Musical notes classification with Neuromorphic Auditory System using FPGA and a Convolutional Spiking Network

E. Cerezuela-Escudero, A. Jimenez-Fernandez, R. Paz-Vicente, M. Dominguez-Morales, A. Linares-Barranco, G.Jimenez-Moreno
Department of Computer Architecture and Technology
University of Seville
Seville, Spain
Email: {ecerezuela,ajimenez, rpaz, mdominguez, alinares, gaji}@atc.us.es

***Abstract*—In this paper, we explore the capabilities of a sound classification system that combines both a novel FPGA cochlear model implementation and a bio-inspired technique based on a trained convolutional spiking network. The neuromorphic auditory system that is used in this work produces a form of representation that is analogous to the spike outputs of the biological cochlea. The auditory system has been developed using a set of spike-based processing building blocks in the frequency domain. They form a set of band pass filters in the spike-domain that splits the audio information in 128 frequency channels, 64 for each of two audio sources. Address Event Representation (AER) is used to communicate the auditory system with the convolutional spiking network. A layer of convolutional spiking network is developed and trained on a computer with the ability to detect two kinds of sound: artificial pure tones in the presence of white noise and electronic musical notes. After the training process, the presented system is able to distinguish the different sounds in real-time, even in the presence of white noise.**

Keywords—musical note recognition, convolutional spiking network, neuromorphic auditory hardware, Address-Event Representation.

I. INTRODUCTION

The sound pitch estimation is necessary for a variety of applications in different fields, e.g. music transcription [1]-[5], instrument recognition [6]-[8] or speech and voice recognition [9], [10]. Pitch estimation is usually treated as a two stage problem: filtering and classification. Filtering is the stage where the signal is processed so only relevant information will pass to the classification stage, where the sound will be identified. One of the two main factors that make pitch estimation a hard task is noise because it can be even louder than the original signal, and, in musical note recognition systems, the other factor is the time estimation of individual musical notes. In this work, to solve the two-stage problem (filtering and classification), we use neuromorphic systems, because of their high level of parallelism, interconnectivity, and scalability, which allow them to carry out complex processing in real time, with a good relation between quality, speed and resource consumption. The signals in these systems are composed of short pulses in time, called spikes or events. The information can be coded in the polarity and spike frequency, often following a Pulse Frequency Modulation (PFM) scheme, or in the inter-spike-interval time (ISI) [11], or in the time-from-reset, where the most important (with the highest priority) events are sent first [12]. Address-Event Representation (AER), proposed by Mead lab in 1991 [13], faced the difficult problem of connecting silicon neurons along chips that implement different neuronal layers using a common asynchronous digital bus multiplexed in time, the AER bus. This representation gives a digital unique code (address) to each neuron, which is transmitted using a simple four-phase handshake protocol [14].

This work is focused on implementing a trained convolutional layer for real-time musical sounds classification suitable for hardware implementation where audio information comes from a Neuromorphic Auditory System (NAS). This NAS (available for FPGA at www.rtc.us.es) is a spike-based set of band-pass filters that decomposes an audio signal into different frequency bands of spiking information, in the same way a biological cochlea sends the audio information to the brain. Analog audio information is digitalized and converted into a PFM signal to be fed to the NAS. The convolutional layer will look for particular spiking patterns along the different frequency channels.

There are highlighted AER systems in the area of artificial audio like [8], [15]-[18] and highlighted convolutional neural networks for performing object recognition in the field of artificial vision, as can be shown in several works [19]-[22], but there is no study in the literature to this day that used convolutional neural network for audio recognition. The new contributions of this work to the Neuromorphic community over those previous studies are: 1) the use of a neuromorphic auditory system for the stage of extraction of acoustic features (this NAS is innovative because its filters completely work with spiking information in a digital and synchronous way (FPGAs), in a real-time environment), and 2) the novel use of an AER convolutional network for real-time sounds classification.

In order to compare our results with previous recognition sound systems, we present a summary of some pitch detection methods. There are several proposals to recognize the musical audio pitch, some of which estimate the fundamental frequency

This work has been supported by the Spanish grant (with support from the European Regional Development Fund) BIOSENSE (TEC2012-37868-C04-02)

[23]-[25],[3] and others classify the sounds [2], [4], [5], [26], [27]. Reference [15] proposes a sensory-fusion system that uses a Dynamic Vision Sensor silicon retina [28], silicon cochlea [16] and deep belief network for written digit classification. In this system a pure tone is associated to each digit. We cannot compare the proposed system in [15] with this work because [15] does not show the results without the sensory fusion. In [3] the fundamental frequency is estimated with the autocorrelation method and the result of implementation attained a hit rate (percentage of successes) of 96.0% for plunking points of six guitar strings. The recognition method proposed in [2] transcribes guitar chords by a multiple fundamental frequency estimator and hidden Markov model. The accuracy of the system is 95% for 48 possibilities. Reference [27] proposes an algorithm to detect the diatonic pitch from the audio (Major and Minor classification). The algorithm achieves 92% correct detection of key signature for the popular pieces and 81% for the classical pieces. In [4] the Fast Fourier Transform and the Harmonic Product Spectrum are proposed for the filtering stage and a Multi-Layer Perceptron for the classification stage. The system achieved 97.5% recognition accuracy for 12 notes using 20 neurons in the first hidden layer and 10 neurons in the second one.

II. NEUROMORPHIC AUDITORY SYSTEM

Audio information acquisition has been made by a novel digital cochlear model implementation fully based on PFM, Neuromorphic Auditory System (NAS). Previous digital cochleae process audio signals using classical Digital Signal Processing (DSP) techniques. On the contrary, NAS processes information directly encoded as spikes with a Pulse Frequency Modulation (PFM), performing Spike Signal Processing (SSP) techniques [29], [30], and using AER interfaces. The architecture of the NAS is shown in Fig. 1. The general system is composed of two digitalized audio streams, which represent the left and right audio signals of a stereo system. Two Synthetic Spike Generators [31] convert these audio digital sources into spike streams. Then, the left and right NAS Filter Banks (NFB) split the two spike streams in 2N (N is the number of the channels of each NAS) frequency bands using 2N different spiking outputs that are combined by an AER monitor block into an AER output bus [32], which encodes each spike according to AER and transmits this information to the recognition system. A NFB is composed of a set of spike-based low-pass filters (SLPF) [29] connected in cascade, as many as the number of channels that are implemented for a particular auditory application. As in previous implementations of AER cochlea [16]-[18], there are several stages connected in cascade, subtracting the spike information of consecutive SLPF output spikes in order to reject out-of-band frequencies, obtaining a response equivalent to a band-pass filter. All the elements required for designing the NAS components (Synthetic Spike Generators, NFB and the AER monitor) have been implemented in VHDL and designed as small spike-based building blocks [30]. Each of them performs a specific operation on spike streams and can be combined with other blocks in order to build complex spike-processing systems. This kind of system has been used before, for example, in closed-loop spike-based PID controllers [33] and trajectory generators for object tracking [34].

In this work, the NAS has been designed with 64 channels in each NFB because this is the highest number of channels that can be synthesized into the selected FPGA for this work (Xilinx Virtex-5 FXT FPGA ML507) [35]. Table I shows the NAS requirements. The 18 I/O signals are: 1 for clock signal, 1 for reset signal, 6 for AC-link [36], 2 for AER control and 8 for AER address, providing an AER space of 256 addresses (0 to 127 for the left cochlea, and 128 to 255 for the right cochlea including polarity bit). It should be remarked that this NAS implementation does not demand specialized FPGA resources (e.g.: multipliers, embedded processors, DSP); it only requires common digital logic components (counters, comparators, adders and registers), working with a low number of bits. In order to show its frequency response, the NAS was stimulated with an audio frequency sweep from 10Hz to 22kHz and $1V_{RMS}$. Fig. 2 shows the frequency response of the 64 channel of the left NAS (bode diagram), where the spike rate (Y-axis) is plotted as a function of the sound frequency (X-axis) for each of the left NFB channels (diverse colors). This figure denotes how in general the cochlea channels behave like a set of band-pass filters, rejecting out of band audio tones. This figure also shows that the spike rate of the band-pass filters of the NFB channels decrease with lower frequencies, and the higher frequency channels present the addition of an offset. The highest frequency channel has a mid-frequency of 14.06kHz, and the lowest frequency channel has a mid-frequency of 9.6Hz, getting a global cochlear equivalent bandwidth greater than 14 kHz. The frequency response of the 64 channel of the right NAS is not shown because it is similar to the left one.

TABLE I. STEREO NAS HARDWARE REQUERIMENTS

Slices / Utilization	Max. clock (MHz)	Power (mW)	I/O Signals
11,141/ 99.47%	87.31 (we used 27)	29.7	18

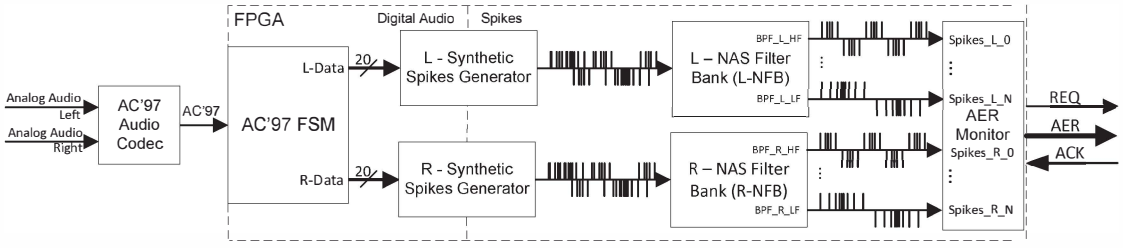

Fig. 1. NAS Architecture

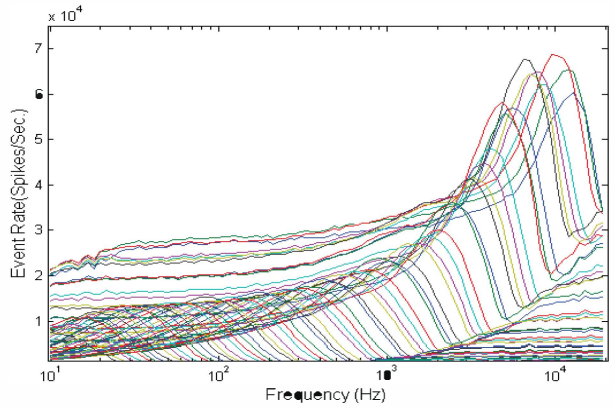

Fig. 2. Frequency response of the left NAS stimulated with audio frequency sweep. The stimulatied sounds have 1V_{RMS} amplitude with a varying frequency from 10Hz to 22kHz. The X-axis represents the sound frequency, the Y-axis is the spike rate and the different colors represent the response of each channel.

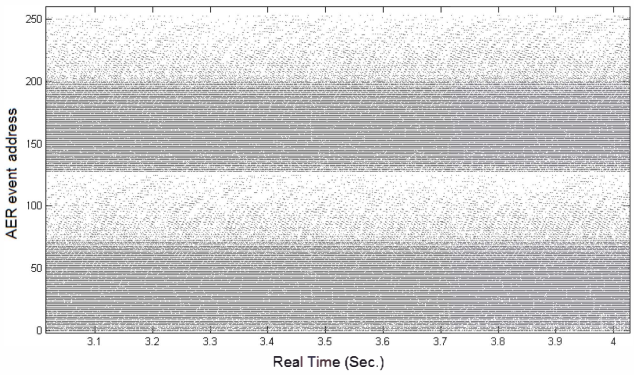

Fig. 3. Experimental cochleogram of a 64 channel stereo NAS corresponding to the electronic piano note A4

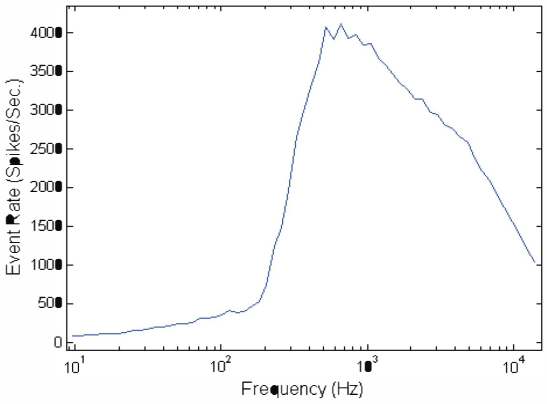

Fig. 4. Left NAS frequency response corresponding to the electronic piano note A4

Fig. 3 shows the output cochleogram in the presence of an electronic piano with note A4 (440Hz). In this figure, the X-axis represents time and the Y-axis represents the AER address. The NAS uses 256 addresses because it has 64 channels for each audio source and each channel can fire positive and negative spikes. Therefore, every time a specific event appears, it is represented in this figure by a dot in its correct position. Fig. 3 bottom (addresses from 0 to 127) shows the left source activity and Fig. 3 top (addresses from 128 to 255) shows the right one. In general, both outputs present a specific output delay due to the NFB cascade architecture. Fig. 4 shows the frequency response of the left NAS in the presence of an electronic piano with note A4 (440Hz), where the Y-axis represents the event rate as a function of the channel frequencies (X-axis).

The NAS has been used before as a rotation frequency sensor for an industrial motor [37], and it has also been used in a classificatory system that can distinguish different cricket species by their chirping [38].

III. Convolutional Spiking Network Architecture

In order to recognize musical sounds, a pattern recognition approach based on convolutions is used in this work. To achieve this goal, every time an event is received from the NAS, a one-dimensional convolution kernel is applied in the following way: a memory array stores an array of neuron membrane potentials. The memory array has the same number of components as audio channels (frequencies). Each incoming event-address represents the center of the one-dimensional kernel. The result of the convolution is stored back in memory if the value is below a threshold. Otherwise, the value in memory is reset and an output event is sent out. The one-dimensional convolution for one neuron of our recognition system is defined mathematically by (1), where x is a clock cycle, W is the kernel of the convolution, S is the output of each channel of the cochlea (the input of our recognition system) and Y is the convolved output. Equation (2) shows the output produced by a generated event. In these equations, the M length is determined by the number of channels that the FPGA cochlea has. M length is 128, because a 64-binaural NAS has been used in these experiments. In order to take advantage of the binaural cochlea strengths, two microphones have been used, and the output of both cochleae has been taken into account. Therefore, we have a double kernel size and twice the input values to obtain the output of the recognition system.

$$Y(x+1) = Y(x) + \sum_{m=0}^{M}(W(m) * S(x)) \qquad (1)$$

$$Out = Y(x) \geq \theta \rightarrow Y(x) = 0 \qquad (2)$$

The number of neurons implemented in the recognition system depends on the number of sounds that are going to be classified. The kernel values ($W(m)$) have been obtained from the normalized frequency value of each channel output during a sound playback. The values are normalized, in range [0,1], in order to get a volume-independent recognition system. Fig. 5 shows the values of the kernel obtained after the playback of four electronic piano notes: F3, F4, F5 and F6. The X-axis represents channels of the left NAS and the Y-axis represents the channel normalized event rate from each note. These values are used as the convolution kernel.

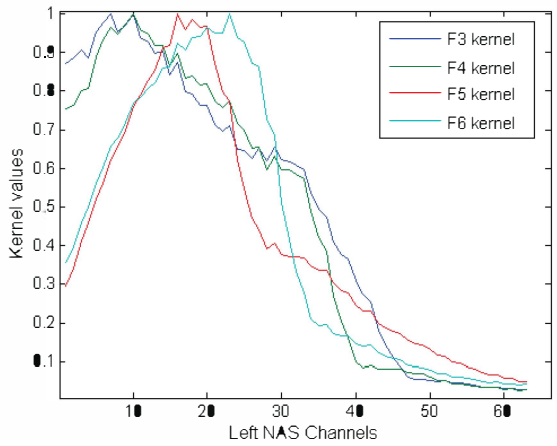

Fig 5. Kernel values from the left NAS output with electronic piano notes F3, F4, F5 and F6

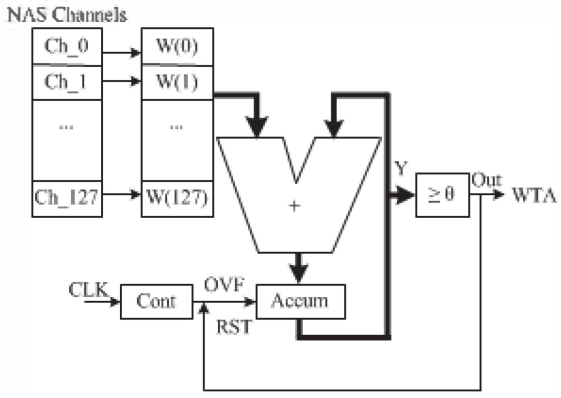

Fig. 6. Architecture of a single neuron from the convolution layer

The threshold values (θ) have been calculated with the result of (1) with sound samples of 10ms duration and using values of W obtained previously. The system has been implemented by a two-layer neural network, composed of a convolution layer and a Winner-Take-All (WTA) step in the second layer. Fig.6 shows the architecture of a single neuron of the convolutional spiking layer.

IV. EXPERIMENTAL SETUP

We have evaluated the capabilities of our convolutional architecture using two types of sound: pure tone sounds in the presence of white noise and electronic piano notes. In order to achieve this goal, the NAS was stimulated with these two kinds of musical sounds and its outputs were used as stimuli for the convolutional layer previously explained; thus, this layer worked with an AER bus that represented the corresponding musical tone. We decided to use these frequencies [130.813, 174.614, 261.626, 349.228, 698.456 and 880] Hz for pure tones because they are the fundamental frequencies of C3, F3, C4, F4, F5 and A5 notes. In the electronic piano notes, we chose F notes from the 3rd octave to the 6th, which are notes with the fundamental frequency shown in Table II.

TABLE II. FUNDAMENTAL FREQUENCY OF MUSICAL NOTES

Note	F3	F4	F5	F6
Freq.(Hz)	174.61	349.23	698.46	1396.91

Fig. 7 shows the real test scenario that contains a Xilinx ML507 development board [35] and the USB AERmini2 board [39], used as the AER event monitor. The FPGA of the ML507 development board, which is a Virtex5 FPGA (XC5VFX70T) [35], contains the NAS. Connected to a ML507 GPIO (General-Purpose Input-Output) port there is a small adapter that transforms the GPIO signals to the AER codification, following the CAVIAR standard [39]. The AER output bus is connected to the USB AERmini2 board [40], which sends AER events to the convolutional layer for sound processing, using the USB port. In the final implementation of the system, this convolutional layer will be placed in a FPGA.

Fig. 8 shows the experimental scenario, in which two different experiments were used to test the convolutional layer:

- In the first experiment, pure tones were used to train the convolutional layer using fundamental frequencies of C3, F3, C4, F4, F5 and A5 notes; after that, the system was stimulated again with the same tones but, in this case, white noise was added to check the noise tolerance of the system. These sounds were generated in MATLAB.
- In the second experiment, the system was trained with electronic piano notes: F3, F4, F5 and F6 which have a fundamental frequency shown in Table II. Finally, the system was excited with a fifty real-time piano note playback for each note.

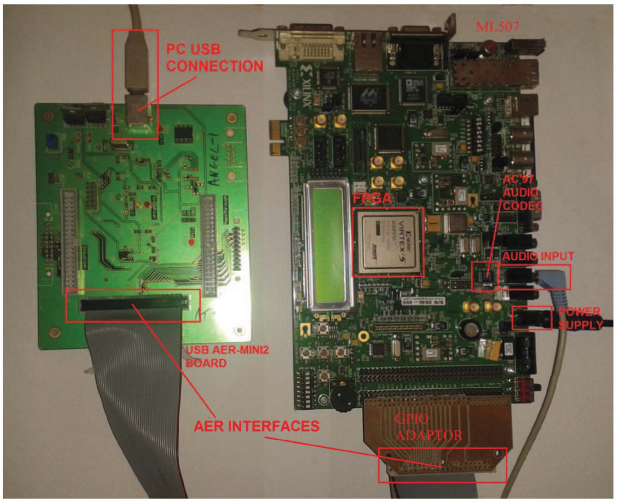

Fig. 7. Picture of the experimental test setup

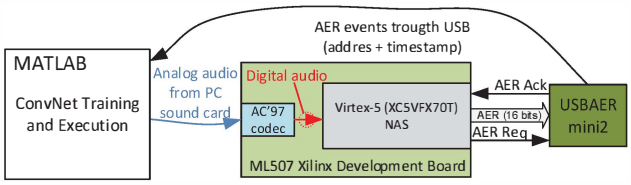

Fig. 8. Experiments test scenario block diagram

A. *Offline training*

The computer sent the audio information to the FPGA through audio output; thus, the USB-AERmini2 received the FPGA outputs and sent them to the computer for the convolutional layer training. The weights of the convolutional layer were obtained using the frequency response of each frequency band.

B. *Sound recognition performance*

In the first experiment, pure tones were emitted with -27.73dBW; after that, white noise was added, varying the noise level between -92.1 and 25.67 dBW. In the second experiment, a test set consisted on fifty played samples of each electronic piano note. Then, the NAS received the sounds, processed them in the spikes domain and sent the AER output events to the USB AERmini2. Finally, the USB AERmini2 sent them to the convolutional layer processing. The convolutional layer outputs were compared with the target of the test set to obtain the hit rate values (percentage of successes) of the recognition system.

V. EXPERIMENTAL RESULTS

The proposed pure tone classification system was quantitatively evaluated in the first experiment. The system was stimulated with fifty samples for each pure tone with white noise power sweep from -92.1 to 25.67 dBW. The system achieved 97.2% recognition accuracy for tones without white noise. Fig. 9 shows this value for each pure tone in the presence of different white noise powers, being the X-axis the frequencies between 130.813 and 880 Hz, the Y-axis the white noise varying the sound level between -92.1 and 25.67 dBW, and the Z-axis the percentage of successes. Fig. 10 is a detail from Fig. 9 to clarify the results of the F5 tone, where the X-axis is the white noise power, which varies between -92.1 and 25.67dBW. These figures show that the hit rate (percentage of successes) decreases with the increase of white noise (inversely proportional), and that there are frequencies more robust to white noise like C3, C4, F4 and A5 tones. Most of the pure tones have a hit rate over 90% with a noise power below -20dBW. However, the F3 pure tone showed a lower hit rate, which is below 10%, with a noise power of -48 dBW.

In the second experiment, the musical notes classification system was quantitatively evaluated by the playback of fifty samples of each electronic piano note. The experiment results are shown in blue columns in Fig. 11, where the X-axis indicates the F3, F4, F5 and F6 notes, and the Y-axis is the hit rate (experimental success rate). In this experiment, a low hit rate was obtained; generally, the hit rate for F4 note is below 20%. In order to improve the rate, a trained winner-take-all neuron layer was added at the output of the convolution layer. The WTA layer was trained with the outputs of the convolutional layer and the target vector. With this change, the system was significantly improved, obtaining a hit rate of 97.5% (a 60% improvement), as shown in yellow columns in Fig. 11.

VI. CONCLUSIONS

This work presents a musical sound recognition system based on a convolutional spiking network. Audio information acquisition was carried out by a novel neuromorphic auditory system that was also used to train the classification system. The output of this auditory system had the information obtained from a bank of spike-based band pass filters, which provided streams of spikes representing audio frequency components. These features of the input sound were sent to an AER convolutional network to identify the sound.

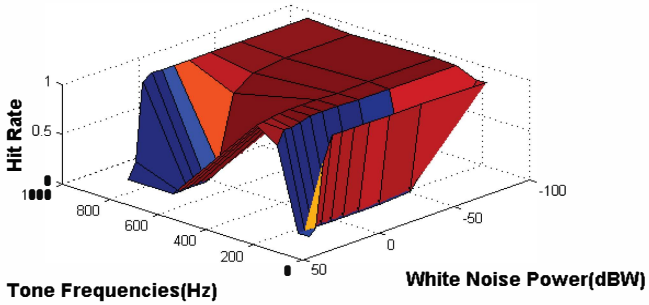

Fig. 9. Hit rate of a set of pure tones with white noise.

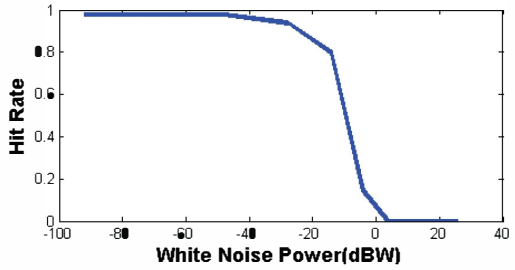

Fig. 10. Hit rate of 698.456Hz pure tone with white noise.

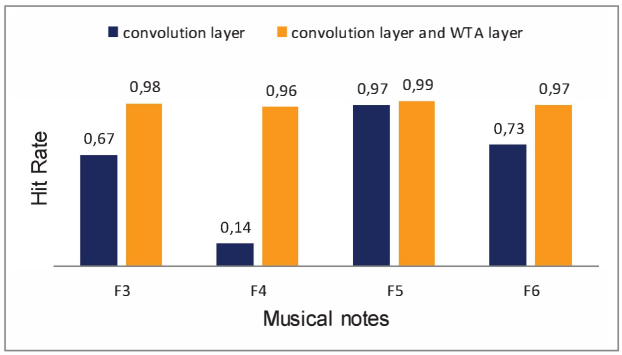

Fig. 11. Hit rate of the musical notes test with convolution layer (dark columns) and enhanced system (yelow columns).

Two different kinds of sound, pure tones and musical notes, were used to train and test the efficiency of the classification system. To date, there are no previous audio recognition systems that use convolutional spiking networks.

The pure tone recognition system accuracy was above 90%, even when the sound had white noise with the same power. On the other hand, the same convolutional layer architecture applied to musical note sounds obtained worse results but, using the WTA neuron layer, these results improved significantly. This musical notes recognition system hit rate is 97.5%, which is similar or better than the works appointed in the *Introduction*. The method proposed in [4] achieved 97.5% accuracy for 12 electric guitar notes with 30 neurons. The system proposed in this work has the same hit rate for 4 notes, but it only uses 8 neurons.

With this work we have tested the performance of a convolutional spiking layer in sound recognition applications. According to the results presented, this low cost system could be successful for research in this field.

REFERENCES

[1] N.J. Sieger and A. H. Tewfik, "Audio coding for representation in MIDI via pitch detection using harmonic dictionaries," Journal of VLSI Signal Processing, vol. 20, pp. 45–59, 1998.

[2] A. Barbancho, A. Klapuri, L. Tardon, and I. Barbancho, "Automatic transcription of guitar chords and fingering from audio," Audio, Speech, and Language Processing, IEEE Transactions on, vol. 20, no. 3, pp. 915-921, 2012.

[3] H. Penttinen, J. Siikonen, and V. Valimaki, "Acoustic guitar plucking point estimation in real time," Lab. Acoust. Audio Signal Process., vol. 2, no. 1, pp. 209–212, 2005.

[4] J. D. J. Guerrero-turrubiates, S. E. Gonzalez-reyna, S. E. Ledesma-orozco, and J. G. Avina-cervantes, "Pitch Estimation For Musical Note Recognition Using Artificial Neural Networks," pp. 53–58, 2014.

[5] F. Pishdadian and J. K. Nelson, "On the transcription of monophonic melodies in an instance-based pitch classification scenario," 2013 IEEE Digit. Signal Process. Signal Process. Educ. Meet. DSP/SPE 2013 - Proc., pp. 222–227, 2013.

[6] M. J. Newton and L. S. Smith, "Biologically-inspired neural coding of sound onset for a musical sound classification task," in Proceedings of the International Joint Conference on Neural Networks, 2011, pp. 1386–1393.

[7] A. Azarloo and F. Farokhi, "Automatic musical instrument recognition using K-NN and MLP neural networks," Fourth International Conference on Computational Intelligence, Communication Systems and Networks (CICSyN), 2012, pp. 289-294.

[8] D. Jaeckeld, R. Moeckel and S.-C. Liu, "Sound Recognition with spiking Silicon Cochlea and Hidden Markov Models". IEEE Prime 2010, Berlin, Germany, 18-21, July, 2010.

[9] B. Resch, M. Nilsson, A. Ekman, and W. Kleijn, "Estimation of the instantaneous pitch of speech," Audio, Speech, and Language Processing, IEEE Transactions on, vol. 15, no. 3, pp. 813–822, 2007.

[10] I. Uysal, H. Sathyendra and J.G. Harris, "A Biologically Plausible System Approach for Noise Robust Vowel Recognition," 49th IEEE International Midwest Symposium on Circuits and Systems (MWSCAS '06), Volume:1,2006, pp. 1529-1532.

[11] G. Indiveri, E. Chicca and R. Douglas, "A VLSI Array of Low-Power Spiking Neurons and Bistables Synapses with Spike-Timig Dependant Plasticity," IEEE T. Neural Networks 17(1), pp. 211-221, 2006.

[12] S. J. Thorpe, A. Brilhault, and J. A. Perez-Carrasco, "Suggestions for a biologically inspired spiking retina using order-based coding," *ISCAS 2010 - 2010 IEEE Int. Symp. Circuits Syst. Nano-Bio Circuit Fabr. Syst.*, pp. 265–268, 2010

[13] M. Mahowald, "VLSI analogs of neuronal visual processing: a synthesis of form and function," Ph.D. dissertation, California Institute of Technology, Pasadena,1992.

[14] K. A. Boahen, "Communicating Neuronal Ensembles between Neuromorphic Chips". Neuromorphic Systems. Kluwer Academic Publishers, Boston 1998.

[15] 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 Neuromorphic Engineering 7(178), 2013

[16] S.C. Liu, et al. Event-based 64-channel binaural silicon cochlea with Q enhancement mechanisms. In ISCAS 2010 - 2010 IEEE International Symposium on Circuits and Systems: Nano-Bio Circuit Fabrics and Systems. pp. 2027–2030.

[17] Yu, T. et al., 2009. Periodicity detection and localization using spike timing from the AER EAR. In Proceedings - IEEE International Symposium on Circuits and Systems. pp. 109–112.

[18] Kumar, N.; Himmelbauer, W.; Cauwenberghs, G.; Andreou, A.G. An analog VLSI chip with asynchronous interface for auditory feature extraction. IEEE Trans. Circuits and Systems II: Analog and Digital Signal Processing. 1998, 45, 600-606.

[19] R. Serrano-Gotarredona, T. Serrano-Gotarredona, A. Acosta-Jimenez, C. Serrano-Gotarredona,J.A. Perez-Carrasco, B. Linares-Barranco, A. Linares-Barranco, G. Jimenez-Moreno and A. Civit-Ballcels,"On Real-Time AER 2-D Convolutions Hardware for Neuromorphic Spike-Based Cortical Processing," IEEE T. Neural Network 19(7), pp. 1196-1219,2008

[20] A. Rios, C. Conde, I. Martin de Diego and E. Cabello, "Driver's Hand Detection and Trackingbased on Address Event Representation,"Computacional Modeling of Objetcs Represented in Images. ISBN: 978-0-415-62134-2, pp. 131-144

[21] A. Linares-Barranco, R. Paz-Vicente, F. Gómez-Rodriguez, A. Jiménez, M. Rivas, G. Jiménez and A. Civit, "On the AER Convolution Processors for FPGA," Proceedings of 2010 IEEE International Symposium on Circuits and Systems (ISCAS), pp. 4237-4240.

[22] L. Camunas-Mesa, A. Acosta-Jimenez, C. Zamarrefio-Ramos, T. Serrano-Gotarredona and B. Linares-Barranco,"A 32x32 Pixel Convolution Processor Chip for Address Event Vision Sensors With 155 ns Event Latency and 20 Meps Throughput,"IEEE Transactions on Circuits and Systems I , vol.58, no.4, pp.777-790, April 2011

[23] U. Zölzer, S. V. Sankarababu, and S. Möller, "PLL-based pitch detection and tracking for audio signals," Proc. 2012 8th Int. Conf. Intell. Inf. Hiding Multimed. Signal Process. IIH-MSP 2012, no. 6, pp. 428–431, 2012

[24] S. R. Mahendra, H. a. Patil, and N. K. Shukla, "Pitch estimation of notes in Indian classical music," Proc. INDICON 2009 - An IEEE India Counc. Conf., pp. 0–3, 2009.

[25] R. G. Amado and J. V. Filho, "Pitch detection algorithms based on zero-cross rate and autocorrelation function for musical notes," ICALIP 2008 - 2008 Int. Conf. Audio, Lang. Image Process. Proc., pp. 449–454, 2008.

[26] A. B. Nielsen, L. K. Hansen, and U. Kjems, "Pitch Based Sound Classification," 2006 IEEE Int. Conf. Acoust. Speech Signal Process. Proc., vol. 3, 2006.

[27] Y. Zhu and M. S. Kankanhalli, "Precise pitch profile feature extraction from musical audio for key detection," IEEE Trans. Multimed., vol. 8, pp. 575–584, 2006.

[28] P. Lichtsteiner, C. Posch, and T. Delbrück, "A 128x128 120dB 15µs Latency Asynchronous Temporal Contrast Vision Sensor," IEEE J. Solid-State Circuits, vol. 43, pp. 566–576, 2008.

[29] M. Dominguez-Morales, A. Jimenez-Fernandez, E. Cerezuela-Escudero, R. Paz-Vicente, A. Linares-Barranco and G. Jimenez,"On the Designing of Spikes Band-Pass Filters for FPGA,"Artificial Neural Networks and Machine Learning. (ICANN 2011). LNCS2011, 6792, pp. 389-396.

[30] A. Jimenez-Fernandez, A. Linares-Barranco, R. Paz-Vicente, G. Jiménez, A. Civit, "Building blocks for spikes signals processing,"In Proceedings of the 2010 International Joint Conference onNeural Networks (IJCNN), 18–23 July 2010; pp. 1-8.

[31] F. Gomez-Rodriguez, R. Paz, L. Miro, A. Linares-Barranco, G. Jimenez and A. Civit,"Two hardware implementation of the exhaustive synthetic AER generation method". LNCS 2005, 41, pp.534–540.

[32] E.Cerezuela-Escudero, M.J. Dominguez-Morales, A. Jiménez-Fernández, R. Paz-Vicente, A. Linares-Barranco and G. Jiménez-Moreno, "SpikesMonitors for FPGAs, an Experimental Comparative Study," In Proc. of the 12th International Work-Conference on Artificial Neural Networks, IWANN, Tenerife, Spain, 12–14 June 2013; Volume 7902, pp. 179-188.

[33] A. Jimenez-Fernandez, G. Jimenez-Moreno, A. Linares-Barranco, M.J. Dominguez-Morales, R. Paz-Vicente and A. Civit-Balcells, "A neuro-inspired spike-based PID motor controller for multi-motor robots with low cost FPGAs,"Sensors. 2012, 12, pp. 3831-3856.

[34] F. Perez-Peña, A. Morgado-Estevez 1, A. Linares-Barranco 2, A. Jimenez-Fernandez 2, F. Gomez-Rodriguez 2, G. Jimenez-Moreno 2 and J. Lopez-Coronado, "Neuro-Inspired Spike-Based Motion: From Dynamic Vision Sensor to Robot Motor Open-Loop Control through Spike-VITE". *Sensors*2013, *13*, pp. 15805-15832.

[35] Xilinx. Virtex-5 FXT FPGA ML507 evaluation platform http://www.xilinx.com/products/boards-andkits/HW-V5-ML507-UNI-G.htm

[36] Intel, 2002. Audio Codec '97. Available at: http://www-inst.eecs.berkeley.edu/~cs150/Documents/ac97_r23.pdf

[37] A. Rios-Navarro, A. Jimenez-Fernandez, E. Cerezuela-Escudero, M. Rivas, G. Jimenez, andA. Linares-Barranco, "Live Demostration: Real-Time Motor Rotation Frequency Detection by Spike-Based Visual and Auditory Sensory Fusion on AER and FPGA,"Artificial Neural Networks and Machine Learning (ICANN 2014) Lecture Notes in Computer Science. 2014, 8681, 847-848.

[38] A. Jimenez-Fernadez, E. Cerezuela-Escudero, L. Miro-Amarante, M.J. Dominguez-Morales, F. Gómez-Rodríguez,A. Linares-Barranco, G. Jimenez-Moreno, "On AER Binaural Cochlea for FPGA: Abstract for work-group Insect-inspired neuromorphic behaving systems: 'pro et contra'," The 2014 CapoCaccia Cognitive Neuromorphic Engineering Workshop.https://capocaccia.ethz.ch/capo/wiki/2014/inbs14

[39] R. Berner, et al., "A 5 Meps $100 USB2.0 Address-Event Monitor-Sequencer Interface". IEEE International Symposium on Circuits and Systems. ISCAS 2007, pp. 2451-2454, 2007.

[40] R. Serrano-Gotarredona, et al., "CAVIAR: A 45k Neuron, 5M Synapse, 12G Connects/s AER Hardware Sensory-Processing-Learning-Actuating System for High-Speed Visual Object Recognition and Tracking".IEEE Transactions on Neural Networks2009, 20, pp. 1417-1438