 1: Reduction on the circuit area and the power consumption by the bit-sparse data representation

	16-bit	1/16 bit-sparse
Area (μm^2)	46,144	28,289
Power (mW)	24.64	15.19

circuit area and power consumption before and after using the bit-sparse quantization are shown in Table 1. Using the 1/16 bit-sparse data type reduces the circuit area and the power consumption by 38.7% and 38.4%, respectively.

4.2 Calcium Image Enhancement Accelerator

The morphological opening has been shown to be effective for the calcium image enhancement [12]. It estimates the background of the calcium image from the current frame. The morphological opening is made up of two consecutive steps, the erosion and the dilation, and they are defined as:

$$(f \ominus b)(u,v) = \inf_{(x,y)\in B} [f(u+x, v+y) - b(x,y)] \quad (4)$$

$$(f \oplus b)(u,v) = \sup_{(x,y)\in B} [f(u+x, v+y) + b(x,y)] \quad (5)$$

where $f(u,v)$ represents the image to be processed and $b(x,y)$ represents the operational template defined in $B \in N^2$. We set $b(x,y) = 0$ so that the erosion/dilation operation can be realized by calculating local minima/maxima within the template region B. Assume the size of the B is $N_K \times N_K$, the complexity of the morphological opening is

$$C_{OPEN} = 2H \times W \times N_K^2 \quad (6)$$

where H and W represent the height and the width of the image.

Fig. 8(a) shows the enhancement accelerator architecture, which is composed of consecutive denoise accelerator, erosion and dilation accelerators. The erosion and the dilation are stencil computations calculating local minima and maxima within an $N_K \times N_K$ region. We designed the two-level reduction circuits shown in Fig. 8(b) and the systolic reduction tree shown in Fig. 8(c) for high computation performance and low hardware cost.

4.3 Trace Extraction Accelerator

The trace extraction accelerator is designed as a chain of tracing elements (TEs), each of which is tasked with computing the fluorescence value from multiple individual neurons, as Fig. 9 shows. As the image is streamed through the chain, all of the calcium traces are extracted based on the $N_C \times N_C$ templates stored locally at TEs according to Section 2.2. This tracing accelerator obviates the need for a separate image buffer, and avoids the inefficient off chip memory access. The computation time is hidden behind the image read out at 1 pixel/cycle throughput.

5 EVALUATION

We prototyped the proposed calcium image processing pipeline on the Ultra96 platform. We set N_K = 19 and N_C = 25, and we implemented 32 TEs inside the tracer chain, in which each TE traces 8

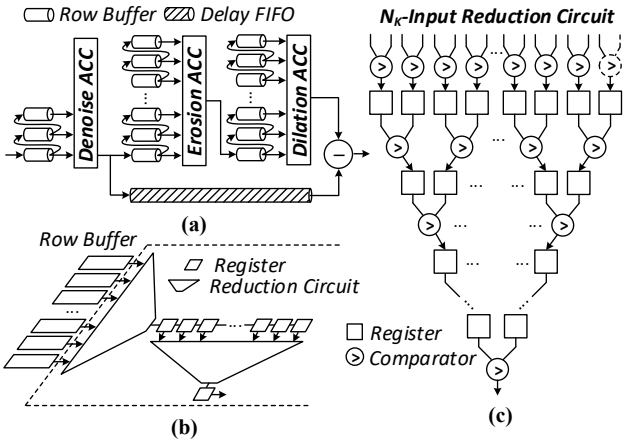

Figure 8: (a) Architecture of the enhancement accelerator. (b) Two-level reduction circuits. (c) Systolic reduction tree.

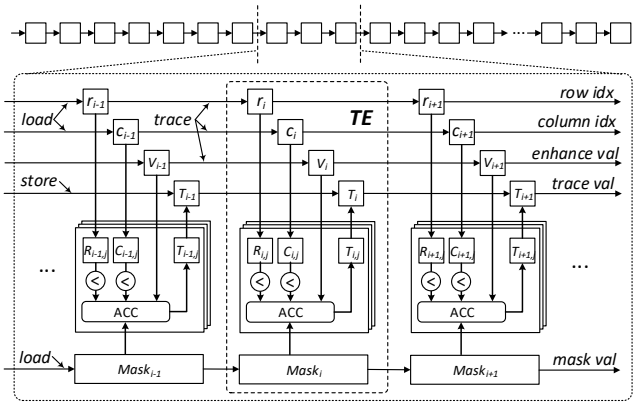

Figure 9: Architecture of the trace extraction accelerator.

different cells and the tracer chain is reused for 4 times for each frame. For the LSTM inference, we demonstrated 4 LSTM inference kernels that can be shared among the calcium trace extraction for all the cells. Fig. 10(a) shows a breakdown of the FPGA hardware resource utilization. We implemented the FPGA design in 300 MHz and compared its performance against the evaluation on the Xeon E5-2860 CPU using 12 threads. As Fig. 10(b) shows, the enhancement and the trace extraction accelerators achieve 18.1x and 2.4x speedup over the multicore CPU. Fig. 10(c) shows the runtime and the latency of executing the proposed pipeline on the FPGA. Since the motion correction and a large part of the enhancement can be overlapped with the image sensor read out, the latency can be reduced to 764 μs from the 2-ms runtime. The measured power consumption of the FPGA implementation is 7.2 W, which reduces the power consumption of the CPU by 88%.

We evaluated the performance and the energy efficiency of the BLINK circuits, and compared it against the evaluation on the Xeon E5-2860 CPU with 16 threads using the OpenMP and the evaluation on the V100 GPU using the CUDA. The BLINK circuits achieve 37.0

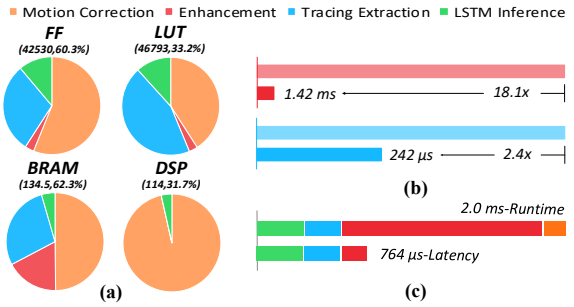

Figure 10: (a) FPGA hardware resource utilization, (b) the speedup over the evaluation on the multicore CPU, and (c) the performance evaluation on the Ultra96.

M inference/s performance and 410 pJ/inference energy efficiency, which outperforms the evaluation on the CPU by 10.4x in performance and outperforms the evaluations on the CPU and the GPU by 6293x and 52.4x gain in energy efficiency, respectively.

6 RELATED WORK

6.1 Efficient LSTM inference Accelerator

Various approaches have been proposed for the energy efficient LSTM/RNN inference accelerators [3, 4, 6, 7, 11, 13, 15–17]. [6] designed a low power LSTM accelerator for the keyword spotting under 5 µW with 60 nJ/inference energy efficiency. [15] realized the LSTM implementation in the spike domain, and mapped the LSTM inference onto the TrueNorth neuromorphic processors. [3] designed a compact LSTM inference kernel by fully pipelining the forward and recurrent paths. [4] took advantage of the temporal redundancy of the gated recurrent unit and enabled zero-skipping. [7] and [17] proposed pruning and structured compression to improve both the performance and the energy efficiency. [16] improved the temporal locality of the LSTM weights by separating forward and recurrent paths and increasing the reuse rate of the LSTM weights. [11] used the stochastic computing to improve the energy efficiency of the RNN inference. [13] leveraged the coarse grained parallelism by taking advantage of the column-wise multiplication. This paper proposes the bit-sparse data representation to simplify the multiplication into the bit shift, and it can be used in perpendicular with previous optimization strategies in improving the energy efficiency of the LSTM inference.

6.2 Efficient Online Calcium Image Processing

Most existing pipelines for the calcium image processing [12, 18] require the bulk image processing only suitable for the offline analysis. [18] achieved the real-time calcium image processing in terms of throughput, but it does not guarantee short processing latency and high energy efficiency for closed-loop feedback applications. [12] took advantage of the LSTM inference for the cell detection. This paper uses the LSTM inference to approximate the offline trace extraction with high accuracy. The proposed method enables high energy efficiency and short latency calcium image processing and is suitable for neurofeedback devices.

7 CONCLUSION

In this paper, we propose the bit-sparse data representation for the LSTM inference for the trace extraction from calcium images. This method improves the energy efficiency of the LSTM inference by turning the multiplication into the bit shift operation while remaining the inference accuracy. Based on this method, we propose a customized pipeline for the real-time calcium image processing, and it has the potential to be used as an energy-efficient single-device solution for closed-loop neurofeedback devices.

ACKNOWLEDGMENTS

This work is supported by the NSF under Grant No.: CCF-1436827 and No.:DBI-1707408. The authors would like to thank Prof. Golshani for leading the project and Dr. Daniel Aharoni for his support on the miniscope device.

REFERENCES

[1] Daniel Aharoni, Baljit S. Khakh, Alcino J. Silva, and et al. 2019. All the light that we can see: a new era in miniaturized microscopy. *Nat. Methods* 16, 1 (2019), 11–13.
[2] Zhe Chen, Hugh T. Blair, and Jason Cong. 2019. LANMC: LSTM-assisted non-rigid motion correction on FPGA for calcium image stabilization. In *Proc. Int. Symp. Field-Programmable Gate Arrays (FPGA '19)*. ACM, New York, NY, USA, 104–109.
[3] Zhe Chen, Andrew Howe, Hugh T. Blair, and et al. 2018. CLINK: Compact LSTM inference kernel for energy efficient neurofeedback devices. In *Proc. Int. Symp. Low Power Electron. Des.* ACM, New York, NY, USA, 2:1–2:6.
[4] Chang Gao, Daniel Neil, Enea Ceolini, and et al. 2018. DeltaRNN: A power-efficient recurrent neural network accelerator. In *Proc. Int. Symp. Field-Programmable Gate Arrays (FPGA '18)*. ACM, 21–30.
[5] Andrea Giovannucci, Johannes Friedrich, Pat Gunn, and et al. 2019. CaImAn an open source tool for scalable calcium imaging data analysis. *eLife* 8 (jan 2019), e38173.
[6] J. S. P. Giraldoand and Marian Verhelst. 2018. Laika: A 5µW programmable LSTM accelerator for always-on keyword spotting in 65nm CMOS. In *Proc. IEEE European Solid State Circuits Conf. (ESSCIRC '18)*. 166–169.
[7] Song Han, Junlong Kang, Huizi Mao, and et al. 2017. ESE: Efficient speech recognition engine with sparse LSTM on FPGA. In *Proc. Int. Symp. Field-Programmable Gate Arrays (FPGA '17)*. ACM, New York, NY, USA, 75–84.
[8] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Comput.* 9, 8 (nov 1997), 1735–1780.
[9] Yangqing Jia, Evan Shelhamer, Jeff Donahue, and et al. 2014. Caffe: Convolutional architecture for fast feature embedding. In *Proc. ACM Int. Conf. Multimedia (MM '14)*. 675–678.
[10] Abbas Kazemipour, Ondrej Novak, Daniel Flickinger, and et al. 2019. Kilohertz frame-rate two-photon tomography. *Nat. Methods* 16, 8 (2019), 778–786.
[11] Yidong Liu, Leibo Liu, Fabrizio Lombardi, and Jie Han. 2019. An energy-efficient and noise-tolerant recurrent neural network using stochastic computing. *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.* 27, 9 (2019), 2213–2221.
[12] Jinghao Lu, Chunyuan Li, Jonnathan Singh-Alvarado, and et al. 2018. MIN1PIPE: A miniscope 1-photon-based calcium imaging signal extraction pipeline. *Cell Rep.* 23, 12 (jun 2018), 3673–3684.
[13] Zhiqiang Que, Hiroki Nakahara, Eriko Nurvitadhi, and et al. 2020. Optimizing Reconfigurable Recurrent Neural Networks. In *Proc. IEEE Int. Symp. Field-Programmable Custom Computing Machines (FCCM '20)*. IEEE, 10–18.
[14] William M Reichert. 2007. *Indwelling neural implants: Strategies for contending with the In Vivo environment.*
[15] Amar Shrestha, Khadeer Ahmed, Yanzhi Wang, and et al. 2018. Modular spiking neural circuits for mapping long short-term memory on a neurosynaptic processor. *IEEE Trans. Emerg. Sel. Topics Circuits Syst.* 8, 4 (dec 2018), 782–795.
[16] Franyell Silfa, Gem Dot, Jose-Maria Arnau, and Antonio Gonzàlez. 2018. E-PUR: An energy-efficient processing unit for recurrent neural networks. In *Proc. Int. Conf. Parallel Architectures and Compilation Techniques (PACT '18)*. ACM, 18:1–18:12.
[17] Shuo Wang, Zhe Li, Caiwen Ding, and et al. 2018. C-LSTM: Enabling efficient LSTM using structured compression techniques on FPGAs. In *Proc. Int. Symp. Field-Programmable Gate Arrays (FPGA '18)*. ACM, New York, NY, USA, 11–20.
[18] Pengcheng Zhou, Shanna L Resendez, Jose Rodriguez-Romaguera, and et al. 2018. Efficient and accurate extraction of in vivo calcium signals from microendoscopic video data. *eLife* 7 (feb 2018), e28728.