Design of a Spatiotemporal Correlation Filter for Event-based Sensors

Hongjie Liu, Christian Brandli, Chenghan Li, Shih-Chii Liu, Tobi Delbruck
Institute of Neuroinformatics, University of Zurich and ETH Zurich, Switzerland

***Abstract*— This paper reports the design of a 1mW, 10ns-latency mixed signal system in 0.18µm CMOS which enables filtering out uncorrelated background activity in event-based neuromorphic sensors. Background activity (BA) in the output of dynamic vision sensors is caused by thermal noise and junction leakage current acting on switches connected to floating nodes in the pixels. The reported chip generates a pass flag for spatiotemporally correlated events for post-processing to reduce communication/computation load and improve information rate. A chip with 128×128 array with 20×20µm^2 cells has been designed. Each filter cell combines programmable spatial subsampling with a temporal window based on current integration. Power-gating is used to minimize the power consumption by only activating the threshold detection and communication circuits in the cell receiving an input event. This correlation filter chip targets embedded neuromorphic visual and auditory systems, where low average power consumption and low latency are critical.**

Keywords—neuromorphic; event-based; AER; DVS; background activity; uncorrelated noise; filter; spatiotemporal correlation

I. INTRODUCTION

An event-based vision sensor is a sensor that produces asynchronous address events (AE) from active pixels. The dynamic vision sensor (DVS) is such an event-based sensor that generates events as soon as the change in log intensity since the last event exceeds an upper (ON) or lower (OFF) threshold [1]. If event-based sensors are interfaced to synchronous systems, the timing information of the AE events is coded in a timestamp. Since dynamic vision sensors have sparse, low-latency output, they are used in applications which require high-speed object tracking with low-latency feedback such as high-speed robotics [2].

However background activity (BA) in event-based sensor output is generated even when there is no real activity. BA is caused by thermal noise and junction leakage currents acting on switches connected to floating nodes. It affects the quality of the data and consumes unnecessary communication bandwidth and processing. In the DVS128 sensor [1], for example, the BA rate is about 0.05Hz per pixel at room temperature, while in the latest DAVIS (Dynamic and Active-Pixel Vision Sensor) prototypes [3], the BA rate is about 0.1 Hz per pixel at room temperature resulting a 16keps (kilo events per second) event rate which is significant, especially for an embedded system. Since the junction leakage current doubles with every 6- 8ºC increase in temperature, at 60ºC, the rate would increase by a factor of 30 to a background activity rate of almost 480keps, which is a significant amount of noise to process. Moreover, the leakage current and hence BA rate increases as CMOS technology scales down. The BA is particularly undesirable when the real event rate is low, for example when a DVS is used for tracking small objects that move slowly. In this case, the BA might lead to incorrect tracking.

The difference between the BA and the real activity events is that the BA of a pixel lacks temporal correlation with events to its spatial neighborhood unlike the real events, which arise from moving features or changes in illumination. Based on this difference, the BA can be filtered out by detecting events without the spatiotemporal correlation. An effective implementation of this noise reduction filter (*BackgroundActivityFilter)* is included in jAER, the software used with many event-based sensors [4]. This filter is routinely used to filter out background activity at a cost of about 30ns computing time per event on a fast PC. A hardware solution embedded with the sensor could save bandwidth and computation power and enables fully embedded applications without relying on a PC.

Fig. 1 shows an event-based sensor data processing system which includes the hardware BA filter solution. The sensor chip transmits events by activating the event request signal (Req) and presenting the event address bits (Addr<0:N>) to both the Logic and Filter chip. The Filter chip generates a Pass logic signal after processing the event and sends it to the Logic chip. The Logic chip then activates the acknowledge signal (Ack) to both the Sensor and the Filter chip. The correlated events (those that are tagged with Pass = logic 'High') are sent to the PC via the USB microcontroller unit (MCU) after they are tagged with a timestamp by the Logic chip. This architecture allows filtering out BA without reducing the system's maximum throughput.

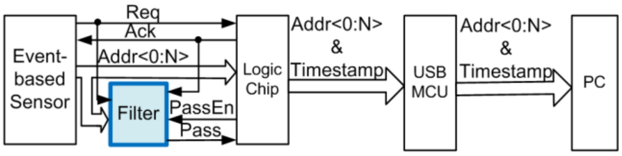

Fig. 1. The event-based data processing system.

II. THE SPATIOTEMPORAL CORRELATION PRINCIPLE

Fig. 2 shows the spatiotemporal correlation principle implemented in the BA filter chip. It first subsamples the address of its input event stream by ignoring a programmable number of X (row) or Y (column) address LSBs. This operation effectively projects events of certain blocks of sensor pixels, e.g. 1x1, 1x2, 4x4, etc. onto the same filter cell. This subsampling defines the spatial neighborhood. Each event that arrives at a filter cell opens a time window in which all following events of the same pixel block are allowed to pass the filter. The size of this time

This work is supported by the European Union funded project SeeBetter (FP7-ICT-2009-6), VISUALIZE (FP7-ICT-2011.9.11), the Swiss National Science Foundation through the NCCR Robotics, ETH Zurich, and the University of Zurich.

window dT, i.e. the temporal neighborhood, can also be programmed.

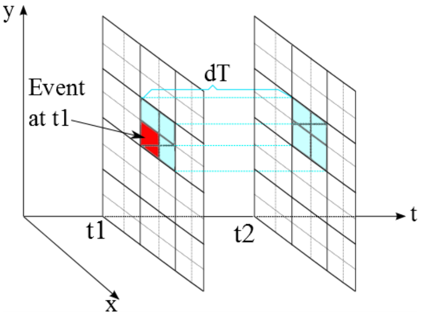

Fig. 2. Spatiotemporal correlation principle. An example showing that every 2x2 pixels on the sensor chip are projected to one cell on the BA filter chip (subsampling rate is 1 for both row and column). An event in the arrow pointed pixel at t1, provides spatiotemporal support for the 4 pixels including itself within time dT.

III. Circuit Design

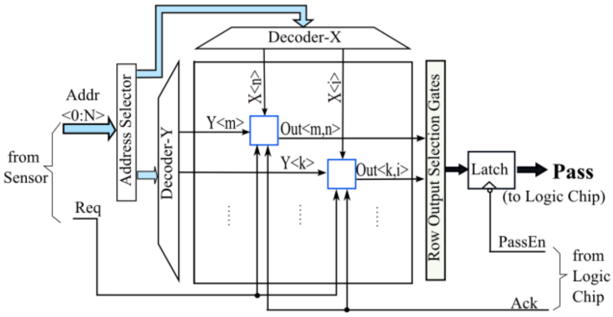

Fig. 3. Architecture of the BA filter chip.

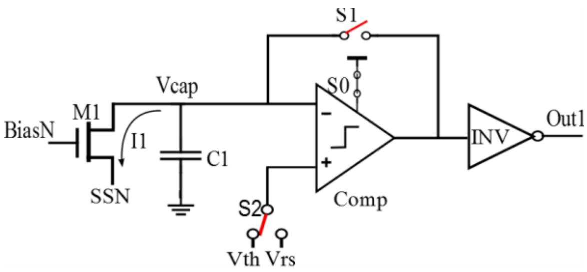

Fig. 4. Discharge - Compare - Reset circuitry within a cell to determine timing correlation.

Fig. 3 is a sketch of the overall architecture of the BA filter chip. The chip receives the word-serial AER output (Addr<0:N> and Req) [5] from the vision sensor, acknowledge signal Ack and PassEn signal (to Latch its output) from the Logic chip. It outputs a Pass signal to the Logic chip. The filter chip contains an array of 128x128 cells (only two cells are displayed).

The Address Selector can accept up to 12 bit input AER address and routes them to the input addresses of Decoder-X and Decoder-Y. This routing step can be configured to determine the address subsampling rate that determines the spatial neighborhood. The outputs of Decoder-X and Decoder-Y activate the cell which has the incoming event in its receptive field (the field in the sensor's pixel array which projects to one particular filter cell). Only the selected cell being powered on, greatly reduces power consumption of the chip. To reduce latency, transmission gates (Row Output Selection Gates) are added to allow only the selected row to write to the output of the whole array.

The filter cell design is shown in Fig. 4. Each filter cell has a comparator Comp, a capacitor C1 to store the event timing information and a transistor M1 to discharge the capacitor. The digital logic which controls the switches is not shown in Fig. 4. Upon the arrival of an event in its receptive field i.e. when the cell is selected by the Logical AND of the outputs of the decoders and Req, S2 is switched to Vth and S1 stays open. The comparator checks if Vcap is above Vth and if this is the case, the event is in the time window of a preceding event. The Out1 signal is therefore raised to generate a Pass signal. To avoid glitches on the Pass signal, it is latched using the PassEn signal coming from the Logic chip shown in Fig. 3. As soon as the Logic Chip acknowledges the transmission of the event and the Pass signal by raising the Ack line, S1 is closed and S2 is switched to Vrs to reset the capacitor C1 to Vrs. After the Req goes low, S1 is open again. Current through M1 then continuously discharges the capacitor C1 until the next time this cell receives an event. The time window in which Vcap is above Vth is thereby given by the following equation:

$$dT = C *(Vrs-Vth) \,/\, I_1$$

The current through M1, I_1 is controlled by a configurable bias generator [6]. It can range from 100fA to 20uA. Small current values are achievable by tying the source of M1 to a shifted source voltage SSN instead of ground.

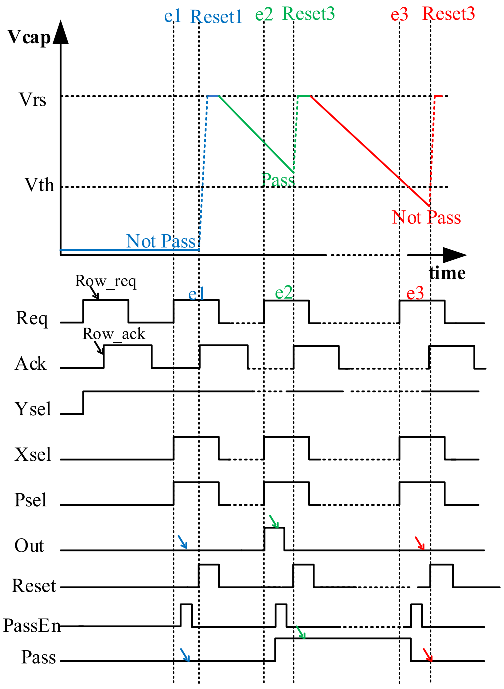

Fig. 5. Timing diagram to process three consecutive events in one filter cell. All the signals are active high. e1, as the first event in the cell is filtered away since it is not in the temporal neighborhood of a preceding event; e2 is passed because ISI (Interspike Interval, the time interal between two consecutive events) is smaller than dT; e3 is filtered away because ISI is larger than dT.

Fig. 5 shows a timing diagram of the operation of the chip over three incoming events in the same cell. In this particular case, to make illustration easier, we assume they share the same row request Row_req and row acknowledge Row_ack, which means that the events are from the same row in the sensor's pixel array. Xsel and Ysel are outputs of Decoder-X and Decoder-Y respectively and Psel is the logical AND of Xsel and Ysel which selects a cell. PassEn is turned high after a certain delay to latch the new Pass signal. Reset, which is the logical AND of Ack and Psel is then turned high to reset this particular pixel.

IV. Software Model

The correlation filtering principle of the designed circuit was modeled in a Java-based filter in jAER using the data recorded from a DAVIS sensor [3] observing a slot car racing around a track [7]. The car causes isolated and correlated activity just at the projected location of the car on the vision sensor and is well-suited to illustrate the filter action.

Like the filter chip, the Java-based filter has the following parameters: integrating current I_1, capacitor value Cap, threshold voltage Vth, reset voltage Vrs, and the subsampling rate which determines the spatial support. To model the mismatch in CMOS implementation, $I_{1,}$ Cap, Vth, Vrs are set as Gaussian distributed variables each with their own programmable standard deviation σ.

The steps to process each event are as follows:

1) Calculate the ISI (Interspike Interval, the time interal between two consecutive events) from the event timestamp map lastTimesMap;
2) Calculate Vcap based on ISI and I_1;
3) Compare Vcap to Vth and determine whether to filter the event away or not;
4) Reset Vcap to Vrs and update lastTimesMap.

Fig. 6 shows the 8s accumulated recording of the slot car for one cycle with and without the spatiotemporal correlation filter. Setting the subsampling rate to 2, I_1 to 100pA which provides dT=1ms and σ to 5%, value from the Monte-Carlo simulation results, the filter removes 99.8% of the 4.7keps background activity with less than 10% loss of the real activity.

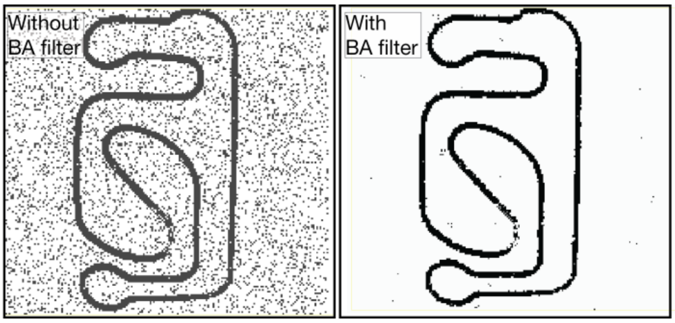

Fig. 6. 8s accumulated recording of the slot car for one run around the track without and with the spatiotemporal correlation filter. The fitler removes the BA leaving mostly only the slot car events.

Mismatch in the circuit does not add additional noise to the events but affects the filter performance. Fig. 7 plots the influence of device mismatch on the effectiveness of the filter. Monte-Carlo simulation results in distributions with σ < 5% for $I_{1,}$ Cap, Vth and Vrs. The plot shows that even with σ = 50% for all the parameters the filter can remove 98% of the BA. This increases our confidence in the performance of the fabricated chip to be tested later.

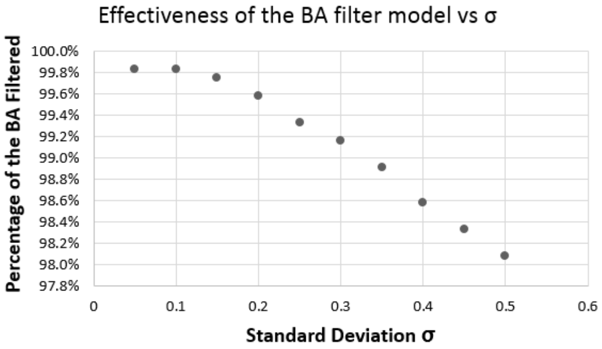

Fig. 7. The influnce of mismatch on the effectiveness of the BA filter.

V. Hardware Implementation

The BA filter is implemented in Tower's 1P6M 0.18um technology. The capacitor value is 165fF, implemented as MIM cap to save area. Given the current range of 100fA to 20uA, the time window of the cell can range from 5ns to 1s. Configuration circuit in the address selector was designed so that if the number of cells on a single chip is not sufficient, the chip can also be used in multi-chip setups. In such setups, each chip stores several programmable address bits which is then compared against the MSBs of an incoming address to determine if the address is within the address space of given chip. Latency of the peripheral circuitry (decoders, Address Selector, Output Row Selection Gates) is 7ns. The power consumption of the peripheral circuitry is dependent on the event rate. At 50Meps (million events per second), it is 800uW.

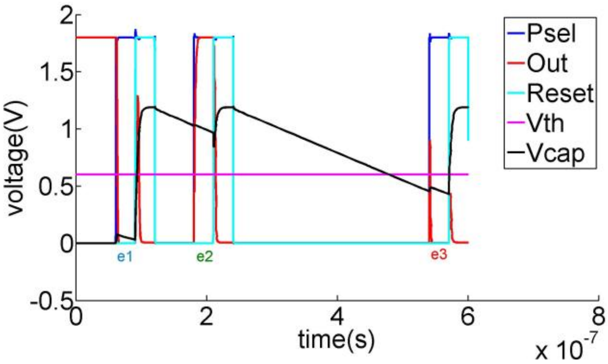

Fig. 8. The response of the cell for 3 consecutive input events. e2 is tagged with pass, e1 and e3 not pass. The result corresponds to the timing diagram shown in Fig. 5 except some glitches at the time when the cell is turned on.

Fig. 8 shows the transistor-level simulation of the filter cell over three consecutive input events. The current through the NMOS is set much larger than real so that the decrease of the capacitor voltage caused by discharging can be seen within a limited simulation time. In reality, the time difference is usually set to 1~10ms, hence the current should be around 4 orders smaller. The load in the output includes the post layout extracted capacitance. The response time of the cell is 3ns which adds up to a 10ns total latency of the filter. Simulated power consumption of a single cell is 180uW. Since only one cell in the BA filter chip is active when receiving an event, the total power consumption of the filter chip is less than 1mA at an event rate of 50Meps.

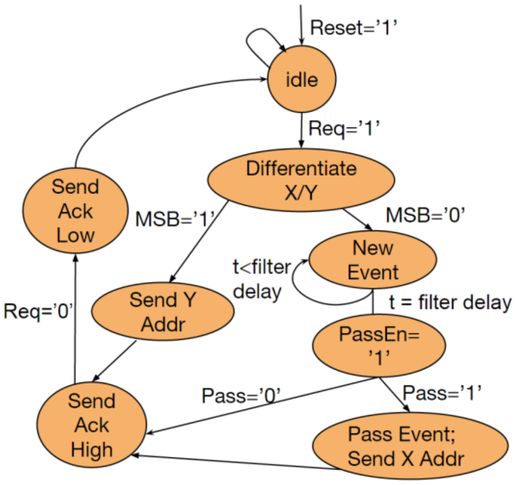

Fig. 9. Finite state machine implemented in the Logic chip to integrate the Filter chip in the system .

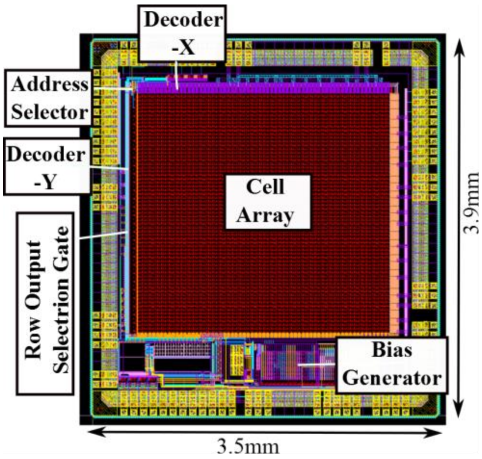

Fig. 10. Chip layout.

To integrate the chip into the data processing system, the previous state machine in the Logic Chip which handles handshaking with the sensor chip is modified and shown in Fig. 9. The Logic leaves the idle state after receiving an active Req from the vision sensor and first determines whether the request is from a row or a column. After receiving a column request, the Logic chip waits until the Pass signal is ready and then raises the PassEn signal. The Logic chip determines whether to filter out or store the event based on whether Pass is high or low. Afterwards it activates the Ack signal to both the sensor and the filter.

Fig. 10 shows the layout of the chip with a size of 3.5x3.9mm^2 using Tower's 0.18um technology.

Table. 1 summarizes the specifications of the BA filter chip.

Table. 1 Design Specifications

Fabrication process	0.18um process 1P6M with MIM cap
Supply voltage	3.3V analog, 1.8V digital
Cell size	20um by 20um
Expected Power Consumption at Event Rate 50Meps	1mW
Cell array	128x128
Die size	3.5mm x 3.9mm
Expected noise reduction percentage	98%
Spatiotemporal correlation	Programmable
Temporal correlation dynamic range	5ns to 1s
Expected latency	10ns

VI. Conclusion

A correlation filter chip for removing background uncorrelated noise events from event-based sensors was designed with Tower's 1P6M 0.18μm CMOS technology. The fabricated chip is being prepared for testing. This filter has a programmable spatiotemporal correlation window to meet the requirement of different array sizes and different application scenarios. The filter only adds 10ns latency to the whole event processing flow. A behavioral model of the filter chip shows that it can filter out 98% of the background noise with 10% loss of real activity events, taking device mismatch into account. The filter chip could also be used to filter uncorrelated noise for other event-based sensors, e.g. the AER cochlea [8].

[1] P. Lichtsteiner, C. Posch, and T. Delbrück, "A 128 x 128 120dB 15us Latency Asynchronous Temporal Contrast Vision Sensor," *IEEE J Solid-State Circuits*, vol. 43, no. 2, pp. 566–576, 2008.

[2] J. Conradt, R. Berner, M. Cook, and T. Delbruck, "An embedded AER dynamic vision sensor for low-latency pole balancing," in *Proc. 12th IEEE Int. Conf. on Computer Vision Workshops (ICCV)*, 2009, pp. 780–785.

[3] C. Brandli, R. Berner, M. Yang, S.-C. Liu, and T. Delbruck, "A 240x180 130 dB 3us Latency Global Shutter Spatiotemporal Vision Sensor," *IEEE J. Solid-State Circuits*, vol. Early Access Online, 2014.

[4] "jAER Open Source Project," *jAER Open Source Project*. [Online]. Available: http://jaerproject.org. [Accessed: 17-Sep-2013].

[5] K. A. Boahen, "A Burst-Mode Word-Serial Address-Event Link-I: Transmitter Design," *IEEE Trans. Circuits Syst. Regul. Pap.*, vol. 51, no. 7, pp. 1269–1280, Jul. 2004.

[6] M. Yang, S.-C. Liu, C. Li, and T. Delbruck, "Addressable Current Reference Array with 170dB Dynamic Range," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, 2012, pp. 3110–3113.

[7] *Slot car racer controlled by DVS Capo Caccia 2014*. 2014.

[8] S.-C. Liu, A. van Schaik, B. A. Minch, and T. Delbruck, "Asynchronous Binaural Spatial Audition Sensor with 2x64x4 Channel Output," *IEEE Trans Biomed. Circuits Syst.*, vol. 8, no. 4, pp. 453 – 464, 2013.