Effective Sensor Fusion with Event-Based Sensors and Deep Network Architectures

Daniel Neil and Shih-Chii Liu
Institute of Neuroinformatics, University of Zurich and ETH Zurich
Winterthurerstrasse 190, CH-8057 Zurich, Switzerland

***Abstract*—The use of spiking neuromorphic sensors with state-of-art deep networks is currently an active area of research. Still relatively unexplored are the pre-processing steps needed to transform spikes from these sensors and the types of network architectures that can produce high-accuracy performance using these sensors. This paper discusses several methods for pre-processing the spiking data from these sensors for use with various deep network architectures. The outputs of these pre-processing methods are evaluated using different networks including a deep fusion network composed of Convolutional Neural Networks and Recurrent Neural Networks, to jointly solve a recognition task using the MNIST (visual) and TIDIGITS (audio) benchmark datasets. With only 1000 visual input spikes from a spiking hardware retina, the classification accuracy of 64.5% achieved by a particular trained fusion network increases to 98.31% when combined with inputs from a spiking hardware cochlea.**

Keywords*—Event-Driven Sensors, Deep Networks, Recurrent Neural Networks, Dynamic Vision Sensor, Sensor Fusion.

I. Introduction

In recent years, increasing work has gone towards interfacing event-based sensors, in particular the Dynamic Vision Sensor (DVS) [1], to deep networks such as Deep Belief Networks (DBNs) and Convolutional Neural Networks (CNNs). The benefits of these networks including robustness to noise, bit precision of the hardware platform, and efficient processing, have been demonstrated in some studies [2], [3]. Although deep networks are used heavily in the machine learning community [4], their use in the neuromorphic field is still in the early stages due to the dramatic differences in the nature of the continuous-time event-driven input data and the frame-based data that machine learning typically uses.

The combination of spiking deep networks together with event-based sensors has been considered in previous studies, for a example, through a spiking CNN receiving DVS spikes [5], [6]. Spiking deep networks have also been implemented in hardware, for example, a spiking DBN was implemented on a hardware platform and interfaced to a DVS [7], [8].

Because of recent spiking network conversion methods that show how frame-based deep networks such as CNNs and fully-connected networks can be trained so that the converted spiking network has a classification performance that is almost equivalent to that of the analog network [9], the performance of spiking networks together with spiking inputs can be compared more easily to the performance of the trained networks using frame-based input.

What is still relatively unexplored are the pre-processing steps that are useful for the outputs of the event-based sensors, in particular the audio sensor; and other new deep network architectures such as RNNs which are better suited for temporal sequences.

This work extends the previous contributions in three ways. First, it presents several methods for preprocessing spiking data from event-based sensors for use with state-of-the-art deep network architectures. Second, it demonstrates the use of architectures such as deep CNNs [4] and deep RNNs [10] as powerful classifiers for event input streams. Third, the accuracy of a sensor fusion deep network is quantitatively evaluated. Multimodal fusion with deep neural networks have been demonstrated with event-based sensors, for e.g. a spiking DBN was successfully used to fuse visual spikes from a DVS together with audio spikes from a Dynamic Audio Sensor (DAS) cochlea [3], [11]. However, the audio stimuli consisted of pure tones and a quantification of the network classification accuracy using the sensors was not performed in the first study.

This work focuses only on CNNs for processing the visual event stream because they currently produce state-of-the-art performance in visual tasks [12]. The audio input is processed by either CNNs or RNNs, both which are currently used in audio classification tasks. The performance of these networks is tested on two standard benchmark datasets: the MNIST handwritten digit recognition dataset and the TIDIGITS audio dataset. In addition, the corresponding spike databases consisting of recordings through the DVS on MNIST (N-MNIST) [13], and recordings through the DAS cochlea on TIDIGITS [14] are used in the evaluation of the networks. Section II describes the data processing methods for the event-based sensors as well as the deep network architectures used. Section III presents classification results using visual and audio input representations and Section IV discusses the findings.

II. Methods

A. Input data processing methods

The primary goal of the data processing methods is to produce a frame-based representation of the event stream from the spiking sensors compatible with the input format needed for training deep networks and allowing the subsequently trained networks to be tested with non-framed event streams.

This work was partially supported by the Swiss National Science Foundation grant #200020-153565 "Fast Separation of Auditory Sounds," EU H2020 COCOHA #644732, and the Samsung Advanced Institute of Technology.

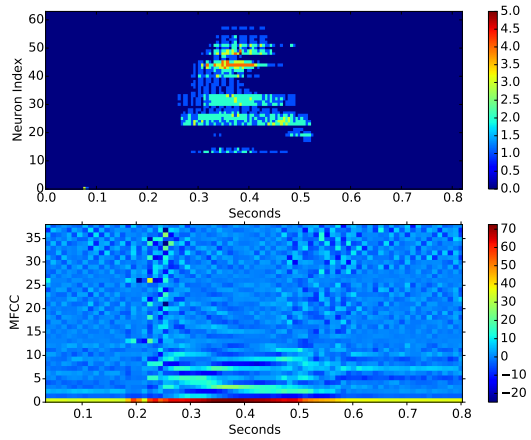

Fig. 1. (**Top**) Cochleagram representation of a spoken digit from [14]. (**Bottom**) MFCC representation of a spoken digit, with first and second derivatives as well as power.

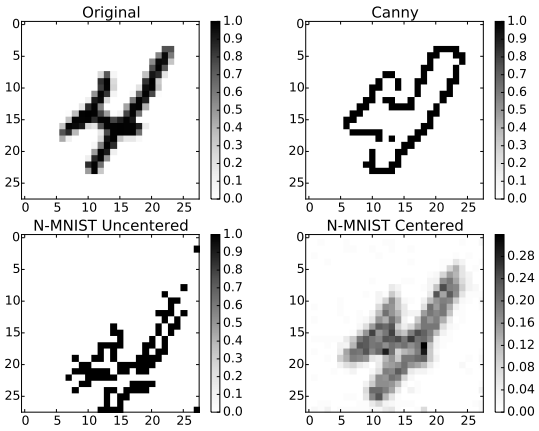

Fig. 2. Visual transformations on MNIST. (**Upper Left**) Original image. (**Upper Right**) Same digit after Canny filtering. (**Lower Left**) Raw, unstabilized N-MNIST dataset (100 events). (**Lower Right**) Stabilized N-MNIST dataset (full dataset).

In machine learning, the deep network architectures and their training methods have been developed on the assumption that the data is primarily static. This scenario differs from the field of event-based research where huge databases are not yet available and the data is always dynamic and event-driven in nature.

1) Audio Network Input: Single digits ("oh" and zero through nine) from the TIDIGITs database are used in our study with a total of 2464 digits in the training set and 2486 digits in the test set.

Spikes are pre-recorded from the DAS cochlea system [15] in response to the digits in this database. The DAS system holds a custom AEREAR2 binaural silicon cochlea chip. Each cochlea consists of a 64-stage cascaded second-order filter bank (frequency channels) followed by a half-wave rectifier modeling the inner hair cell, and an integrate-and-fire neuron which models spiral ganglion cells. The 64 frequency channels have individual characteristic frequency selectivity ranging from 100 Hz to about 10 kHz on a log frequency scale. By binning the DAS channel spike outputs into 5 ms time bins, the resulting 2D histogram of frequency channels versus time bins (cochleagram) can be interpreted as an image for a CNN, or read time slice-by-time-slice into a RNN. An example can be seen in Fig. 1.

Mel-Frequency Cepstral Coefficient (MFCC) features are often used in state-of-the-art audio processing networks, and are used as a comparison here. Each audio digit waveform is preprocessed using a 25 ms window, 10 ms frame shift, and 20 filterbank channels to produce 12 cepstral coefficients which are concatenated with the overall power and the first and second derivatives to form a 39-dimensional feature vector.

2) Visual Network Input: The benchmark MNIST dataset consisting of 60,000 28x28 handwritten digits in greyscale and a test set of 10,000 digits, is used in the visual classification task. In previous investigations [3], [7], [9], the original image is treated as a rate-based approximation of the visual sensor event stream during training. During the testing phase, spikes are drawn from the image with a probability proportional to the intensity of the pixel. With an increasing number of spikes, binning them produces an image similar to the original input.

However, the event stream from the DVS is produced by changes in the temporal contrast at each pixel, which does not have the same statistics as the spikes generated according to the intensity value. The approach taken in collecting the N-MNIST dataset [13] was to present each digit to a DVS while moving the image sensor in a controlled way. The N-MNIST dataset includes a script to counter the movement of the image sensor so that the triggered events can also be centered in the original image position. In this work, both the unstabilized and stabilized spike data versions are considered. The unstabilized version is more relevant for real-world stimuli where the events cannot easily be back-projected to counter the movement of the sensor, while the stabilized version is more similar to standard machine learning inputs.

Since the binned unstabilized N-MINST digit spikes in Fig. 2, lower left, resembles an edge-filtered image, the Canny edge filter [16] was applied to the static images before training to see if this filtering would improve recognition performance. See Fig. 2, upper right, for an example of a filtered image.

B. Deep network architectures for event-based sensors

The network architectures used in this work are presented in Fig. 3. These networks were trained using the Keras [17] Python and Theano-based deep learning software. The code to train and test these networks is available online.[1] The networks constructed as shown in Fig. 3, were trained using the Adam [18] stochastic optimization method for 15-20 epochs.

1) Convolutional Neural Networks: Convolutional Neural Networks (CNNs) are used here to process both visual and audio inputs (see Fig. 3). For the visual input, the N-MNIST image size of 36x36 was not resized; instead, inputs that fall

[1] https://github.com/dannyneil/sensor_fusion_iscas_2016

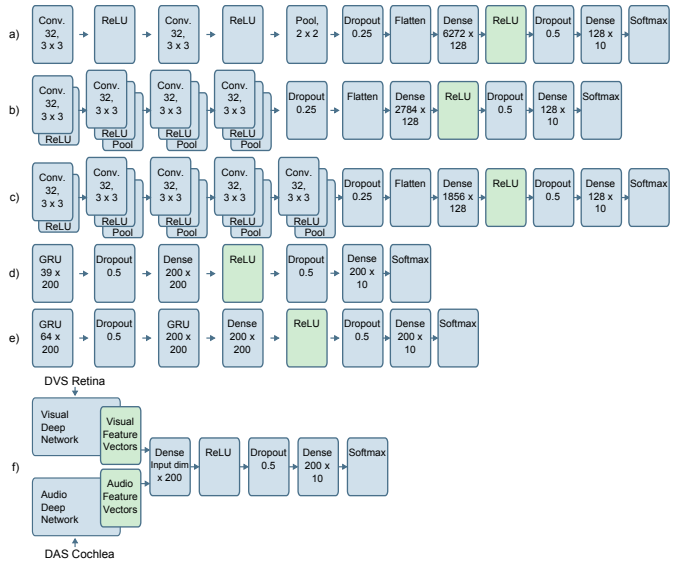

Fig. 3. Network types. (**a**) CNN for visual input. (**b**) CNN for MFCC audio inputs. The stacked layers are the same as in (a), collapsed here for readability. (**c**) CNN for cochleagram audio inputs. (**d**) Deep RNN for audio classification on MFCCs. (**e**) Deep RNN for audio classification on cochleagrams. (**f**) Fusion architecture, which fuses the feature vector layers (highlighted in green) to produce a joint classification.

outside of the 28x28 region were ignored as this method gave the best results.

Because CNNs expect a fixed-size input, each audio digit's features had to be pre-padded with zeros to equalize their lengths. Because the MFCC time length (10 ms shifts) of a digit was shorter than the time dimension of the cochleagram (5 ms bins), the network architectures using MFCC inputs and the cochleagram differ slightly. The network that processes the cochleagram includes an additional layer of convolution, activation, and pooling, to decrease the hidden layer dimensionality before the last fully-connected layers. These convolutional networks were trained using the method described in [9] to produce networks capable of being converted to spiking networks, although they were used in their nonspiking form.

2) Recurrent Neural Networks: The deep RNNs are composed of gated recurrent units as introduced in [10]. At each time step, an input feature (either the MFCC or the cochleagram) as computed over a time window was used as the input to the RNN. Note that this input presentation is purely causal, unlike bidirectional RNN models that operate forward and backward on the input data at the same time. This causal presentation was chosen to ensure compatibility with the real-time implementation in which the binned spikes are summed and propagated to a RNN in an online setting.

C. Architecture of Fusion Network

The architecture of the fusion network in Fig. 3(f) includes a visual feature input layer from the network in (a) and an audio feature input layer from one of the networks in (b) to (e). These feature layers are highlighted in green in the different networks. The rate-based continuous-valued outputs of these layers go to two non-spiking fully-connected layers. Each neuron in the first fully-connected layer can connect to either modality or to both, and performs a nonlinear combination of the features produced from each modality.

TABLE I. NETWORK PERFORMANCE SUMMARY, TRAIN AND TEST FRAME-BASED

Network Type	Classification Accuracy
Visual CNN (Intensity)	99.26%
Visual CNN (Canny)	97.52%
Visual CNN (N-MNIST)	98.30%
Audio MFCC CNN	95.86%
Audio MFCC RNN	96.10%
Audio Cochlea CNN	87.65%
Audio Cochlea RNN	82.82%

III. RESULTS

A. Individual Network Performance

The classification accuracy of the different non-spiking networks which are both trained and tested on frame-based inputs, can be found in Table I. Their accuracies are viewed as the ideal target accuracies for the spiking networks. The first three networks are trained on visual data of three corresponding different input types: 1) a network trained to identify spikes drawn from the intensity of the pixels ("Intensity"), 2) a network trained to identify spikes from edges ("Canny"), and 3) a network trained on the summed spikes over the duration of the test from the real N-MNIST data ("N-MNIST"). These three networks all achieve high accuracy in their own test classification. The next four networks were trained on audio input. The first two networks were trained on MFCC features, and the remaining two were trained on cochleagrams generated from spikes of the DAS sensor. The network accuracies show that processing the cochlear spikes is more challenging than using the MFCCs of the audio, but overall classification accuracy is still quite high. Indeed the recognition accuracy of many of these networks (Intensity, Canny, N-MNIST, and Cochleagram CNN) establishes a new state-of-the-art benchmark ([9], [13]).

B. Approximation Methods for Visual Input

In the second set of experiments, the frame-based networks were converted to spiking networks as described in [9] and these networks were then tested on spiking inputs. Table II compares the performance of these networks (columns) on spike inputs generated in different ways (rows). The "N-MNIST (uncentered, 1k)" row refers to uncentered N-MNIST data using the first thousand events; "N-MNIST (centered, 1k)" refers to the centered N-MNIST data using the first thousand events; and "N-MNIST (centered, full)" refers to the full centered N-MNIST dataset. The intensity-trained model performed well on the event-based intensity data, but suffered significant losses when using the realistic N-MNIST DVS data. Surprisingly, the use of the Canny-filtered data during training only slightly improved the recognition accuracy on the N-MNIST (1k) dataset and performed worse on the N-MNIST (full) dataset; using an N-MNIST-trained network was best.

C. Performance of the Fusion Network

In the third experiment, the visual and audio inputs were fused using the networks in Fig. 3 to produce the results in Table III. Even with relatively low individual sensor classification

TABLE II. COMPARISON OF VISUAL SPIKING METHODS

	Trained Networks		
Spiking inputs from:	**Intensity (99.26%)**	**Canny (97.52%)**	**N-MNIST (98.30%)**
Intensity	99.17%	–	–
Canny	82.51%	96.60%	–
N-MNIST (uncentered, 1k)	22.88%	23.17%	47.50%
N-MNIST (centered, 1k)	42.74%	43.51%	64.50%
N-MNIST (centered, full)	74.79%	57.71%	95.72%

TABLE III. SUMMARY OF FUSION PERFORMANCE

	CNN (99.26%)	**N-MNIST 1k Spikes (64.50%)**	**N-MNIST Full (95.72%)**
MFCC CNN (95.86%)	99.96%	99.19%	99.88%
MFCC RNN (96.10%)	99.94%	99.40%	99.83%
Cochlea CNN (87.65%)	99.67%	97.40%	99.64%
Cochlea RNN (82.82%)	99.86%	**98.31%**	**99.66%**

accuracies, fusing streams from multiple modalities increased accuracy dramatically. In all combinations of visual and audio inputs, accuracy improved when both modalities were used. The joint score of a CNN trained on N-MNIST with an RNN trained on cochleagrams yielded an impressive 99.66% classification accuracy. Moreover, while a brief thousand input visual spikes caused the network to achieve a poor 64.5% accuracy, when combined with audio, this same network always achieved performance greater than 97%.

IV. DISCUSSION

This work examined data preprocessing methods and deep networks for fusing multimodal spiking sensor inputs. It establishes new state-of-the-art classification accuracy numbers on event-based sensor datasets, and proposes architectures as starting points for future work. For audio, the novel use of deep RNNs and CNNs improved the classification accuracies significantly for the spiking audio sensor.

For visual inputs, training with Canny-filtered images only modestly improved the classification accuracy on the N-MNIST dataset, although preliminary work suggests that it may help greatly on alternative datasets especially in conjunction with further data augmentation such as translation, scaling, and rotation. Visual networks face a tradeoff; additional input spikes create new output spikes, but too many input spikes smear the input image over time as in the case of the N-MNIST (full) dataset. In this case, the smeared binned spike image led to a lower accuracy number (see Table II) for the Canny-filtered trained network versus the Intensity network.

Finally, fusion network architectures like those presented in Fig. 3(f) demonstrate how networks can overcome limitations in input modalities to achieve extremely accurate performance. As no single modality is ideal and free of noise, a network trained to fuse a representation from different modalities is inherently robust. If the error sources themselves are decorrelated, then the joint probability of an error in both streams is the product of the two error rates. This dramatically decreases the number of errors, as shown by the results in Table III, and permits extremely accurate classification even with very noisy input streams.

ACKNOWLEDGMENTS

We acknowledge Michael Purghart and Maria Karapetkova for their explorations in this work.

REFERENCES

[1] P. Lichtsteiner, C. Posch, and T. Delbruck, "A 128× 128 120 db 15 μs latency asynchronous temporal contrast vision sensor," *IEEE Journal of Solid-State Circuits*, vol. 43, no. 2, pp. 566–576, 2008.

[2] E. Stromatias, D. Neil, M. Pfeiffer, F. Galluppi, S. B. Furber, and S.-C. Liu, "Robustness of spiking Deep Belief Networks to noise and reduced bit precision of neuro-inspired hardware platforms," *Frontiers in Neuroscience*, vol. 9, 2015.

[3] P. O'Connor, D. Neil, S.-C. Liu, T. Delbruck, and M. Pfeiffer, "Real-time classification and sensor fusion with a spiking Deep Belief Network," *Frontiers in Neuroscience*, vol. 7, 2013.

[4] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[5] C. Farabet *et al.*, "Comparison between frame-constrained fix-pixel-value and frame-free spiking-dynamic-pixel convnets for visual processing," *Frontiers in Neuroscience*, vol. 6, 2012.

[6] J. Pérez-Carrasco and others., "Mapping from frame-driven to frame-free event-driven vision systems by low-rate rate coding and coincidence processing–Application to feedforward ConvNets," *IEEE Trans on Pattern Analysis and Machine Intelligence*, vol. 35, no. 11, pp. 2706–2719, 2013.

[7] D. Neil and S.-C. Liu, "Minitaur, an event-driven FPGA-based spiking network accelerator," *IEEE Trans on Very Large Scale Integration (VLSI) Systems*, vol. 22, no. 12, pp. 2621–2628, 2014.

[8] E. Stromatias, D. Neil, F. Galluppi, M. Pfeiffer, S.-C. Liu, and S. Furber, "Event-driven deep neural network hardware system for sensor fusion," in *2015 IEEE International Symposium on Circuits and Systems (ISCAS)*, 2015, pp. 1901–1901.

[9] P. U. Diehl, D. Neil, J. Binas, M. Cook, S.-C. Liu, and M. Pfeiffer, "Fast-classifying, high-accuracy spiking deep networks through weight and threshold balancing," in *International Joint Conference on Neural Networks (IJCNN)*, 2015.

[10] K. Cho, B. van Merrienboer, C. Gulcehre, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using RNN encoder-decoder for statistical machine translation," *arXiv preprint arXiv:1406.1078*, 2014.

[11] I. Kiselev, D. Neil, and S.-C. Liu, "Event-driven deep neural network hardware system for sensor fusion," in *2016 IEEE International Symposium on Circuits and Systems (ISCAS)*, 2016.

[12] A. S. Razavian, H. Azizpour, J. Sullivan, and S. Carlsson, "CNN features off-the-shelf: an astounding baseline for recognition," in *Computer Vision and Pattern Recognition Workshops (CVPRW), 2014 IEEE Conference on*. IEEE, 2014, pp. 512–519.

[13] G. Orchard, A. Jayawant, G. Cohen, and N. Thakor, "Converting static image datasets to spiking neuromorphic datasets Using saccades," *arXiv: 1507.07629*, 2015. [Online]. Available: http://arxiv.org/abs/1507.07629

[14] A. Zai, S. Bhargava, N. Mesgarani, and S.-C. Liu, "Reconstruction of audio waveforms from spike trains of artificial cochlea models," *Frontiers of Neuromorphic Engineering: Special Issue on Benchmarks and Challenges in Neuromorphic Engineering*, 2015.

[15] S.-C. Liu, A. van Schaik, B. Minch, and T. Delbrück, "Asynchronous binaural spatial audition sensor with $2 \times 64 \times 4$ channel output," *IEEE Trans. Biomed. Circuits Syst.*, vol. 8, no. 4, pp. 453–464, 2014.

[16] J. Canny, "A computational approach to edge detection," *IEEE Transactions onPattern Analysis and Machine Intelligence*, vol. PAMI-8, no. 6, pp. 679–698, 1986.

[17] F. Chollet, "Keras," https://github.com/fchollet/keras, 2015.

[18] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.