

Reconocimiento de micro partículas de polen con algoritmos de procesamiento de imágenes implementados en dispositivos reconfigurables

Juan Manuel Ruiz Varela

Centro de Investigación y de Estudios Avanzados del Instituto Politécnico Nacional,

jmruiz@gdl.cinvestav.mx

Susana Ortega Cisneros

Centro de Investigación y de Estudios Avanzados del Instituto Politécnico Nacional,

sortega@gdl.cinvestav.mx

Adrian Pedroza de la Cruz

Centro de Investigación y de Estudios Avanzados del Instituto Politécnico Nacional,

apedroza@gdl.cinvestav.mx

Juan José Raygoza Panduro

Centro de Ciencias Exactas e Ingeniería "CUCEI",
Universidad de Guadalajara

juan.raygoza@cucei.udg.mx

Jorge Rivera Domínguez

Centro de Investigación y de Estudios Avanzados del Instituto
Politécnico Nacional

jrivera@gdl.cinvestav.mx

Resumen: En este artículo se presenta una propuesta de un dispositivo capaz de identificar y enumerar las diversas familias de esporas de polen existentes a partir de una muestra obtenida. Este dispositivo cuenta con la implementación de dos algoritmos, el primero se enfoca al procesamiento de imágenes y el segundo a la extracción de firmas de las imágenes ya procesadas, de ésta forma es posible obtener la identificación y clasificación de cada una de las esporas de polen en la muestra, para así posteriormente hacer el conteo. Este proyecto se lleva a cabo utilizando una videocámara Costar SI-C400N adaptada a un microscopio, la tarjeta de desarrollo DE2i Altera, basada en un dispositivo Cyclone IV FPGA, memorias embebidas y un Softcore NIOS II. Esto será de gran utilidad en el área de la medicina, siendo más objetivos en el área de la alergología e inspección. Para la realización del proyecto se desarrollan las siguientes etapas, la primera está basada en la extracción de la imagen, esto se desarrolla con la ayuda de una videocámara montada en un microscopio con el cuál obtendremos el video de la muestra de esporas de polen a tratar. Se utiliza una videocámara la cual proporciona a su salida una señal analógica, dicha señal es introducida en la tarjeta para desarrollar el procesamiento adecuado. Al adquirir el video, éste es almacenado en la memoria SDRAM, la siguiente etapa se basa principalmente en leer el video almacenado y procesarlo correctamente con la ayuda del Softcore NIOS II, el cual tiene los algoritmos programados con lenguaje de alto nivel para hacer el procesamiento adecuado sobre el video entrante. En la tercera etapa, se guarda el video ya procesado en la memoria SRAM, para finalmente ser mostrado mediante el puerto VGA.

Palabras clave: FPGA, Softcore NIOS II, memorias embebidas, Procesador, procesamiento de imágenes, sistemas embebidos, microscope, SDRAM memory, SRAM memory, VGA port.

Recognition of pollen micro particles with image processing algorithms implemented in reconfigurable devices

Abstract: This paper presents a proposal of a device capable of identifying and listing the various families of existing pollen spores from a sample obtained is presented . This device has the implementation of two algorithms, the first focuses the image processing and the second to the extraction of signatures of images already processed , this way it is possible to obtain the identification and classification of each pollen spores in the sample , so then do the counting. This project is carried out using a camcorder Costar SI- C400N adapted to a microscope, DE2i Altera development board , based on a Cyclone IV FPGA device , embedded memories and Nios II Softcore . This will be useful in the area of medicine , being more goals in the area of allergy and inspection. For the project the following stages are developed , first is based on the extraction of the image, it develops with a camcorder mounties on a microscope what we get the video of sample pollen spores. Using a camcorder which provides an output analog signal, this signal is introduced into the card to develop appropriate processing. By acquiring the video, it is stored in the SDRAM memory , the next stage is mainly based on reading the stored video and process it correctly with the help of Softcore NIOS II , which has programming algorithms with high-level language for processing right on the incoming video. In the third stage, video already processed is stored in the SRAM memory , to finally be shown through the VGA port.

Keywords: FPGA, Softcore NIOS II, embedded memories, Processor, image processing, embedded systems, microscopio, memoria SDRAM, memoria SRAM, puerto VGA.

1. Introducción

El estudio de clasificación y conteo de esporas de polen que afectan todas las regiones del mundo se desarrolla de forma manual, lo que resulta en una tarea muy agobiante y demorada para el investigador, por lo que con éste proyecto, el operador podrá desarrollar ésta labor de forma automática, invirtiendo menor

cantidad de tiempo. Se considera que también será de impacto en el área de diagnóstico médico, debido a que con esta clasificación se podrá diagnosticar que esporas, en específico y en que concentración, son portadas por un paciente el cuál sufra de intoxicación por esporas de polen, haciendo más fácil la intervención médica, y se pueda suministrar el tratamiento adecuado de forma más rápida y exacta.

El procesamiento de imágenes ha sido muy utilizado en diversas aplicaciones; tales son el sector industrial, comercial, civil y militar durante más de un par de décadas. La detección de bordes es siempre el primer paso a seguir en muchos algoritmos de procesamiento de imágenes, ya que hace una reducción en la cantidad de datos y filtra toda información inútil. Con el avance en ésta disciplina se han propuesto gran cantidad de detectores de bordes, como el detector de Roberts, el detector de Prewitt, detectores Laplacianos, el detector de Sobel y el detector de Canny, entre otros. De los algoritmos anteriores, el de Canny es el mejor detector de bordes en casi todas las situaciones (Raman, M. et al., 2009).

Hoy en día, con el rápido desarrollo de la tecnología de la imagen, las resoluciones se han multiplicado en los últimos años, por esto se propone una plataforma de hardware novedosa en FPGA, debido a su capacidad de procesamiento paralelo y características reconfigurables, además, se ha convertido en una plataforma paralela útil para procesamiento de imágenes. En (Sowmya,S. et al.,2011), se propone una arquitectura de hardware basada en FPGA para la mejora de la imagen, en (Wenhao, H. et al.,2008), se propone el algoritmo de Canny, igualmente basado en FPGA.

Hoy en día, con el rápido desarrollo de la tecnología de la imagen, las resoluciones se han multiplicado en los últimos años, por esto se propone una plataforma de hardware novedosa en FPGA, debido a su capacidad de procesamiento paralelo y características reconfigurables, además, se ha convertido en una plataforma paralela útil para procesamiento de imágenes. En (Sowmya,S. et al.,2011), se propone una arquitectura de hardware basada en FPGA para la mejora de la imagen, en (Wenhao, H. et al.,2008), se propone el algoritmo de Canny, igualmente basado en FPGA.

2. Procesamiento digital de imágenes

Actualmente el procesamiento digital de imágenes es una herramienta muy específica en computación, se puede definir como las técnicas que se aplican a las imágenes digitales. Este procedimiento se utiliza en diversas áreas importantes (Osio, J. et al., 2011):

- Mejorar la calidad de la información de una imagen.
- Procesamiento de los datos de una imagen de forma autónoma.
- Suavizar la imagen: reducir la cantidad de variaciones de intensidad entre píxeles vecinos.
- Eliminar ruido: eliminar aquellos píxeles cuyo nivel de intensidad es muy diferente al de sus vecinos y cuyo origen puede estar tanto en el proceso de adquisición de la imagen como en el de transmisión.
- Realzar bordes: destacar los bordes que se localizan en una imagen.
- Detectar bordes: detectar los píxeles donde se produce un cambio brusco en la función intensidad.

Para el procesamiento es necesario tratar a la imagen como una matriz de valores numéricos, cada elemento de la matriz representa un pixel. Los valores de cada pixel se encuentran entre 0 y 255, donde el 0 corresponde al negro que es el tono más oscuro y el 255 corresponde al blanco que es el tono más claro, el proceso de filtrado se puede desarrollar tanto en el dominio de la frecuencia como en el dominio del espacio (Rodriguez, R. et al., 2012).

3. Detección de bordes

Los bordes de una imagen son las secciones en donde ocurre un cambio brusco de intensidad, es decir, en donde existe una transición entre dos partes que tienen diferentes niveles de gris, el resultado de este proceso sirve como entrada para tareas adicionales como reconocer objetos, compresión de imágenes, registro y alineación (Rajbahadur, S. et al., 2013).

Hasta el momento se cuenta con varios algoritmos de detección de bordes, entre los más destacados están:

- Algoritmo de Sobel.
- Algoritmo de Prewitt.
- Algoritmo Laplaciano.
- Algoritmo de Roberts.
- Algoritmo de Canny.

El último de los antes mencionados se eligió para la implementación, debido a la eficiencia en el tratamiento de la imagen que mostró en (Raman, M. et al., 2009) y también por ser el adecuado para este tipo de aplicación.

4. Algoritmo de Canny

El algoritmo de Canny es un operador desarrollado por John F. Canny en 1986 que utiliza un algoritmo de múltiples etapas para detectar una amplia gama de bordes en imágenes. Lo más importante es que Canny también desarrolló una teoría computacional acerca de la detección de bordes que explica por qué la técnica funciona (Gonzalez, R. et al., 1992).

A. *Desarrollo del algoritmo de Canny*

El propósito de Canny era descubrir el algoritmo óptimo de detección de bordes. Para que un detector de bordes pueda ser considerado óptimo debe cumplir los siguientes puntos:

Buena detección - el algoritmo debe marcar el mayor número real en los bordes de la imagen como sea posible.

Buena localización - los bordes de marca deben estar lo más cerca posible del borde de la imagen real.

Respuesta mínima - El borde de una imagen sólo debe ser marcado una vez, y siempre que sea posible, el ruido de la imagen no debe crear falsos bordes.

Para satisfacer estos requisitos se utiliza el cálculo de variaciones, una técnica que encuentra la función óptima para un funcional indicado. En el algoritmo de Canny,

esta función se describe por la suma de cuatro términos exponenciales, pero se puede aproximar por la primera derivada de una gaussiana.

Entre los métodos de detección de bordes desarrollado hasta el momento, el algoritmo de detección de bordes es uno de los métodos más estrictamente definido que proporciona la detección buena y confiable. Debido a su optimización para cumplir con los tres criterios para la detección de bordes y la simplicidad del proceso de implementación, se convierte en uno de los algoritmos más populares para la detección (Xiaoyang, L. et al., 2012).

B. Etapas del algoritmo de Canny

El proceso del algoritmo de detección de bordes de Canny puede ser descrito en 5 etapas diferentes:

- 1. Aplicar el filtro gaussiano para suavizar la imagen con el fin de eliminar el ruido.
- 2. Encontrar los gradientes de intensidad de la imagen.
- 3. Aplicar no supresión máxima para deshacerse de la respuesta espuria de detección de bordes.
- 4. Aplicar doble umbral para determinar los bordes potenciales.
- 5. Seguimiento de borde por histéresis: Finalizar la detección de bordes mediante la supresión de todos los otros bordes que son débiles y no están conectados a los bordes fuertes (Acharya, T. et al., 2005).

5. Extracción de las firmas y Clasificación

Para hacer el desarrollo de éste punto se hará uso de descriptores, los cuales representan las características elementales en un objeto, además de realizar una adecuada discriminación, ser suficientemente fiables y lo más independiente posible. Existen descriptores de contornos y de regiones para el reconocimiento e identificación de objetos en una imagen. Además de un conjunto de operadores morfológicos que permiten la manipulación de la forma de los objetos, extracción de contorno, y clasificación de objetos según su tamaño (Gonzalez, R. et al., 1992).

Los esquemas de representación de forma deben tener ciertas propiedades deseables:

Unicidad: Cada objeto debe tener una única representación.

Invariancia frente a transformaciones geométricas, como traslaciones, rotaciones, cambios de escala y reflexiones.

Sensibilidad a capacidad de diferenciar objetos casi iguales.

Abstracción del detalle o capacidad para representar los rasgos característicos básicos de los objetos y abstraer los detalles.

6. Diseño e implementación

Para la elaboración del proyecto, se presenta la metodología de forma gráfica (Fig. 1), la cual posteriormente se explica detalladamente (Xiaoyang, L. et al., 2012).



Figura 1. Diseño de algoritmo.

A. Obtención de la imagen

Para obtener la imagen a procesar, primeramente se pondrá una muestra con polen en el microscopio óptico, y mediante la videocámara se obtendrá una imagen a procesar, la cuál será enviada a la tarjeta por medio del jack de video.

B. Almacenamiento en SDRAM

Una vez obtenidas las imágenes son almacenadas en la memoria SDRAM, aquí son almacenadas en escala de grises, asignando un valor correspondiente a cada pixel de la matriz que esta entre 0 y 255, y con formato de imagen bmp (Archivo de mapa de bits) (Woods, R. et al., 2009).

En la Fig. 2 se muestra la espora de polen que se considera para el desarrollo de esta prueba.

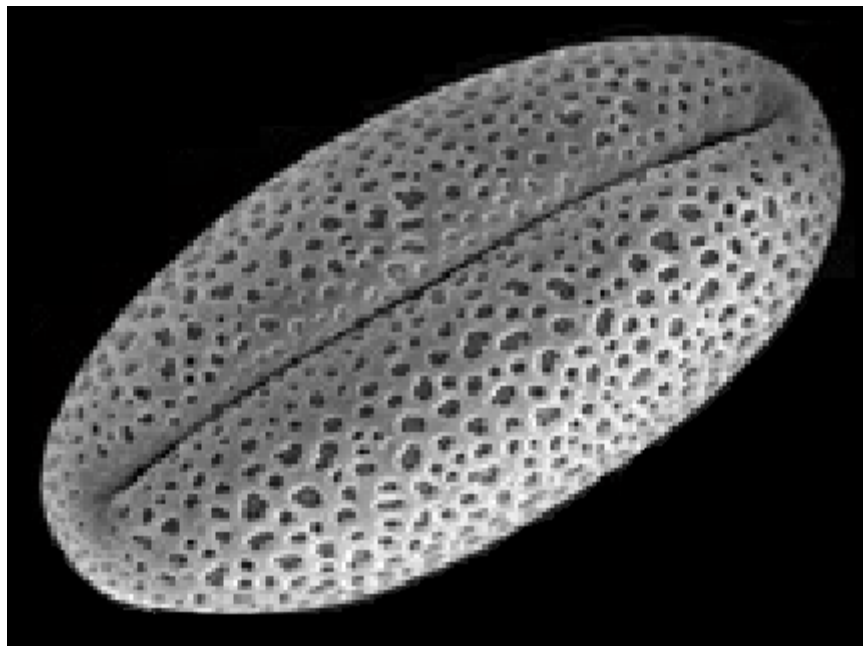


Figura 2. Espora de polen *Lilium auratum* a procesarse en escala de grises.

C. Detección de bordes

Una vez almacenadas las imágenes en la memoria SDRAM, el Softcore embebido procede con la rutina de lectura a la memoria SDRAM, para extraer la información de la imagen y aplicar el algoritmo de detección de bordes.

Mediante el detector de bordes de Canny, se aplica el filtro Gaussiano para eliminar todo el ruido y obtener un suavizado de la imagen.

El objetivo del suavizado es el de aumentar o disminuir ciertas características que se encuentran presentes en la imagen, el filtro que hemos usado está basado en la convolución de la media aritmética que tiene como fin reducir la presencia de ruido, es decir, eliminar el componente de altas frecuencias y mantener las bajas Fig. 3.

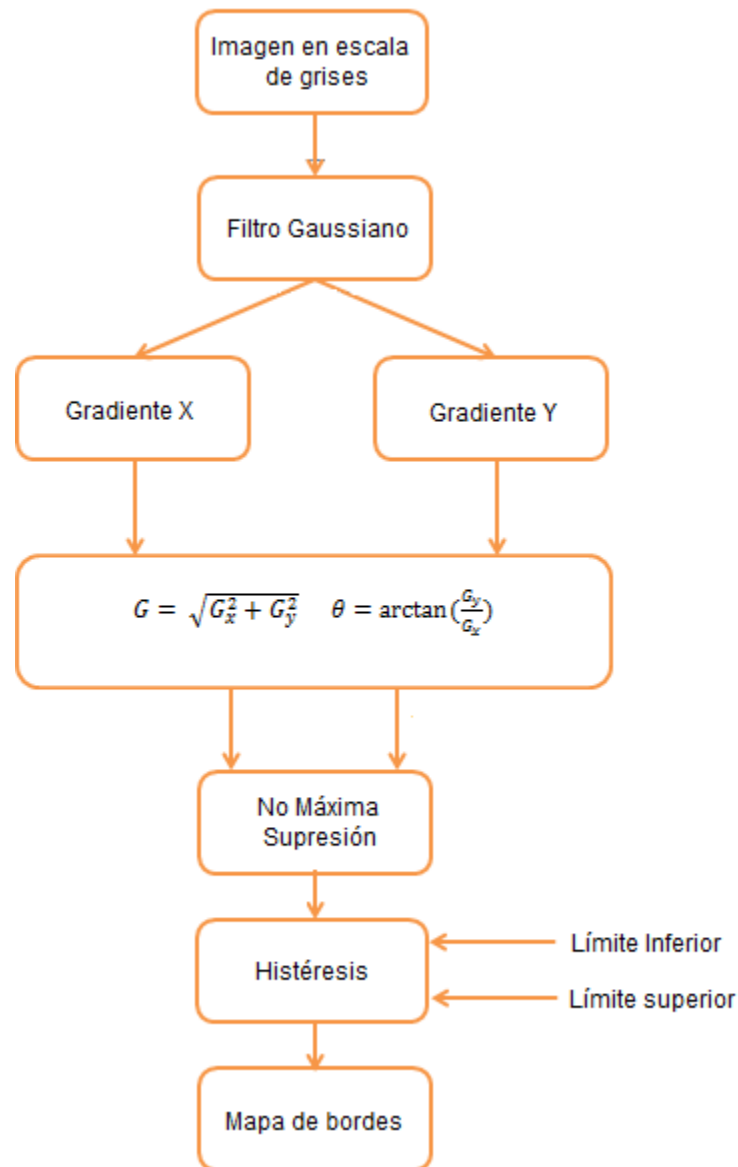


Figura 3. Diagrama de flujo del detector de bordes Canny.

Diagrama de flujo del detector de bordes Canny.

Ésta imagen será la entregada por el algoritmo de detección de bordes que está implementado en el Softcore NIOS II, para posteriormente desarrollar las operaciones correspondientes a los pasos siguientes.

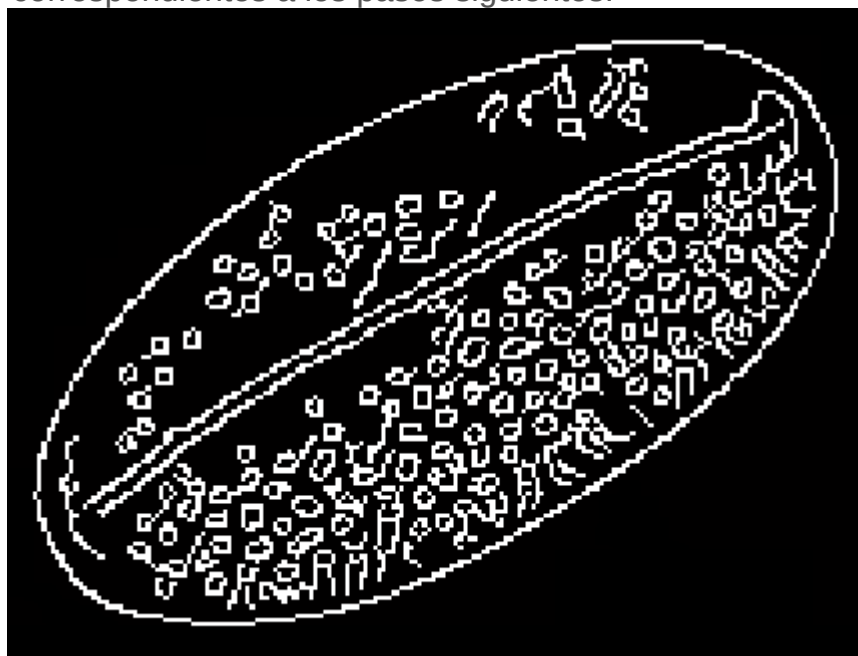


Figura 4. Imagen procesada por el detector de bordes Canny.

D. Firma y clasificación

Una vez obtenida la imagen procesada del detector de bordes de Canny, se procede a la extracción de la firma de la imagen y clasificación.

Por el tipo de objeto se seleccionaron dos clases de descriptores; morfológicos y de textura. Dentro de los descriptores morfológicos se encuentra el área, el perímetro, la circularidad y la firma. Y como descriptor de textura se consideró el histograma de cada partícula de polen y la densidad superficial que combina la distribución del nivel de gris y el tamaño mediante el cociente entre la suma de las intensidades de todos los pixeles pertenecientes al objeto y el área del mismo (Morales, J. et al., 2012).

Por el tipo de objeto se seleccionaron dos clases de descriptores; morfológicos y de textura. Dentro de los descriptores morfológicos se encuentra el área, el perímetro, la circularidad y la firma. Y como descriptor de textura se consideró el histograma de cada partícula de polen y la densidad superficial que combina la

distribución del nivel de gris y el tamaño mediante el cociente entre la suma de las intensidades de todos los píxeles pertenecientes al objeto y el área del mismo (Morales, J. et al., 2012).

Con la imagen de la Fig. 4 procedemos a desarrollar la extracción de la forma, para así poder desarrollar después la extracción de la firma y hacer una base de datos a donde serán almacenadas las firmas en grupos de cada una de las esporas de polen, cabe aclarar que existen varias firmas de cada espora de polen para evitar errores en la clasificación y conteo.

A la imagen de la Fig. 4 se le aplica un filtrado de píxeles, el cual excluye todo píxel dentro de la espora, dando como resultado la Fig. 5, haciendo esto para reducir el procesamiento de datos al momento de extraer la firma y evitar también errores durante el procesamiento.

Con base a lo mencionado anteriormente, se procede a extraer la firma a partir de la forma obtenida, se utiliza el histograma para mostrar de forma gráfica la firma de cada una de las esporas de polen, y así albergar todas las muestras en la base de datos que se encarga de la clasificación y conteo de las partículas. Partiendo de lo antes mencionado, obtenemos una firma para la espora de muestra, la cual se puede observar en la Fig. 6 (Rocha, A. et al., 2010).

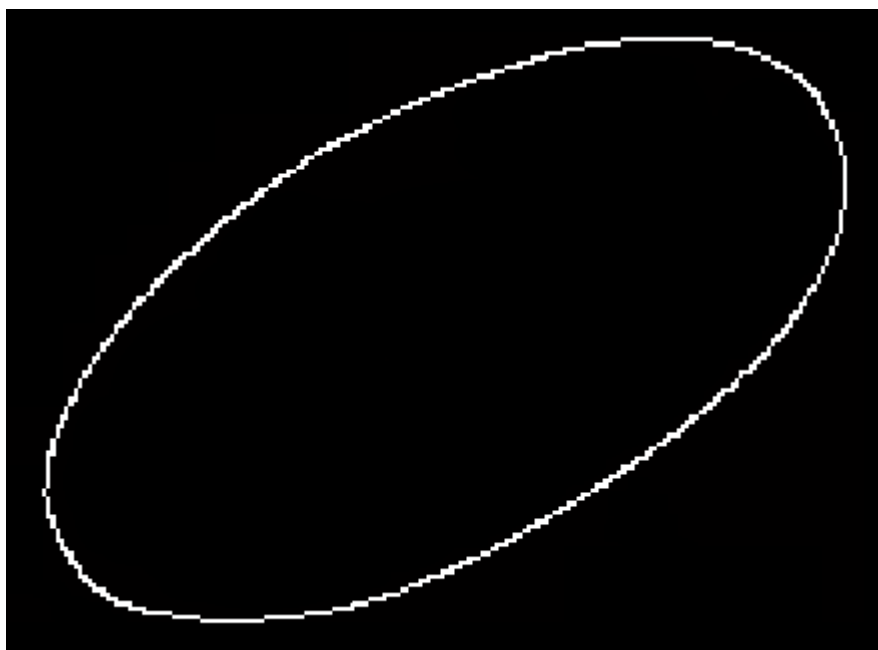


Figura 5. Imagen procesada después de aplicar la exclusión de píxeles.

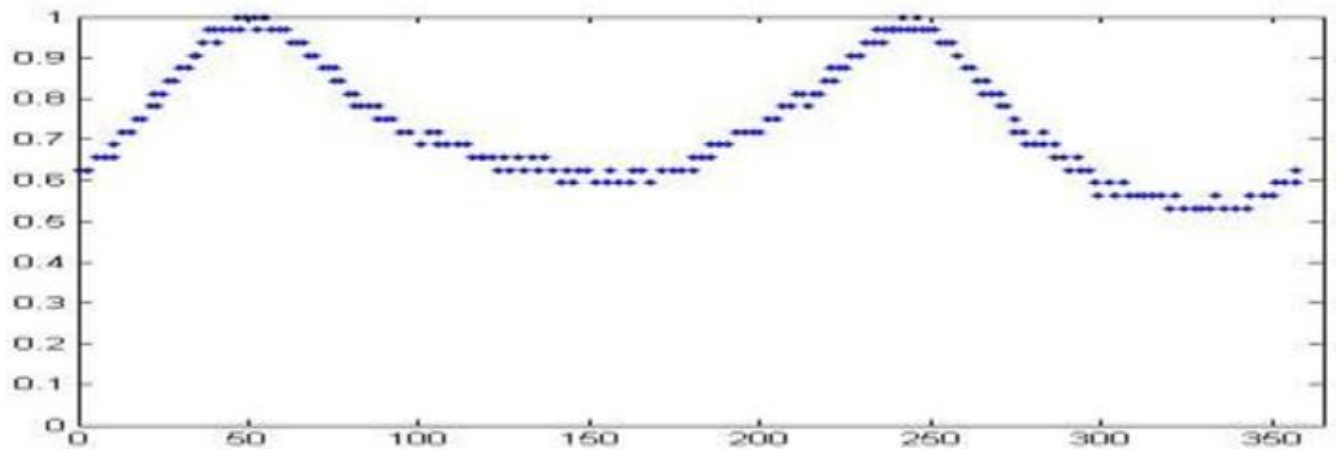


Figura 6. Firma correspondiente a la espora de muestra.

E. Almacenamiento en SRAM

Una vez desarrolladas las etapas anteriores, el Softcore embebido es el encargado de enviar los datos a sus respectivas direcciones, como lo es el caso de la imagen ya procesada por el detector de Canny, la cual será guardada en la SRAM.

F. Resultados en monitor VGA

Posteriormente se leen los datos de la memoria SRAM para enviarse al Puerto VGA y de esta manera procesarse para mostrarlos en un monitor, con el fin de que el operador pueda analizar de forma gráfica el tipo de espora, con la cual un paciente está intoxicado, y de esta forma suministre el tratamiento adecuado. En el caso de estar realizando labores de investigación, podrá analizar más a detalle cada una de las esporas, también tiene visible el conteo y la clasificación de cada una de las esporas existentes en la muestra.

7. Resultados

Como primer paso en esta metodología de diseño, se prueban los algoritmos a implementar en el software MatLab, esto con el motivo de probar su veracidad, y obtener buenos resultados, los cuáles servirán posteriormente como base para realizar las comparaciones pertinentes con los resultados obtenidos en hardware.

En la Fig. 7 se procede a mostrar los resultados de estas implementaciones (Cuevas, E. et al., 2010). Tras haber realizado la implementación de los diferentes detectores de bordes, como se muestra en la Fig. 6, se presentan los resultados, analizando el desempeño y veracidad de cada uno.

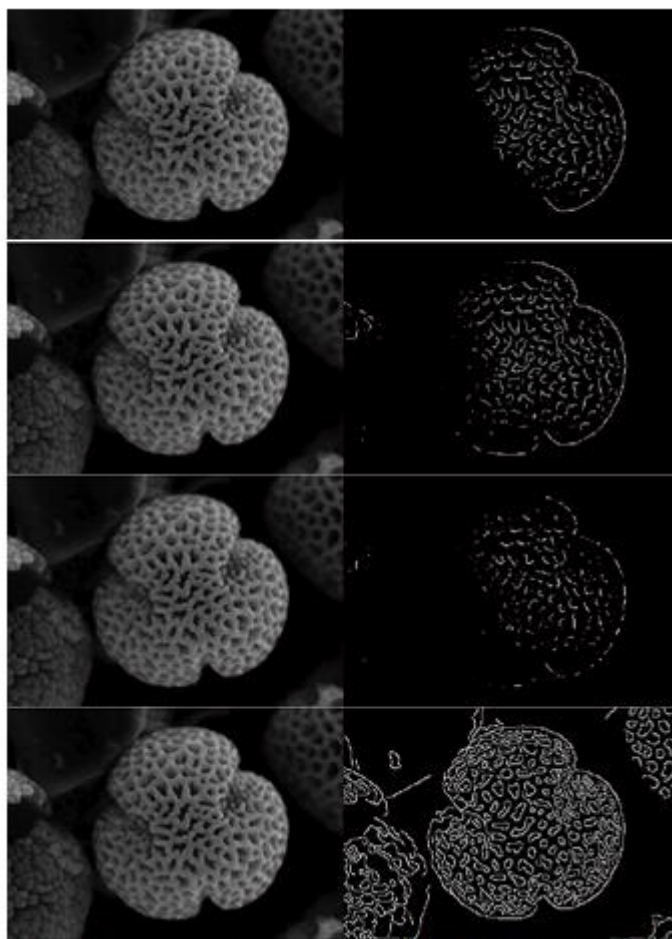


Figura 7. Imagen original a la izquierda, imagen procesada mediante detector de bordes a la derecha, primeramente se observa el detector de bordes de Prewitt, posteriormente el de Sobel, después el de Roberts y finalmente el de Canny.

Primeramente tenemos que para el primer detector de bordes, que es el de Prewitt, hay pérdida de información en la parte de las orillas de la imagen, ya que existen diversas esporas alrededor y no las detecta. El segundo detector de bordes, que es el de Sobel, presenta menor pérdida de información, sin embargo no alcanza a detectar totalmente las esporas que tiene en sus bordes. Posteriormente se presenta el detector de bordes de Roberts, el cual presenta

pérdida de información pero menor en comparación con el de Prewitt. Finalmente se presenta el detector de bordes de Canny, el cual presenta mayor desempeño ya que tiene menor pérdida de datos y muestra únicamente los bordes de la imagen que son de utilidad y los realza, lo demás que el filtro considera despreciable, simplemente es desechado, como se muestra, rescata la información de las esporas que tiene a su alrededor, además de brindar una buena reconstrucción de la espora central.

Por ello es que se implementa dicho detector de bordes, esto con el fin de alcanzar mejores resultados desde el inicio del procesado, recordando que ésta etapa de detección es la inicial, y al obtener los mejores resultados desde un inicio se optimizarán los procesos siguientes, como son: la exclusión de píxeles, firma, numeración y clasificación, obteniendo así una serie de resultados favorables, que le serán de gran utilidad al usuario. Por otro lado se desarrolla una estimación de tiempos, y se concluye que gracias al procesamiento paralelo de la FPGA, el tiempo de respuesta deberá ser menor con base al del software, ya que gracias al Softcore utilizado como bloque principal de control, nos queda un reducido bloque de hardware que a su vez tienen la función de ser bloques aceleradores de procesos, debido a esto se hacen reducciones de tiempos considerables, desarrollando de ésta manera todas las tareas que conforman el proyecto.

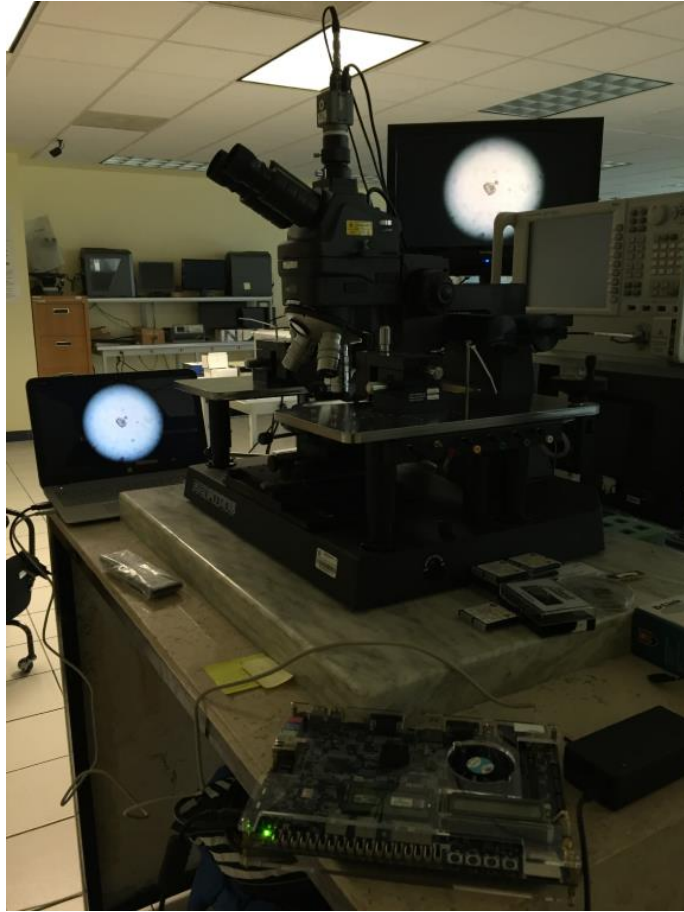


Figura 8. Estación de pruebas analizando una espora de polen.

En la Fig. 8 se muestra la estación de pruebas en donde se realizan las labores de visualización de las esporas de polen, extracción de imagen de las mismas y procesamiento de datos tanto en el dispositivo reconfigurable como en la PC. Se realizan las tareas de visualización y extracción de la imagen, siendo esta misma procesada para posteriormente almacenarse en el FPGA.

A continuación, en la Fig. 9 se muestra un diagrama de la arquitectura del sistema implementado.

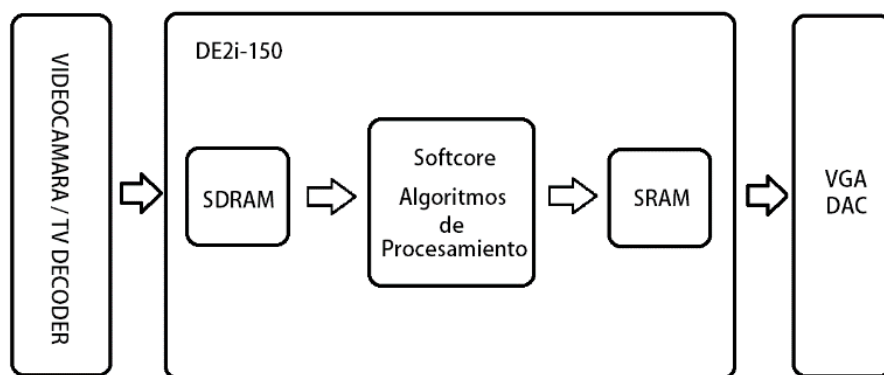


Figura 9. Implementación del Sistema.

En la Fig. 9 se describe la operación del sistema de forma detallada, analizando la entrada de los datos a partir de la videocámara, posteriormente su procesamiento en la tarjeta de desarrollo y el despliegado de datos en el monitor VGA en forma de imagen.

La síntesis del sistema se realizó mediante la herramienta del software Quartus II y se obtuvo el resultado de ocupación, el cual se muestra en la Tabla 1.

Elementos utilizados	Ocupación	Porcentaje
Elementos lógicos	4,460 / 33,216	13 %
F. combinacionales	4,213 / 33,216	13%
Registros lógicos	2,904 / 33,216	9%
Registros totales	3000	-
Pines totales	94/475	20%
Bits de memoria	242,712 / 483,840	50%
PLLs totales	1 / 4	25%

Tabla 1. Resultados de ocupación de área para el sistema desarrollado.

Finalmente se muestra en la Fig. 10 el comportamiento de la primera parte del sistema, la cual consiste en la escritura de los datos adquiridos por la videocámara en la memoria SDRAM, y su salida hacia el Softcore, para posteriormente procesarse.

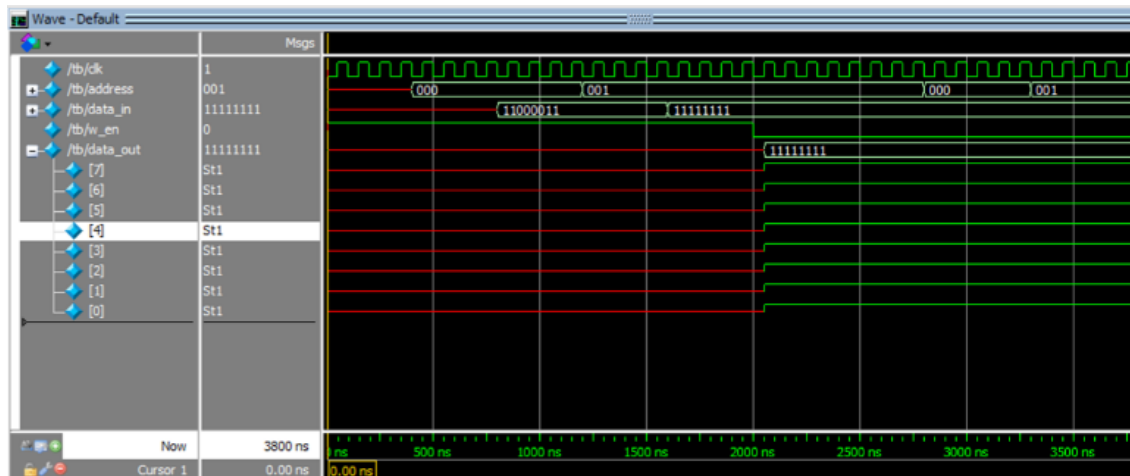


Figura 10. Comportamiento en tiempo de la escritura y lectura de datos en la memoria SDRAM incorporada en el sistema.

Se puede observar con detalle que a los 2000 ns se inician y sincronizan las señales de lectura y escritura de la memoria SDRAM a utilizar.

8. Conclusión

En este trabajo, se propone una metodología en hardware para la detección de micro partículas, que está compuesta de una arquitectura de hardware a la medida para la detección de bordes en imágenes, así como también de un diseño clasificador de micro partículas de polen, junto con un bloque que realiza las tareas de conteo de las mismas, todo esto se ha realizado con base en una FPGA. Mediante el uso del detector de Canny que es un filtro para mejorar la mediana de la imagen, el rendimiento se mejora significativamente cuando la imagen es liberada de la contaminación por ruido. La velocidad de procesamiento se mejorará y el consumo de hardware se reduce. El diseño se realiza en un Cyclone IV FPGA de Altera, los resultados experimentales y el consumo de tiempo indican que esta arquitectura de hardware es suficiente para aplicaciones que incorporen procesamiento de imagen.

RECONOCIMIENTO

Los autores agradecen el apoyo financiero proporcionado por el CONACYT (Consejo Nacional de Ciencia y Tecnología) como una beca de maestría.

Referencias

Acharya, Tinku., Ray, A. K. (2005). Image processing: principles and applications. John Wiley & Sons

Cuevas, Eric., Zaldívar, D., Pérez, M. (2010) "Procesamiento Digital De Imágenes Con Matlab Y Simulink" Alfaomega-Ra-Ma.

Gonzalez, Rafael., Woods, R., Eddins, S. (1992) "Digital Image Processing". Addison-Wesley.

Jorge Osio; Walter Aróztegui; Jose Rapallini; Antonio Adrián Quijano; Jesús Ocampo (2011) "Desarrollo de Algoritmos de Procesamiento de Imágenes Basados en Operadores de Ventana sobre una FPGA." Primeras Jornadas de Investigación y Transferencia, UNLP.

Morales, G.J. (2012) "Diseño de un sistema de reconocimiento y cuantificación de partículas de polen con dispositivos reconfigurables." Tesis de Doctorado Universidad de Guadalajara.

Rajbahadur Singh Raghuvanshi , Ashutosh Datar (2013) "Comparison of Gradient Operator based Pseudocolored Enhanced Medical Images." ; Fourth International Conference on Computing, Communications and Networking Technologies (ICCCNT)

Raman Maini, Himanshu Aggarwal, (2009) "Study and comparison of various image edge detection techniques," Int. Journal of Image Processing.

Rocha, P.A. (2010) "Diseño de un sistema embebido basado en un microprocesador de núcleo flexible con dispositivos reconfigurables." Tesis de Doctorado Universidad de Guadalajara.

Rodriguez, Roberto., Sossa, Juan. (2012) "Procesamiento digital de imágenes" Alfaomega-Ra-Ma.

S. Sowmya, R. Paily. (2011) "FPGA implementation of image enhancement algorithms," Int. Conf. Communications and Signal processing.

Wenhao He, Kui Yuan. (2008) "An improved Canny edge detector and its realization on FPGA," World Congress on Intelligent Control and Automation.

Woods, Richard., Eddins, S., Gonzalez, R. (2009). "Digital image processing using Matlab." Pearson Education.

Xiaoyang Li, Jie Jiang, Qiaoyun Fan (2012) "An Improved Real-time Hardware Architecture for Canny Edge Detection Based FPGA", Third International Conference on Intelligent Control and Information Processing.



Juan R. Varela recibió el grado de Licenciatura en Ingeniería en Comunicaciones y Electrónica en el Instituto Tecnológico de Oaxaca, Oaxaca, México. Es un estudiante de Maestría en Diseño Electrónico en el Centro de Investigación y de Estudios Avanzados del Instituto Politécnico Nacional Unidad Guadalajara, Jalisco, México. Su área de investigación es la de procesamiento digital de imagen.



Susana Ortega Cisneros Recibió su grado de licenciado en ciencias en Comunicaciones y Electrónica de la Universidad de Guadalajara,

México en 1990, el grado de maestría del Centro de Investigación y de Estudios Avanzados del IPN, Zacatenco, México. Susana Ortega recibió su grado de doctor en Ciencias de la Computación y Telecomunicaciones de la Universidad Autónoma de Madrid, España. Se especializa en el diseño digital de arquitecturas basadas en FPGAs, DSPs y microprocesadores. Las principales líneas de investigación en las cuales trabaja son: Control Digital, Sincronización Self-Timed, Sistemas Electrónicos Aplicados a la Biomedicina, Diseño de Microprocesadores Embebidos, electrónica digital y DSPs personalizados en FPGAs.



Juan José Raygoza recibió el grado de Licenciado en Ingeniería en Comunicaciones y Electrónica en la Universidad de Guadalajara, México en 1989, el grado de Maestro en el Centro de Investigación y de Estudios Avanzados del IPN; Zacatenco, México. Juan José Raygoza recibió el grado de Doctor en Ciencias de la Computación y Telecomunicaciones de la Universidad Autónoma de Madrid, España. De 1996 al 2000, trabajó en IBM, participó en la transferencia tecnológica de la fabricación de cabezales de discos duros, en la planta de manufactura de IBM en San José C.A a Guadalajara, México. Se especializa en el diseño de arquitecturas digitales basadas en FPGAs, Microprocesadores, sistemas embebidos y bioelectrónica. Las principales líneas de investigación en las que trabaja son: Sistemas Electrónicos aplicados a la Biomedicina, Diseño de Microprocesadores, Control Digital, Sistemas Embebidos.



Jorge Rivera nació en El Rosario, Sinaloa, México en 1975. Recibió su grado de licenciado en ciencias del Instituto Tecnológico del Mar, Mazatlán, México en 1999, y los grados de maestro en ciencias y doctorado en ingeniería eléctrica del Centro de Investigación y de Estudios Avanzados del Instituto Politécnico Nacional, Guadalajara, México en el 2001 y 2005, respectivamente. Desde 2006, ha estado en la Universidad de

Guadalajara, Guadalajara, México, como profesor a tiempo completo en el departamento de ingeniería electrónica. Actualmente es comisionado como profesor CONACyT para el CINVESTAV, IPN, campus Guadalajara. Sus intereses de investigación se enfocan en teoría de reguladores, control de modo deslizante, sistemas de control no lineales en tiempo discreto, y sus aplicaciones a las maquinas eléctricas. Ha publicado más de 40 artículos técnicos en revistas internacionales y conferencias, y ha servido como revisor para diferentes revistas internacionales y conferencias.



Adrian Pedroza se recibió de la Licenciatura en Ingeniería en Comunicaciones y Electrónica en la Universidad de Guadalajara, Jalisco, México, en el 2008. En el 2010 recibió el grado de Maestro en Ciencias en Ingeniería Electrónica y Computación en la Universidad de Guadalajara. Del 2010 al 2011 trabajó en Intel como Ingeniero de Diseño de Componentes. En 2015 recibió el grado de Doctor en Ciencias en Ingeniería Eléctrica en el Centro de Investigación y Estudios Avanzados del IPN Unidad Guadalajara, Jalisco, México. Hoy en día su investigación se centra en el área de aceleradores con arquitecturas de memorias en paralelo para procesamiento digital de imágenes y verificación de hardware.



Esta obra está bajo una licencia de Creative Commons
Atribución-NoComercial-CompartirIgual 4.0 Internacional.