C14

ESTIMACIÓN DE DISTANCIAS MEDIANTE UN SISTEMA DE ESTÉREO-VISIÓN BASADO EN RETINAS DVS

Domínguez-Morales, Manuel; Jiménez-Fernández, Ángel; Cerezuela-Escudero, Elena; Luna-Perejón, Francisco; Durán-López, Lourdes; Linares-Barranco, Alejandro. Grupo de Robótica y Tecnología de Computadores. Dpto. Arquitectura y Tecnología de Computadores, E.T.S. Ingeniería Informática, Universidad de Sevilla.

RESUMEN

La estimación de distancias es uno de los objetivos más importantes en todo sistema de visión artificial. Para poder llevarse a cabo, es necesaria la presencia de más de un sensor de visión para poder enfocar los objetos desde más de un punto de vista y poder aplicar la geometría de la escena con tal fin. El uso de sensores DVS supone una diferencia notable, puesto que la información recibida hace referencia únicamente a los objetos que se encuentren en movimiento dentro de la escena. Este aspecto y la codificación de la información utilizada hace necesario el uso de un sistema de procesamiento especializado que, en busca de la autonomía y la paralelización, se integra en una FGPA. Esta demostración integra un escenario fijo, donde un objeto móvil realiza un movimiento continuo acercándose y alejándose del sistema de visión estéreo; tras el procesamiento de esta información, se aporta una estimación cualitativa de la posición del objeto.

Palabras clave: *Visión artificial, Ingeniería Neuromórfica, AER, DVS, FPGA.*

ABSTRACT

Image processing in digital computer systems usually considers the visual information as a sequence of frames. Digital video processing has to process each frame in order to obtain a result or detect a feature. In stereo vision, existing algorithms used for distance estimation use frames from two digital cameras and process them pixel by pixel to obtain similarities and differences from both frames; after that, it is calculated an estimation about the distance of the different objects of the scene. Spike-based processing implements the processing by manipulating spikes one by one at the time they are transmitted, like human brain. The mammal nervous system is able to solve much more complex problems, such as visual recognition by manipulating neuron's spikes. The spike-based philosophy for visual information processing based on the neuro-inspired Address-Event- Representation (AER) is achieving nowadays very high performances. In this work, it is proposed a two-DVS-retina connected to a Virtex5 FPGA framework, which allows us to obtain a distance approach of the moving objects in a close environment. It is also proposed a Multi Hold&Fire algorithm in VHDL that obtains the differences between the two retina output streams of spikes; and a VHDL distance estimator.

Keywords: *Computer visión, Neuromorphic Engineering, AER, DVS, FPGA.*

INTRODUCCIÓN Y OBJETIVOS

En los últimos años se han producido numerosos avances en el campo de la visión y el procesamiento de imágenes, ya que pueden aplicarse con fines científicos y comerciales a numerosos campos, como la medicina, la industria o el entretenimiento. Tratando de simular la visión de los seres humanos, los investigadores han experimentado con sistemas basados en dos cámaras, inspirados en la visión humana (8, 9). A partir de ahí se desarrolló una nueva línea de investigación centrada en la visión estereoscópica (1). En esta rama, los investigadores intentan obtener escenas tridimensionales utilizando dos cámaras digitales. Por lo tanto, tratamos de obtener cierta información que no se pudo obtener con una sola cámara como, por ejemplo, la distancia a la que un objeto se encuentra.

Mediante el uso de cámaras digitales, los investigadores han logrado un gran avance en este campo. Sin embargo, los sistemas digitales tienen algunos problemas que, incluso hoy en día, no se han resuelto (como el ajuste de visión estéreo en tiempo real, debido a la fase de extracción de características (10)).

Un objetivo importante en la visión estereoscópica es el cálculo de distancias entre el sistema de visión y el objeto en el que estamos enfocados, objetivo que aún está completamente abierto a la investigación. Los problemas relacionados con esto son el costo computacional necesario para obtener los resultados apropiados y los errores obtenidos después del cálculo de la distancia. Hay muchos algoritmos de alto nivel utilizados en la visión estéreo digital que resuelven el problema de cálculo de distancia, pero esto implica la intervención del pc en el proceso al ser computacionalmente muy costoso. Esto hace que sea difícil desarrollar un sistema autónomo en tiempo real.

Los sistemas o circuitos inspirados en la biología son enfoques novedosos para resolver problemas reales (2). Los sistemas pulsantes son una de las alternativas de la neuro-informática para imitar las capas neuronales del cerebro en las etapas de procesamiento de la información. Estos sistemas procesan la información de forma continua, sin discretización en frames. Las implementaciones de hardware de estos sistemas generalmente se componen de varios pasos: sensores (6), filtros (13), convoluciones (3), actuadores (14), etc.

Un gran problema en estas implementaciones es la comunicación, porque se necesita comunicar miles de neuronas de un chip al siguiente chip, pero existe una limitación en el número de pines. La representación Address-Event-Representation resuelve este problema (4). Por lo general, estos circuitos AER se construyen usando una lógica asíncrona auto-temporizada. El éxito de estos sistemas dependerá en gran medida de la disponibilidad de herramientas robustas y eficientes de desarrollo, depuración e interconexión de herramientas AER (15).

En este trabajo, se presenta un nuevo enfoque sobre la estimación de distancia, utilizando un sistema de visión estereoscópica AER. También se presentan algunos resultados de estimación de distancia, y se comparan con la distancia real.

Los elementos que componen este sistema son los siguientes (de izquierda a derecha, según la Figura 1): dos retinas DVS128 (6), dos placas USB-AER, una placa FPGA Virtex-5, una placa USBAERmini2 (5) y un pc que monitoriza la salida del sistema mediante el software jAER (7).

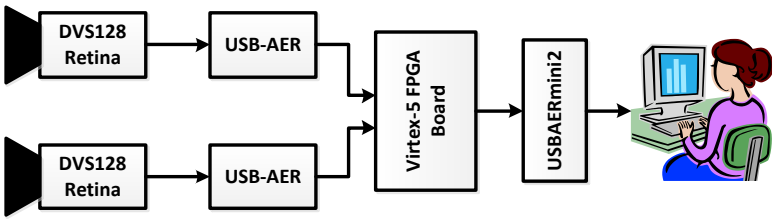

Figura 1. Componentes del Sistema e interconexiones entre ellos.

La placa USB-AER fue desarrollada en nuestro laboratorio durante el proyecto CAVIAR, y está basada en una FPGA Spartan II con dos megabytes de RAM externa y un microcontrolador cygnal 8051. Para comunicarse con el mundo exterior tiene dos puertos paralelos AER (conector IDE): uno de ellos se usa como entrada, y el otro es la salida. En el sistema, se han utilizado dos tarjetas USB-AER, una para cada retina. En estas placas se ha sintetizado en VHDL un filtro llamado Background-Activity-Filter, que nos permite eliminar el ruido del flujo pulsante producido por cada retina. Este ruido se debe a la naturaleza del píxel analógico de la retina. Así pues, a la salida de la USB-AER tenemos la información filtrada y lista para procesar.

La otra placa utilizada es una Xilinx Virtex-5, desarrollada por AVNET (12). Esta placa se basa en un FPGA Virtex-5 y tiene principalmente un gran puerto compuesto por más de ochenta GPIO (puertos de entradas/salidas de propósito general). Usando este puerto, hemos conectado una placa de expansión/prueba, que tiene pines estándar, y los hemos usado para conectar dos entradas AER y una salida. En dicha placa se implementa todo el procesamiento, que trabaja con los pulsos provenientes de cada retina, los procesa y obtiene las diferencias entre ambas retinas y la tasa de pulsos de esta diferencia. El diagrama de bloques del programa completo se muestra en la figura 3. El comportamiento del sistema y su funcionalidad se exponen en las siguientes secciones.

Figura 2. Sistema completo.

METODOLOGÍA

A. Algoritmo MULTI HOLD&FIRE

El tráfico procedente de ambas retinas se debe restar sobre la marcha para calcular las diferencias entre ellos. Para hacer eso, hemos utilizado la idea del bloque Hold&Fire (11) para obtener la diferencia de dos señales pulsantes. Este algoritmo resta dos pulsos, recibidos de dos puertos diferentes. Cuando recibe un evento, espera un corto periodo de tiempo para ver si recibe otro con la misma dirección. Si no recibe un segundo evento y el tiempo termina, dispara el pulso. De lo contrario, si recibe otro evento, dependiendo de su polaridad y su retina de procedencia, el algoritmo dispara un evento o no. Usando esta teoría en las retinas, si el segundo pulso recibido tiene la misma dirección, tenemos dos opciones: si el nuevo evento proviene de la otra retina, el evento se cancela y no se transmite ningún evento; pero si este segundo evento proviene de la misma retina, se envía el primer evento y este segundo evento asume el papel del primer evento y el sistema espera nuevamente el periodo de tiempo fijado. Esta operación Hold&Fire para restar o cancelar dos flujos de pulsos se describe en profundidad en (11).

En este trabajo, se ha extrapolado el bloque Hold&Fire al conjunto del sistema, teniendo un conjunto de 128x128 bloques (una para cada píxel de las retinas). Llamamos a esto el sistema Multi Hold&Fire, que nos permite calcular las diferencias entre dos flujos de eventos de salida de retinas.

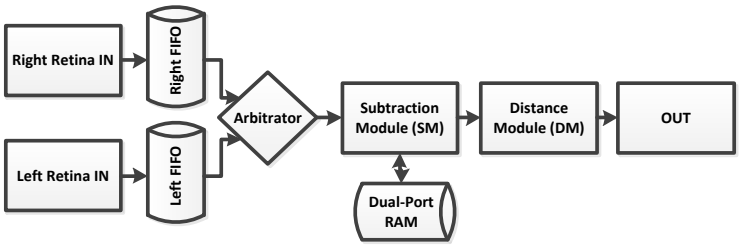

Figura 3. Bloques VHDL.

B. Sistema Completo

Nuestro algoritmo se basa en bloques múltiples Hold&Fire, como se comentó anteriormente (tiene un bloque H&F por cada dos píxeles equivalentes de las dos retinas). Trata cada píxel por separado y obtiene la diferencia entre este píxel en la retina izquierda y el mismo píxel en la retina derecha. Al final, tenemos un flujo de eventos que representa la diferencia de ambas retinas en la salida de nuestro sistema. El proyecto VHDL tiene estos bloques:

- Dos FIFOs: almacenan una gran cantidad de eventos de ambas retinas.

- Un arbitrador: selecciona los eventos de ambas FIFOs dependiendo de la ocupación de ellas.

- Módulo de resta: aplica el algoritmo Multi Hold&Fire a la secuencia de eventos recibidos.

- Módulo de distancia: estima la distancia. Lo profundizaremos en la siguiente sección.

C. Estimación de distancias

Como se explicó en la sección anterior, este sistema usa el módulo de distancia para obtener un acercamiento de la distancia donde se encuentra el objeto móvil. El algoritmo obtiene una estimación cualitativa de la distancia, al igual que la vista humana: no obtendremos un resultado cuantitativo. Este hecho no es un fracaso, porque la visión humana no puede calcular distancias, solo puede estimar distancias, en base a experiencias previas. Entonces, nuestro algoritmo está muy cerca del comportamiento de la visión humana.

En nuestro sistema, ambas retinas se posicionan con un cierto ángulo para obtener una distancia de enfoque de 1 metro. Para hacer eso, hemos puesto nuestras retinas en una base, separadas 13'5 cm entre ellas. Hemos obtenido el sistema que se muestra a continuación (Figura 4).

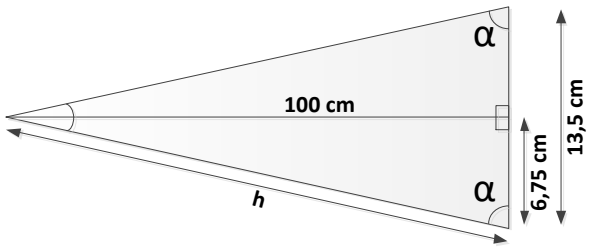

Figura 4. Posicionamiento de retinas.

Aplicando Pitágoras y reglas trigonométricas, podemos obtener:

$$h^2 = 6'75^2 + 100^2; h = 100'22755$$

$$\sin\alpha = \frac{100}{100'22755} = 0'99773$$

$$\arcsin 0'99773 = 86'1387^{o}$$

Lo cual equivale a 1'5 radianes aproximadamente.

Los algoritmos existentes en los sistemas digitales extraen características de ambas cámaras, las procesan y tratan de unir los objetos de ambas cámaras frame a frame (10). Este proceso tiene costos computacionales importantes y no funciona en tiempo real. Queremos hacer lo mismo en tiempo real usando AER. Como primer paso para lograr este objetivo, proponemos un algoritmo basado en la tasa de picos de la salida Multi Hold&Fire (17), pero no solo en él.

Si dos retinas se están enfocando en el mismo punto, el ancho de banda de AER de ambas salidas debería ser muy similar para un objeto en movimiento en el punto de enfoque. Por lo tanto, la diferencia de tráfico de ambas retinas debería ser la más baja posible. Si el objeto se mueve en la dirección de las retinas y está más cerca que el punto de enfoque, el tráfico en ambas retinas debería aumentar porque el objeto se hace más grande y hay más píxeles activos, pero la diferencia de tráfico no se puede cancelar porque el objeto no está centrado. Por lo tanto, la diferencia de tráfico debería aumentar considerablemente cuando el objeto está más cerca. Cuando el objeto está más lejos, ya que se estimulan menos píxeles de la retina y también son diferentes en ambas retinas, el tráfico también es más alto que en el punto de enfoque, pero debe ser menor que cuando el objeto se acerca.
La frecuencia de pulsos de las diferencias de ambas retinas (salida Multi Hold&Fire) no es la misma para cada objeto. Un objeto más grande disparará más pulsos que uno pequeño. Para tener una estimación de distancia normalizada, debemos combinar la tasa de pulsos de la resta con la tasa de pulsos antes de las modificaciones Multi Hold&Fire. Por lo tanto, deberíamos obtener más precisión, mejores resultados y una estimación más real.

RESULTADOS Y DISCUSIÓN

Después de los cálculos y calibraciones anteriores, las retinas se posicionan con un ángulo de 86'1387º entre ellas para obtener una distancia focal de un metro. Después de eso, se introducen trenes de pulsos al algoritmo Multi Hold&Fire usando un objeto en movimiento. Este estímulo se mueve a diversas distancias, cerca y lejos, de las retinas en varias ocasiones, por lo que se obtienen datos de diferentes

distancias; y se monitoriza la frecuencia de pulsos disparados durante este proceso. Se han utilizado estos datos con la tasa de pulsos anteriores del algoritmo MH&F, como se detalló anteriormente. La frecuencia de pulsos resultante se ha almacenado utilizando el software jAER (7). Después de las mediciones, se han registrado todos los resultados y se ha elaborado un gráfico con todos ellos. Este gráfico indica la cantidad de pulsos de media con respecto a la distancia del objeto (ver figura 5).

En el punto de coincidencia central de la distancia focal de cada retina, el MH&F funciona como un restador perfecto y disparará muy pocos pulsos, por lo que la tasa de pulsos en este punto es la más baja de todos los puntos de medición. Si nos acercamos a las retinas, la frecuencia de los pulsos aumentará porque el objeto se hace más grande y cada retina lo ve desde un punto de vista diferente, por lo que el restador no actuará de manera perfecta.

De lo contrario, si aumentamos la distancia del objeto respecto a las retinas (más allá del punto central de foco), la frecuencia de los pulsos aumentará un poco debido al resultado de la resta (diferentes puntos de vista de la retina), pero el objeto se vuelve más pequeño, por lo que compensa este error: a medida que el objeto se aleja, más pequeño es; y, por lo tanto, menor es la tasa de pulsos (se disparan menos, pero la resta actúa peor y dispara más pulsos). Es por eso que estos aspectos se equilibran entre sí, obteniéndose una tasa constante a mayor distancia.

Por lo tanto, se puede ver un gráfico donde la tasa de pulsos se incrementa mucho cerca de las retinas y aumenta ligeramente siempre que alejemos el objeto del punto de colisión focal. Se pueden apreciar los resultados experimentales en la figura 5. Hemos estimulado nuestro sistema utilizando dos objetos de diferentes tamaños, que producen diferentes tasas de eventos de salida en las retinas. Las mediciones se han tomado desde el punto de partida de 10 centímetros hasta los 150 centímetros. Fueron tomadas cada 10 centímetros.

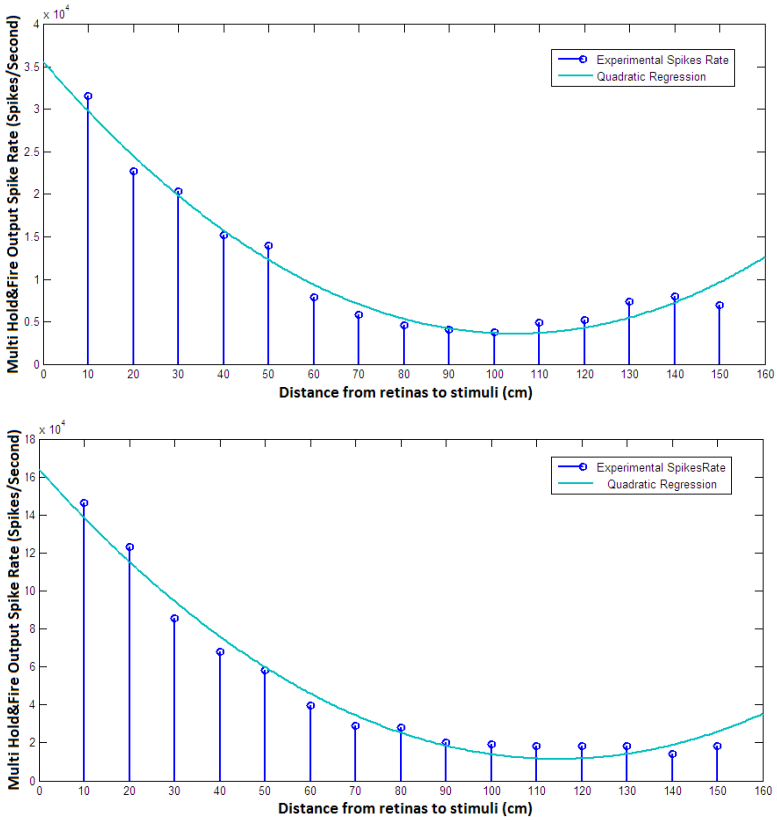

Figura 5. Tasa de eventos VS distancia con dos estímulos diferentes.

En la figura 5 se pueden ver los resultados experimentales obtenidos con dos estímulos diferentes: el primer gráfico corresponde a un péndulo y el segundo a una regla oscilatoria. Es interesante observar que, aproximadamente, a una distancia de 100 centímetros (colisión focal de ambas retinas) obtenemos la tasa de pulsos más baja. Si vemos que las mediciones se toman más cerca, se puede ver que la tasa de pulsos aumenta y, lejos del punto de colisión focal, la tasa de pulsos aumenta un poco. La distancia del objeto en movimiento se puede estimar con una regresión cuadrática (línea en azul que se muestra en la figura 5).

Nuestro sistema se comporta de forma bastante similar a la percepción humana: a priori, sin conocer el tamaño del objeto, no podemos proporcionar distancias exactas, solo una aproximación. Esta aproximación depende de nuestra experiencia, pero el sistema propuesto no aprende. Sin embargo, podemos medir la distancia de forma cualitativa e interactuar con un objeto en un entorno cercano.

Al relacionar la comparación entre este método y los métodos clásicos de visión por computador, tenemos que decir que este algoritmo es solo uno de los pasos implementados en un cálculo de distancia de visión por computadora. Para poder comparar la estimación de la distancia AER con la visión clásica de la computadora, necesitamos unirla a un proceso de adaptación previo y, en algunos casos, con un paso de calibración previa.

En relación con la visión clásica, el mecanismo proporcionado está relacionado de alguna manera con el método llamado "cálculo de distancia usando mapa de disparidad", que intenta determinar una estimación de primer acercamiento sobre la distancia de un objeto usando las disparidades entre ambas cámaras en un sistema de visión estéreo.

Estamos trabajando en mejoras a este sistema, utilizando varios filtros en cascada que agregan restricciones al sistema y mejoran la eficiencia: correspondencia en visión estéreo AER (10), correcciones espaciales (16) y seguimiento de objetos (10).

CONCLUSIONES

Se han expuesto las dificultades existentes para calcular distancias en sistemas de visión digitales. Es por ello que se ha introducido un enfoque biológico (AER) para trabajar, como un nuevo paradigma en Ingeniería Neuromórfica. Las ventajas de este método han sido expuestas y evaluadas.

Tras ello, se ha propuesto un método de estimación de distancias que contempla trabajar con objetos en movimiento utilizando Address-Event-Representation en un entorno cercano. Para lograr este objetivo, se ha utilizado un sistema de visión estereoscópica con dos retinas DVS, implementando el procesado en VHDL sobre hardware programable.

Se ha descrito y mostrado todo el sistema utilizado (y cada elemento de él) con el objeto de obtener la estimación de distancias. Con el sistema de hardware descrito, los algoritmos utilizados han sido explicados en detalle. El primer algoritmo usa un método para obtener diferencias de tasas de pulsos entre ambas retinas en tiempo real. Con estas diferencias se ha introducido el segundo algoritmo, que funciona con la tasa de pulsos obtenida en nuestro sistema después del cálculo de diferencias.

Con los resultados de estos dos algoritmos, hemos sido capaces de modelar la frecuencia de los pulsos con respecto a la distancia del objeto. Los resultados de la simulación son muy alentadores, ya que se puede observar en los gráficos que existe una relación entre la distancia y la velocidad de los picos después de nuestro procesamiento, y que este sistema funciona bastante similar a la percepción humana.

AGRADECIMIENTOS

Queremos agradecer la contribución de Tobias Delbruck y Raphael Berner, que han deserrollado herramientas que han sido utilizadas en este trabajo. Este trabajo ha sido financiado por el proyecto de investigación del MICINN del gobierno español COFNET (TEC2016-77785-P).

BIBLIOGRAFÍA

1. *Barnard, S.T. and Fischler M.A.: Computational Stereo. Journal ACM CSUR. Volume 14 Issue 4 (1982).*
2. *Shepherd, G. M.: The Synaptic Organization of the Brain. Oxford University Press, 3rd Edition (1990).*
3. *Linares-Barranco, A. et al: AER Convolution Processors for FPGA. ISCASS (2010).*
4. *Sivilotti, M.: Wiring Considerations in analog VLSI Systems with Application to Field-Programmable Networks. Ph.D. Thesis, Caltech (1991).*
5. *Berner, R.; Delbruck, T.; Civit-Balcells, A. and Linares-Barranco, A.: A 5 Meps $100 USB2.0 Address-Event Monitor-Sequencer Interface. ISCAS, New Orleans, 2451 – 2454 (2007).*

6. *Lichtsteiner, P.; Posh, C. and Delbruck, T.: A 128×128 120dB 15 us Asynchronous Temporal Contrast Vision Sensor. IEEE Journal on Solid-State Circuits, vol. 43, No 2, pp. 566-576, (2008).*
7. *jAER software: http://sourceforge.net/apps/trac/jaer/wiki*
8. *Benosman, R. and Devars, J.: Panoramic stereo vision sensor. International Conference on Pattern Recognition, ICPR (1998).*
9. *Benosman, R. et al: Real time omni-directional stereovision and planes detection. Mediterranean Electrotechnical Conference. MELECON (1996).*
10. *Dominguez-Morales, M. et al: Image Matching Algorithms using Address-Event-Representation. SIGMAP (2011).*
11. *Jimenez-Fernandez, A. et al: Building Blocks for Spike-based Signal Processing. IEEE International Joint Conference on Neural Networks, IJCNN (2010).*
12. *AVNET Virtex-5 FPGA board: http://www.em.avnet.com/drc*
13. *Serrano-Gotarredona, R. et al.:AER Building Blocks for Multi-Layer Multi-Chip Neuromorphic Vision Systems, NIPS (2005).*
14. *Linares-Barranco, A. et al.: AER Neuro-Inspired interface to Anthropomorphic Robotic Hand, IJCNN (2006).*
15. *Serrano-Gotarredona, R. et al.: CAVIAR: A 45k-neuron, 5M-synapse AER Hardware Sensory-Processing-Learning-Actuating System for High-Speed Visual Object Recognition and Tracking. IEEE Trans. On Neural Networks, Volume 20, Issue 9, Sept. 2009.*
16. *Jimenez-Fernandez, A. et al.: Neuro-inspired system for real-time vision sensor tilt correction, ISCAS (2010).*
17. *Dominguez-Morales, M. et al.: An Approach to Distance Estimation with Stereo Vision Using Address-Event-Representation. ICONIP (2011)*
18. *Gómez-Rodríguez, F. et al.: Real Time Objects Tracking Using a Bio-Inspired Processing Cascade Architecture. ISCAS 2010*