



**INAOE**

# **Análisis morfológico de semillas con redes memristivas**

por

**Ing. Marco Antonio Zamudio  
Hernández**

Tesis sometida como requisito parcial para  
obtener el grado de

**MAESTRO EN CIENCIAS EN LA  
ESPECIALIDAD DE ELECTRÓNICA**

en el

**Instituto Nacional de Astrofísica, Óptica y  
Electrónica**

Febrero 2020

Tonantzintla, Puebla

Supervisada por:

**Dr. Librado Arturo Sarmiento Reyes**

Investigador Titular INAOE

**Dr. Héctor Vázquez Leal**

Investigador Titular UV

©INAOE 2020

El autor otorga al INAOE el permiso de  
reproducir y distribuir copias en su totalidad o en  
partes de esta tesis





# **Análisis Morfológico de semillas con redes memristivas**

Tesis de Maestría

POR:

**Ing. Marco Antonio Zamudio Hernández**

ASESORES:

**Dr. Librado Arturo Sarmiento Reyes**

**Dr. Héctor Vázquez Leal**

Instituto Nacional de Astrofísica Óptica y Electrónica  
Coordinación de Electrónica



# Agradecimientos

---

La culminación del presente trabajo es una etapa en mi vida, la cual me permitió conocer nuevas personas en el ámbito académico que dejan una huella y sabiduría en mi persona. Antes que nada, quiero agradecer a Dios por haberme dado salud y capacidad para el desarrollo del presente trabajo.

A mi familia, pero en especial a mi madre María felix quien siempre me apoyo incondicional a lo largo de mi carrera profesional.

Al Dr. Arturo Sarmiento, un agradecimiento muy especial por la asesoría del presente trabajo, su ayuda, paciencia y observaciones que estuvieron orientadas a la culminación de un buen trabajo.

Agradezco al Consejo Nacional de Ciencia y Tecnología (CONACYT) por la beca otorgada para poder llevar a cabo este proyecto y por la nacionalización de mis estudios como maestro en ciencias.



# Resumen

---

El análisis morfológico de semillas es una tarea esencial para el estudio y clasificación de las mismas, así como para aplicaciones fitosanitarias posteriores tales como la detección de malezas. En este trabajo, el análisis de la morfología de las semillas se realiza mediante una red memristiva que es utilizada como procesador analógico para la detección de bordes en imágenes provenientes de una base de datos. Los memristores de la red memristiva son modelados por una expresión de memristancia en función de la carga. Además, la red es implementada en una versión paralela con un esquema GPU. La solución de la red se obtiene mediante un análisis en el dominio del tiempo de las ecuaciones diferenciales resultantes de la formulación nodal.

Un primer paso, con el objetivo de señalar las ventajas de la implementación paralela con respecto a su contraparte serial, se ha tratado un conjunto de imágenes para la detección de bordes. Los resultados no solo cumplen con los aspectos cualitativos, sino también con las principales figuras de mérito. Además, la red memristiva paralela muestra una sensible ganancia en los tiempos de ejecución en comparación con la red serial.

La meta principal de este trabajo, i.e. el análisis de la morfología de semillas se logra recurriendo a una serie de medidas en los bordes del modelo geométrico de la semilla. Para ello se introduce un modelo elíptico, el cual es válido tanto para dicotiledóneas como monocotiledóneas, las principales dimensiones son usadas para establecer un grupo de figuras de mérito que definen la morfología de las semillas. Finalmente, se presentan varios casos de estudio.



# Abstract

---

Seed morphology analysis is an essential task for studying and classifying seeds, and for any ulterior phytosanitary application such as weed detection. In this work, the analysis of the morphology of seeds is done by means of a memristive grid which is used as an analog image processor in order to achieve edge detection on images from a seed database. The memristors within the grid are modeled by a memristance expression that is given as a charge-controlled function. Besides, the grid is implemented in a parallelized version with a GPU scheme. The solution of the grid is obtained by a time-domain analysis of the resulting differential equations from the nodal analysis.

In a first step, with the aim of highlighting the advantages of such the parallel implementation with respect to its serial counterpart, a set of benchmark images has been treated for edge detection. Not only are the qualitative aspects of the results kept, but also the main quantitative merit figures are almost identical. On the top of this, the parallel memristive grid exhibits a sensible gain in execution times in comparison with the serial grid.

The main goal of this work, i.e. the analysis of the morphology of seeds is achieved by resorting to a series of measures on the seed edges. A geometric model of the seed is introduced in the form of an ellipse that is valid for monocots as well as dicots, and its main dimensions are used for establishing a group of merit figures that define the morphology. Finally, several case studies are presented.



# Tabla de Contenido

---

Agradecimientos	I
Resumen	III
Abstract	V
Índice de Figuras	IX
Índice de Tablas	XI
<b>1. Introducción</b>	<b>1</b>
1.1. Objetivos . . . . .	2
1.2. Hipótesis . . . . .	2
1.3. Justificación . . . . .	2
1.4. Metodología . . . . .	3
1.5. Estructura de la tesis . . . . .	4
<b>2. Procesamiento en paralelo de la red memristiva</b>	<b>5</b>
2.1. Red Memristiva . . . . .	6
2.2. Ecuaciones nodales de la red memristiva . . . . .	7
2.3. Modelo del memristor . . . . .	9
2.4. Red memristiva implementada en paralelo . . . . .	13
2.4.1. Concepto de jerarquía . . . . .	15
2.5. Solución de la red memristiva en paralelo . . . . .	18
2.6. Resultados . . . . .	20
2.6.1. Detección de bordes . . . . .	21
2.6.2. Detección de bordes en función del umbral . . . . .	26

---

2.7. Conclusiones . . . . .	27
<b>3. Estudio morfológico de semillas</b>	<b>29</b>
3.1. Conceptos de Botánica . . . . .	29
3.2. Modelos geométricos de semilla . . . . .	31
3.2.1. Redondez y Compacticidad . . . . .	34
3.2.2. Comparación entre ovoide y elipse . . . . .	35
3.3. Extracción de parámetros morfológicos . . . . .	37
3.3.1. Demostración para diferentes orientaciones . . . . .	39
3.4. Casos de estudio . . . . .	42
3.4.1. Vicia Sativa . . . . .	43
3.4.2. Trifolium Subterraneum. . . . .	45
3.4.3. Crotalaria Zanzibarica . . . . .	48
3.4.4. Frijol . . . . .	49
3.4.5. Vaccaria Hispanica . . . . .	51
<b>4. Conclusiones y Trabajo Futuro</b>	<b>55</b>
<b>Bibliografía</b>	<b>57</b>

# Índice de Figuras

---

2.1. Red resistiva. . . . .	6
2.2. Red memristiva. . . . .	7
2.3. KCL para nodo $i, j$ . . . . .	7
2.4. Escalas de grises para el voltaje de entrada $V(s_{i,j})$ . . . . .	8
2.5. muñeca Matryoshkas. . . . .	11
2.6. $M-q$ curvas para el fusible memristivo. . . . .	12
2.7. a) Característica ideal para la carga de la memristancia, b) Característica del modelo desarrollado de la carga de la memristancia . . . .	13
2.8. Estructura de una computadora convencional. . . . .	14
2.9. Modelo de ejecución de un kernel. . . . .	16
2.10. (a) Hilos como vector. (b) Hilos con estructura matricial. . . . .	17
2.11. Distribución de los hilos sobre los píxeles de la imagen. . . . .	19
2.12. Esquema de nodos paralelizados en la red memristiva. . . . .	20
2.13. Imagen de salida para diferentes tiempos de suavizado. . . . .	21
2.14. (a) Escala de gris. (b) Bordes. . . . .	22
2.15. Graficas precision-recall. . . . .	23
2.16. Grafica de tiempo de CPU vs GPU. . . . .	24
2.17. Extracción de bordes. . . . .	25
2.18. Diferentes porcentaje en umbral para la detección de bordes. . . . .	26
3.1. Ejemplos de Monocotiledonea (maíz) y Dicotiledonea (frijol). . . . .	30
3.2. Modelo ovoidal . . . . .	32
3.3. Modelo elipsoidal . . . . .	33
3.4. Modelo elipsoidal (azul) y modelo ovoidal (rojo). . . . .	36
3.5. Elipse con hilo para modelar la semilla con hilo. . . . .	37
3.6. Semillas rotadas con diferentes ángulos ( $\theta$ ). . . . .	40

---

3.7. <i>Vicia Sativa</i> . . . . .	43
3.8. <i>Vicia Sativa</i> : bordes. . . . .	43
3.9. <i>Trifolium Subterraneum</i> . . . . .	45
3.10. <i>Trifolium Subterraneum</i> : bordes. . . . .	46
3.11. <i>Crotalaria Zanzibarica</i> . . . . .	48
3.12. <i>Crotalaria Zanzibarica</i> : bordes. . . . .	48
3.13. Frijol. . . . .	50
3.14. Frijol: bordes. . . . .	50
3.15. <i>Vaccaria Hispanica</i> . . . . .	52
3.16. <i>Vaccaria Hispanica</i> : bordes. . . . .	52

# Índice de Tablas

---

2.1. Selección de operador. . . . .	11
2.2. Valores nominales de parámetros. . . . .	12
3.1. Parámetros usados para la descripción del ovoide y elipse. . . . .	35
3.2. Características morfológicas de ambos modelos para los valores de la <i>tabla 3.1.</i> . . . . .	36
3.3. Error en porcentaje entre ambos modelos. . . . .	37
3.4. Parámetros usados para semillas. . . . .	39
3.5. Resultados de la semilla obtenidos por MATLAB y fórmula. . . . .	41
3.6. Resultados del hilo obtenidos por MATLAB y fórmula. . . . .	41
3.7. Error relativo de área y perímetro de la semilla. . . . .	41
3.8. Error relativo de área y perímetro del hilo. . . . .	42
3.9. Parámetros morfológicos para modelos elípticos rotados. . . . .	42
3.10. Parámetros morfológicos de Vicia Sativa. . . . .	43
3.11. Redondez y compacticidad para Vicia Sativa. . . . .	44
3.12. Características geométricas del hilo de Vicia Sativa. . . . .	44
3.13. Características geométricas de la semilla de Vicia Sativa. . . . .	44
3.14. Parámetros morfológicos de Trifolium Subterraneum. . . . .	46
3.15. Redondez y compacticidad para Trifolium Subterraneum. . . . .	47
3.16. Características geométricas del hilo de Trifolium Subterraneum (para dos vistas). . . . .	47
3.17. Características geométricas de la semilla de Trifolium Subterraneum. . . . .	47
3.18. Parámetros morfológicos de Crotalaria Zanzibarica. . . . .	48
3.19. Redondez y compacticidad para Crotalaria Zanzibarica. . . . .	49
3.20. Características geométricas del hilo de Crotalaria Zanzibarica. . . . .	49
3.21. Características geométricas de la semilla de Crotalaria Zanzibarica. . . . .	49

---

3.22. Parámetros morfológicos del frijol. . . . .	51
3.23. Redondez y compacticidad para el frijol. . . . .	51
3.24. Características geométricas del hilo de frijol. . . . .	51
3.25. Características geométricas de la semila de frijol. . . . .	51
3.26. Redondez y compacticidad para <i>Vaccaria Hispanica</i> . . . . .	53
3.27. Características geométricas de la semilla de <i>Vaccaria Hispanica</i> . . . . .	53

---

## Capítulo 1

# Introducción

---

El memristor, catalogado como el cuarto elemento básico de Teoría de Circuitos, tiene aplicaciones en procesamiento analógico especialmente cuando se utiliza para constituir arreglos rectangulares en forma de redes memristivas. Las redes memristivas permiten llevar a cabo en realidad una forma de procesamiento analógico masivo, especialmente cuando la solución de la red es implementada en paralelo para resolver sus ecuaciones resultantes. En este trabajo se utiliza un modelo que tiene una función constitutiva de rama que es controlada por carga, i.e. su memristancia está expresada como  $M(q)$  [1]. La solución de la red involucra resolver  $n \times m$  ecuaciones diferenciales en el dominio del tiempo al establecer la Ley de Corrientes de Kirchhoff. Este proceso de solución es llevado a cabo en forma paralela bajo un esquema de GPU, lo que deviene en un notable ahorro en el tiempo de solución de la red comparada con la implementación serial.

En este trabajo, la red memristiva es aplicada a la detección de bordes de imágenes de semillas con la finalidad de determinar sus parámetros morfológicos. Esto reviste gran importancia para futuras aplicaciones fitosanitarias encaminadas a la detección de malezas, pero lo más relevante es que implica el trayecto del memristor desde el mero concepto [2], pasando por el modelado matemático del mismo para utilizarlo en la conformación de redes memristivas [3] y su final aplicación a la solución de un problema práctico que además representa un tópico de un área diferente a la electrónica.

## 1.1. Objetivos

El objetivo de esta tesis es determinar los parámetros morfológicos de semillas utilizando una red memristiva.

Los objetivos específicos se pueden listar como:

- Implementar la red memristiva en paralelo, donde los memristores son modelados por una función de carga eléctrica  $M(q)$ .
- Por ende reducir el tiempo de procesamiento para la detección de bordes a través de comparar la implementación paralelo con la implementación en serie.
- Analizar los parámetros morfológicos para imágenes de semillas.

## 1.2. Hipótesis

La hipótesis del presente trabajo de investigación puede ser dividida en dos argumentos iniciales, el primero de ellos relacionado con la herramienta utilizada para realizar el análisis morfológico de las semillas, y el segundo corresponde al objeto de estudio, i.e. la semilla misma.

La hipótesis sobre la herramienta (la red memristiva) se puede expresar como: la detección de bordes de imágenes puede ser realizada por una implementación en paralelo de la red memristiva. La segunda hipótesis consiste en expresar que es posible aplicar la detección de bordes a imágenes de semillas y a partir de ello obtener los parámetros morfológicos más importantes.

## 1.3. Justificación

Las semillas han pasado de ser consideradas un elemento puramente biológico a un bien de consumo y explotación, especialmente en semillas para alimentación y energía (generación de combustible), lo cual involucra actividades de transporte y por tanto de mayor dispersión geográfica.

Es bajo el contexto de esta mayor dispersión que el actual trabajo se presenta, pues se requiere que la importación y exportación de semillas cumplan con diversas normas fitosanitarias establecidas por los gobiernos y organismo internacionales con la finalidad de evitar la entrada de semillas nocivas (malezas) para el país.

Con el análisis morfológico de semillas, se tiene un punto de partida para un proyecto de mayor alcance, el cual se debe orientar a la detección de malezas. Pues, en un lote de semillas o granos, toda semilla extraña a las semillas de cultivo es considerada como contaminante, incluyendo semillas de otros cultivos. Sin embargo, para regulaciones fitosanitarias, solo son de interés aquellas consideradas como de importancia cuarentenaria, i.e. aquellas que tienen prohibido su ingreso — hablando de importación de granos. Esto es obviamente válido para la exportación de semillas.

## 1.4. Metodología

La metodología se puede precisar en los siguientes pasos:

- Implementación de la red memristiva en paralelo para:
  - Suavizado de imágenes
  - Detección de bordes
- Comparar el rendimiento de la red memristiva paralelizada vs la red memristiva en serie.
- Selección del modelo elíptico para aproximar los odes de semillas.
- Determinar los parámetros morfológicos de varios casos de estudio.

En el capítulo 2 se presentara una red memristiva en serie y paralela, donde se observa si tiene alguna diferencia o ventajas entre ellas, además se observa diferentes porcentaje de umbral para definir los bordes. En el capítulo 3 se presentan 2 modelos geométricos para poder comparar con la forma que presentan las semillas y con la selección de un modelo geométrico se presentan diferentes casos de estudio, finalmente en el capítulo 4 se presenta las conclusiones y trabajo futuro.

## 1.5. Estructura de la tesis

El capítulo 2 trata de la implementación en paralelo de la red memristiva. Se pone especial atención a las propiedades del modelo del memristor, el cual está definido por una función constitutiva de rama  $M(q)$ . Adicionalmente, se realiza un estudio comparativo para determinar las ventajas de la implementación en paralelo contra la implementación en serie. No sólo los aspectos cualitativos, sino también los aspectos cuantitativos son tomados en cuenta para llevar a cabo dicha comparación.

El capítulo 3 se enfoca primeramente a determinar un modelo geométrico para aplicar al estudio de semillas. Posteriormente se introducen los parámetros morfológicos más relevantes. Adicionalmente, se analizan y reportan varios casos de estudio.

Finalmente, el capítulo 4 presenta las conclusiones más relevantes del trabajo y además se señalan algunas posibles líneas de trabajo futuro.

# Procesamiento en paralelo de la red memristiva

---

En el área de procesamiento de imágenes, la detección de los bordes de una imagen es de suma importancia y utilidad; dado que, facilita muchas tareas, entre ellas, el reconocimiento de objetos, reconocimiento de patrones, segmentación de regiones, reconocimiento de trayectorias, reducción de ruido, fusión de imágenes, entre otras.

La detección de bordes se basa en determinar las curvas de nivel de intensidad luminosa similar a las curvas de nivel en topografía. Esto se ejecuta en dos pasos, en el primero se utiliza filtros para poder suavizar la imagen con la finalidad de eliminar gradientes de intensidad luminosa que resulten poco útiles, mientras que en el segundo paso se lleva a cabo la detección de bordes. Los principales desafíos de la detección de bordes incluyen la detección de bordes débiles, un bajo número de detección de bordes falsos y un posicionamiento espacial adecuado de los bordes detectados.

Las redes resistivas son predecesoras de las redes memristivas y han sido usadas para varios tipos de procesamientos de imágenes [4] [5] [6]. Una red resistiva contiene elementos resistivos no lineales en un arreglo rectangular como se muestra en la *figura 2.1*. Los nodos de la red están conectados a una fuente de voltaje que representa los niveles de intensidad luminosa de un píxel en particular fijado en escalas de grises, los voltajes nodales son los valores de salida de los píxeles que están previamente suavizadas por la red. El tamaño de la red resistiva está dado por el tamaño de la imagen en píxeles, usualmente  $n \times m$  donde  $n$  y  $m$  son el largo y ancho de la imagen respectivamente.

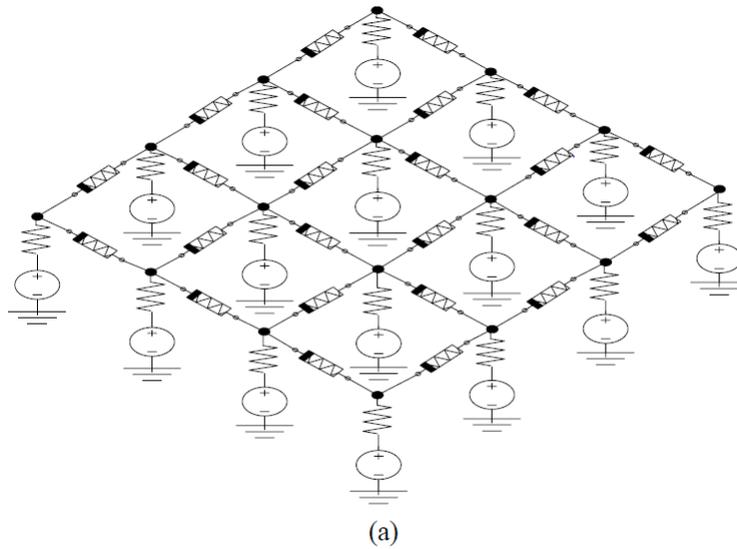


Figura 2.1: Red resistiva.

## 2.1. Red Memristiva

El paso de red resistiva a red memristiva para aplicaciones de procesamientos de imágenes es resultado del desarrollo del memristor como dispositivo real en los Laboratorios HP [7]. El resultado es un arreglo rectangular de memristores como se muestra en la *figura 2.2*.

Recientemente se tienen contribuciones en el área de procesamiento de imágenes con redes memristivas en particular, al suavizado de imágenes y a la detección de bordes en [5], [6] y [3].

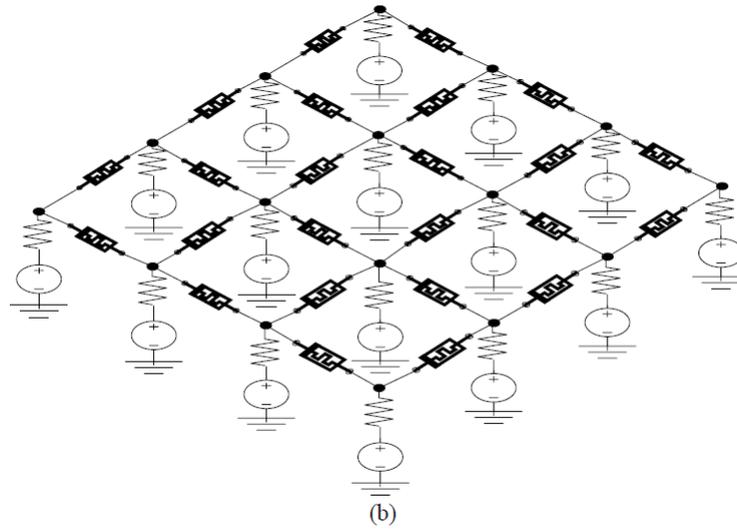
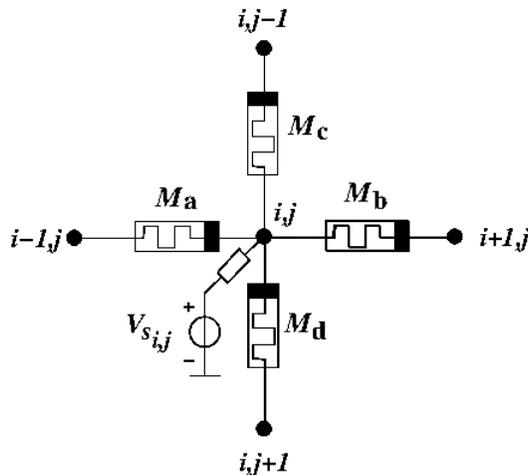


Figura 2.2: Red memristiva.

## 2.2. Ecuaciones nodales de la red memristiva

La red memristiva es analizada utilizando el método de análisis nodal que establece LCK (ley de corrientes de Kirchoff) en todos los nodos de la red [8–11]. El resultado del análisis es un conjunto de ecuaciones diferenciales dada la naturaleza no lineal y dinámica del memristor.

Con la finalidad de establecer las ecuaciones de la LCK, se realiza el análisis en el nodo  $i, j$  tal como se muestra en la *figura 2.3*. La suma de corrientes resulta en:

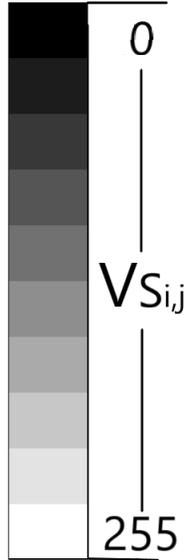
Figura 2.3: KCL para nodo  $i, j$ .

$$I_{in} + I_a + I_b + I_c + I_d = 0 \quad (2.2.1)$$

donde  $I_a, I_b, I_c, I_d$  son las corrientes de rama de los memristores conectados al nodo e  $I_{in}$  es la corriente de entrada del pixel. La ecuación 2.2.1 se puede expresar como:

$$\frac{V(s_{i,j}) - v_{i,j}}{R_{in}} + \frac{v_{i-1,j} - v_{i,j}}{M_a} + \frac{v_{i+1,j} - v_{i,j}}{M_b} + \frac{v_{i,j-1} - v_{i,j}}{M_c} + \frac{v_{i,j+1} - v_{i,j}}{M_d} = 0 \quad (2.2.2)$$

Donde  $M_a, M_b, M_c, M_d$  son las memristancias conectadas al nodo. Los subíndices de los voltajes corresponden a los nodos vecinos al nodo  $i, j$ . Además, la fuente de voltaje  $V(s_{i,j})$  es el valor de entrada del píxel, expresado en escalas de grises, que varía entre niveles discretos de 0 hasta 255. El negro corresponde a 0 y el blanco corresponde a 255. La escala cromatica de la *figura 2.4* muestra estos valores.



**Figura 2.4:** Escalas de grises para el voltaje de entrada  $V(s_{i,j})$ .

El voltaje de salida del nodo  $i, j$  puede ser expresado como:

$$v_{i,j} = \frac{V_{i,j}G_{in} + u_{i-1,j}W_a + u_{i+1,j}W_b + u_{i,j-1}W_c + u_{i,j+1}W_d}{G_{in} + W_a + W_b + W_c + W_d} \quad (2.2.3)$$

Donde  $G_{in}$  y  $W$  son los inversos de  $R_{in}$  y  $M$ , respectivamente. Esta expresión corresponde a cualquier nodo localizado en el interior de la red, existiendo casos particulares para los nodos que se encuentran en las orillas y las esquinas de la red.

Las expresiones para los voltajes en los nodos de las esquinas son:

$$\begin{aligned}
N_{1,1} : v_{1,1} &= \frac{V_{1,1}G_{in} + v_{2,1}W_{2,1} + v_{1,2}W_{1,2}}{G_{in} + W_{2,1} + W_{1,2}} \\
N_{1,n} : v_{1,n} &= \frac{V_{1,n}G_{in} + v_{2,n}W_{2,n} + v_{1,n-1}W_{1,n-1}}{G_{in} + W_{2,n} + W_{1,n-1}} \\
N_{m,1} : v_{m,1} &= \frac{V_{m,1}G_{in} + v_{m-1,1}W_{m-1,1} + v_{m,2}W_{m,2}}{G_{in} + W_{m-1,1} + W_{m,2}} \\
N_{m,n} : v_{m,n} &= \frac{V_{m,n}G_{in} + v_{m-1,n}W_{m-1,n} + v_{m,n-1}W_{m,n-1}}{G_{in} + W_{m-1,n} + W_{m,n-1}}
\end{aligned} \tag{2.2.4}$$

De la misma manera, las expresiones para los voltajes en los nodos de las orillas son:

$$\begin{aligned}
N_{1,j} : v_{1,j} &= \frac{V_{1,j}G_{in} + v_{2,j}W_{2,j} + v_{1,j-1}W_{1,j-1} + v_{1,j+1}W_{1,j+1}}{G_{in} + W_{2,j} + W_{1,j-1} + W_{1,j+1}} \\
N_{m,j} : v_{m,j} &= \frac{V_{m,j}G_{in} + v_{m-1,j}W_{m-1,j} + v_{m,j-1}W_{m,j-1} + v_{m,j+1}W_{m,j+1}}{G_{in} + W_{m-1,j} + W_{m,j-1} + W_{m,j+1}} \\
N_{i,1} : u_{i,1} &= \frac{V_{i,1}G_{in} + v_{i-1,1}W_{i-1,1} + v_{i+1,1}W_{i+1,1} + v_{i,2}W_{i,2}}{G_{in} + W_{i-1,1} + W_{i+1,1} + W_{i,2}} \\
N_{i,n} : v_{i,n} &= \frac{V_{i,n}G_{in} + v_{i-1,n}W_{i-1,n} + v_{i+1,n}W_{i+1,n} + v_{i,n-1}W_{i,n-1}}{G_{in} + W_{i-1,n} + W_{i+1,n} + W_{i,n-1}}
\end{aligned} \tag{2.2.5}$$

Estas ecuaciones son válidas en términos generales pero resulta imprescindible tener una expresión adecuada para las memristancias involucradas, es decir un modelo matemático del memristor.

## 2.3. Modelo del memristor

El modelo utilizado para los elementos memristivos de la red se reporta en [3], y se genera a partir del método de perturbación homotópica. El modelo resulta en una expresión controlada por carga de la memristancia en forma completamente analítica. Las expresiones resultantes para los tres primeros órdenes de la formulación homotópica son:

## Modelo orden-1

$$M_{O1} = [\Theta] \left[ \begin{array}{l} R_d(X_o - 1)[(X_0 - 2)e^{\Lambda 4\kappa q} - (X_o - 1)e^{\Lambda 8\kappa q}] \\ + R_{ON} \end{array} \right] + [R_d X_0 [X_0 e^{\Lambda 8\kappa q} - (X_0 + 1)e^{\Lambda 4\kappa q}] + R_{off}] [-\Theta + 1] \quad (2.3.1)$$

## Modelo orden-2

$$M_{O2} = M_{O1} + Rd([\Theta][(X_o - 1)^3[-e^{\Lambda 4\kappa q} - 2e^{\Lambda 8\kappa q} - e^{\Lambda 12\kappa q}]] + [X_0^3[-e^{\Lambda 12\kappa q} - 2e^{\Lambda 8\kappa q} - e^{\Lambda 4\kappa q}]][-\Theta + 1]) \quad (2.3.2)$$

## Modelo orden-3

$$M_{O3} = M_{O2} + Rd([\Theta][(X_o - 1)^4[-e^{\Lambda 4\kappa q} - 3e^{\Lambda 8\kappa q} - 3e^{\Lambda 12\kappa q} - e^{\Lambda 16\kappa q}]] + [X_0^4[-e^{\Lambda 16\kappa q} - 3e^{\Lambda 12\kappa q} - 3e^{\Lambda 8\kappa q} - e^{\Lambda 4\kappa q}]][-\Theta + 1]) \quad (2.3.3)$$

Donde la constante  $\kappa$  está dada como:  $\kappa = \frac{\mu R_{on}}{\Delta^2}$ ,  $\mu$  es la movilidad de los portadores,  $\Delta$  es la longitud del memristor,  $R_{off}$  es la resistencia en el estado de apagado y  $R_{on}$  es la resistencia en el estado de encendido. Adicionalmente, la constante resistiva  $R_d$  esta definida como:

$$R_d = R_{off} - R_{on} \quad (2.3.4)$$

Así mismo, los operadores  $\Theta, \Lambda$  son seleccionados como se indica en la [Tabla 2.3](#). Donde  $\eta$  toma los valores  $\pm 1$  dependiendo de la dirección del desplazamiento dentro del memristor.

El modelo desarrollado tiene la propiedad de que las expresiones de orden mayor contienen expresiones de ordenes menor. Es decir,  $M_{O2}$  contiene a  $M_{O1}$ , y  $M_{O3}$  contiene a su vez a  $M_{O2}$ . de manera representativa, esto resulta similar a las muñecas Matryoshkas como lo muestra la *figura 2.5*.



Figura 2.5: muñeca Matryoshkas.

	$\Theta = 0$	$\Theta = 1$
$\Lambda = 1$	$\eta^+$ $q \leq 0$	$\eta^-$ $q \leq 0$
$\Lambda = -1$	$\eta^-$ $q > 0$	$\eta^+$ $q > 0$

Tabla 2.1: Selección de operador.

Los elementos memristivos de la red mostrada en la *figura 2.2* son en realidad fusibles memristivos compuestos de 2 memristores conectados en anti-serie (espalda con espalda), tal como se muestra en la *figura 2.6a*. Los memristores están definidos para valores opuestos de  $\eta$ . Cada uno de ellos posee las curvas características memristancia-carga ( $M-q$ ) que se muestran en las *figura 2.6b* y *figura 2.6c*.

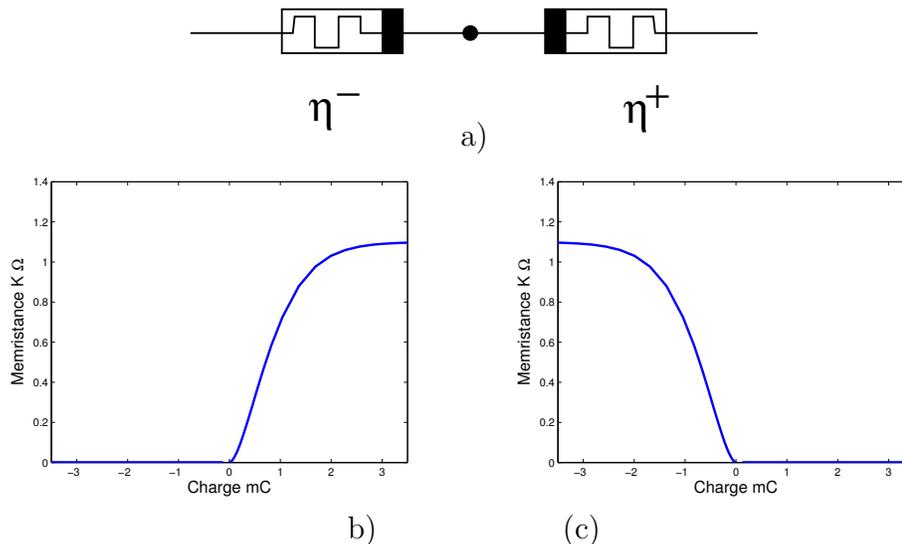


Figura 2.6:  $M$ - $q$  curvas para el fusible memristivo.

La conexión anti-serie permite combinar ambas curvas características en una curva  $M$ - $q$  asociada al fusible memristivo. Esquemáticamente esto se muestra en la *figura 2.7a*, donde la característica resultante exhibe dos niveles limite de memristancia denotados como  $M_{on}$  y  $M_{off}$ , las cuales son denominadas como memristancia de encendido y apagado del fusible, respectivamente. Esto permite definir un nivel (umbral) de memristancia  $M_{th}$  entre dos valores simétricos  $\pm Q_t$ . La *figura 2.7b* muestra una curva característica  $M$ - $q$  del fusible para los valores de parámetros mostrados en la Tabla 2.2.

Parm.	$\mu_v \frac{m^2}{Vs}$	$R_{on} \Omega$	$\Delta nm$	$R_{init} \Omega$	$X_0$	$R_{off} \Omega$	$M_{on} \Omega$	$M_{off} \Omega$	$R_{in} \Omega$
Value	$1 \times 10^{-14}$	1	10	1.1	0.999	1100	2.2	1101	50

Tabla 2.2: Valores nominales de parámetros.

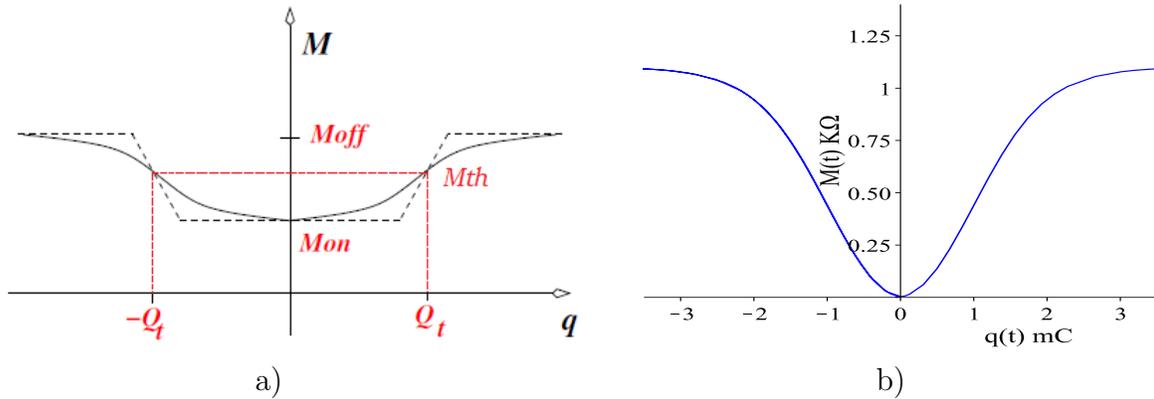


Figura 2.7: a) Característica ideal para la carga de la memristancia, b) Característica del modelo desarrollado de la carga de la memristancia

La característica memristancia-carga de la *figura 2.7* resulta especialmente útil para realizar tareas simultáneas de suavizado de imágenes y detección de bordes [3]. El suavizado de la imagen es consecuencia de la definición  $M_{th}$  durante la solución de las ecuaciones nodales en cada píxel. El procedimiento de la detección de bordes consiste en detectar aquellos píxeles vecinos que han alcanzado o sobrepasado el mismo porcentaje de memristancia después de un lapso de tiempo.

## 2.4. Red memristiva implementada en paralelo

Las expresiones que resuelven los voltajes de salidas de los píxeles han sido obtenidos a través de la implementación de la red memristiva desarrollada en [3], donde el algoritmo de la red es programado utilizando un procesamiento serial de las  $n \times m$  ecuaciones nodales 2.2.3. Por lo tanto, el tiempo de cómputo para determinar las soluciones de las ecuaciones nodales de la red depende mucho del tamaño de la imagen de entrada. Llegado a este punto, los tiempos de procesos para los análisis nodales se incrementan para imágenes grandes, de modo que se realiza la implementación de la red memristiva utilizando una GPU para poder tener las soluciones nodales en un menor tiempo. Por lo tanto, al tratar de reducir los tiempos de cómputo de la red memristiva es considerado como procesos de supercomputación.

La supercomputación es el motor de muchas de las tecnologías que vemos en los procesadores modernos. Gracias a la necesidad de procesadores cada vez más rápidos para procesar conjuntos de datos cada vez más grandes, la industria produce computadoras cada vez más rápidas, por lo tanto, debido a los grandes éxitos de los

video juegos en los ordenares, la GPU ha sido mejorada en gran medida, la utilización de este dispositivo se emplea para acelerar ciertos procesamientos y operaciones. Es a través de las necesidades de los usuarios, la GPU (Unidad de procesador de gráficos) y CUDA (Arquitectura unificada de dispositivos de cómputo) están evolucionando sus tecnologías. Tanto las supercomputadoras como la informática de escritorio se están moviendo hacia una informática heterogénea enrutada [12], es decir, están tratando de lograr un mejor rendimiento con una combinación de tecnología de CPU (Unidad de procesador central) y GPU.

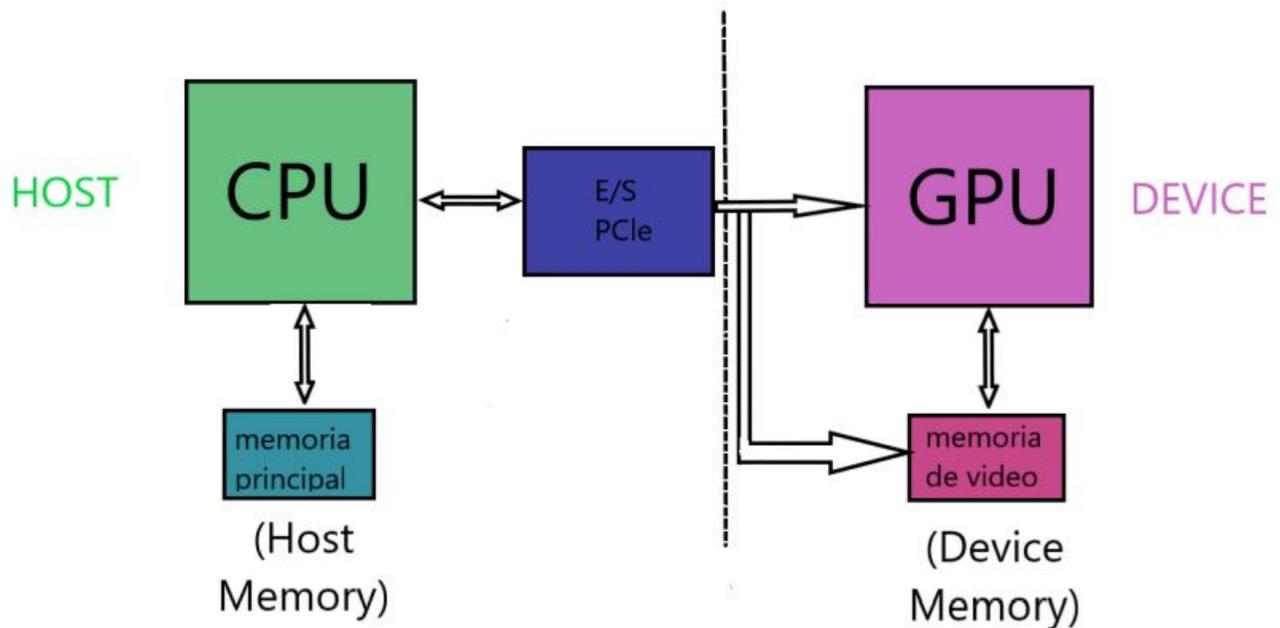


Figura 2.8: Estructura de una computadora convencional.

Como se muestra en la *figura 2.8*, la estructura de un ordenador está dividida o formada por dos partes principales, a una se le denomina host y la otra device, estas se pueden relacionar al termino maestro y esclavo, por la tanto, en el host se puede encontrar el procesador de la computadora o CPU (como mayormente es conocido), y en donde también se encuentra junto con la memoria principal, así como los dispositivos de entrada y salidas que establecen la comunicación entre el host y el device. En el device se encuentra la tarjeta gráfica o GPU y está junto con una memoria de vídeo asociada [12] [13].

La ejecución de los algoritmos se ha ejecutado sobre la host, el cual utiliza la CPU y el cual tiene un modelo de ejecución secuencial. Una de las grandes problemáticas de este sistema secuencial es cuando se pretende utilizar una gran cantidad procesos, por lo tanto, el sistema se vuelve más lento y aumenta el tiempo de la ejecución del algoritmo, y no se obtiene los resultados esperados en un tiempo que se pretendía que fuera muy pequeño.

Entonces para ser más preciso al momento de ejecutar códigos o algoritmos en paralelo se debe de tener la CPU y la GPU, en donde la CPU se realiza los códigos o algoritmos los cuales no es necesarios que sean paralelizados debido a que son algoritmos que obtienen soluciones de manera muy rápida, y los códigos o algoritmos que son los que se necesitan ser paralelizados se envían a la GPU [14]. Por lo tanto, con la previa selección sobre las partes del código que deben ser ejecutados en la GPU o CPU, esta selección permite velocidades de cálculo de cómputo muy rápidas o superiores.

Actualmente las computadoras tienen una estructura para mejorar su rendimiento en procesos, las cuales son requeridas con características especiales (computadora con CPU y GPU). En la *figura 2.8* se puede observar la estructura de una computadora convencional, esta estructura comprende las funciones de maestro y esclavo. Además la GPU en su implementación para paralelización tiene jerarquías que son importantes. En particular las jerarquías de la GPU depende una de las otras, así mismo hace mucho más fácil la programación.

### 2.4.1. Concepto de jerarquía

Dentro de la GPU podemos encontrar múltiples núcleos de procesamiento capaces de ejecutar programas de manera paralela. Dichos elementos se organizan siguiendo una jerarquización que facilita la programación y el correcto reparto de recursos. La GPU está dividido en mallas como elemento que contiene a todos los demás, cada malla dispone de distintos bloques y cada bloque está formado por un número determinado de hilos, para poder ejecutar una malla en la GPU se debe de ejecutar una función denominada kernel, esta función es lanzada desde la CPU hacia la GPU.

## Malla

La *figura 2.9* representa el esquema general del modelo de programación CUDA. Donde se puede observar los dos elementos principales ya mencionados con anterioridad, GPU y CPU, estos dos elementos están representados en los rectángulos azules de la *figura 2.9*. Como se puede observar la CPU o también conocido como “host” es el que se encarga de organizar y lanzar la ejecución de los kernels, y los cuales se ejecutan en dos mallas como se muestra en la *figura 2.9*, pero también se pueden utilizar múltiples mallas, todo depende de las demandas que el programa a ejecutar necesita. Observando detalladamente la malla está constituida por una matriz de bloques, la cual está formada por tres columnas y dos filas. Teniendo como referencia al bloque (1,1), el cual está constituido por una matriz de hilos de cinco columnas y tres filas; y donde cada hilo será el encargado de ejecutar el kernel [15] [16].

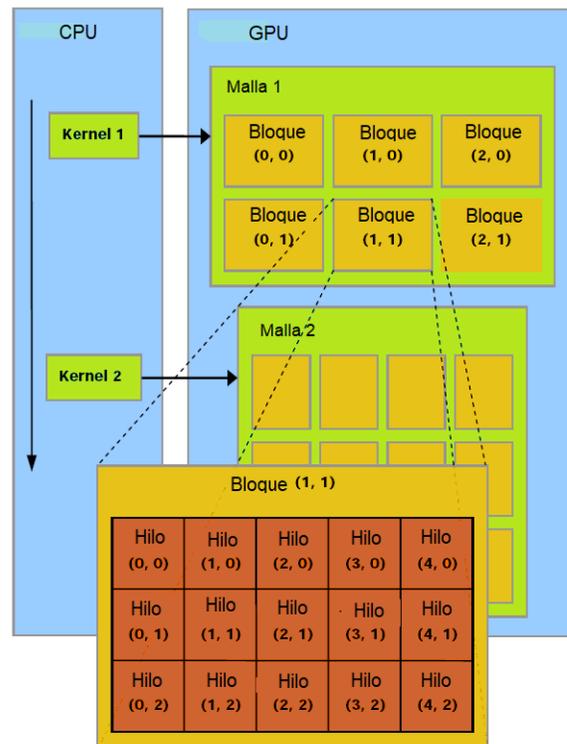
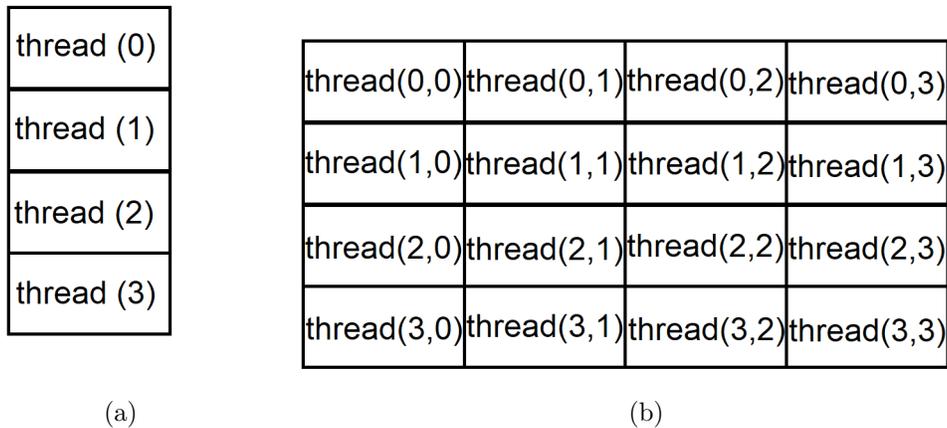


Figura 2.9: Modelo de ejecución de un kernel.

## Hilo

El código que se desea implementar de manera intensiva en la GPU se le llama kernel. Para poder lanzar este código se invoca desde la CPU, la CPU decide u ordena el trabajo que se realizara en la GPU, donde se crean múltiples hilos los cuales están encargados de ejecutar el kernel lanzado por la CPU, donde cada hilo se ejecuta de manera individual e independiente. También se establece que los datos que procesa cada hilo pueden ser distintos.

La característica que tienen los hilos en la GPU es constar de información que le permite identificarse de manera única, cada uno a la vez puede tomar sus propias decisiones de control y también pueden tomar la dirección de su memoria local. Para poder tener un control eficaz y preciso de los hilos, estos se pueden agrupar en forma de vector o en forma de una matriz como se muestra en la *figura 2.10*, también estos hilos se pueden organizar de manera tridimensional pero esto es decisión del programador según las dimensiones que se desea o para que tipo de aplicación se necesitan [16] [15] [14].



Organización de los hilos.

Figura 2.10: (a) Hilos como vector. (b) Hilos con estructura matricial.

## Bloque

Un bloque están compuestos por una serie de hilos, en la sección 2.4.1 se mencionó como pueden estar distribuidos o formados los hilos , por lo tanto los bloques son un conjunto de hilos; los bloques deben de cumplir con ciertas características:

- Los bloques pueden estar formados con un conjunto de hilos, los cuales tiene una forma de vector, matriz bidimensional o matriz tridimensional.
- Cuando el kernel es lanzado, ya se ha especificado la estructura de los hilos para el bloque, estos no pueden ser cambiados por algún otro tamaño o tipo de estructura durante el proceso.
- Todos los hilos dentro de un bloque pueden cooperar su información entre sí, pero estos no pueden relacionarse con otros hilos de otros bloques.
- Los bloques a su vez se agrupan en una malla.
- El identificador de hilo es único dentro de cada bloque, es decir cada hilo tiene su propio nombre.

## 2.5. Solución de la red memristiva en paralelo

En la sección 2.2 se definió y resolvieron las ecuaciones de la red memristiva, entonces la solución de las ecuaciones de la red memristiva con un algoritmo paralelizado se ilustra en la *figura 2.11*. Cada uno de los píxeles es enviado a un hilo, donde cada hilo pertenece a un bloque y la malla está compuesta por los bloques de hilos. Cada uno de estos hilos toma sus propias decisiones para poder resolver las ecuaciones nodales. Como ya se mencionó anteriormente, los hilos tienen sus propios identificadores entre ellos y de acuerdo a estos identificadores, los píxeles de la *figura 2.11* se pueden ir enviando a un hilo en particular para su proceso de cómputo.

La clave para efectuar un buen funcionamiento en los procesos paralelos implementados en la GPU es la correcta programación sobre los hilos. De tal modo que, los hilos son independientes al momento de tomar decisiones en los procesos, característica que permite calcular los voltajes nodales de salida simultáneamente para cada uno de los nodos abarcados por la longitud del hilo. A su vez la carga controladora de las expresiones de memristancia ( $M(q)$ ) de los fusibles se obtiene por integración numérica de la corriente en un proceso paralelo para todos los fusibles contenidos en el hilo. El lanzamiento del kernel debe de tener todas las jerarquías para que este puede tener una mayor eficacia en la implementación de la red memristiva, además depende del programador cuantas mallas, bloques e hilos se debe de utilizar; por lo tanto, los hilos constituyen la base de la ejecución de aplicaciones en paralelo.

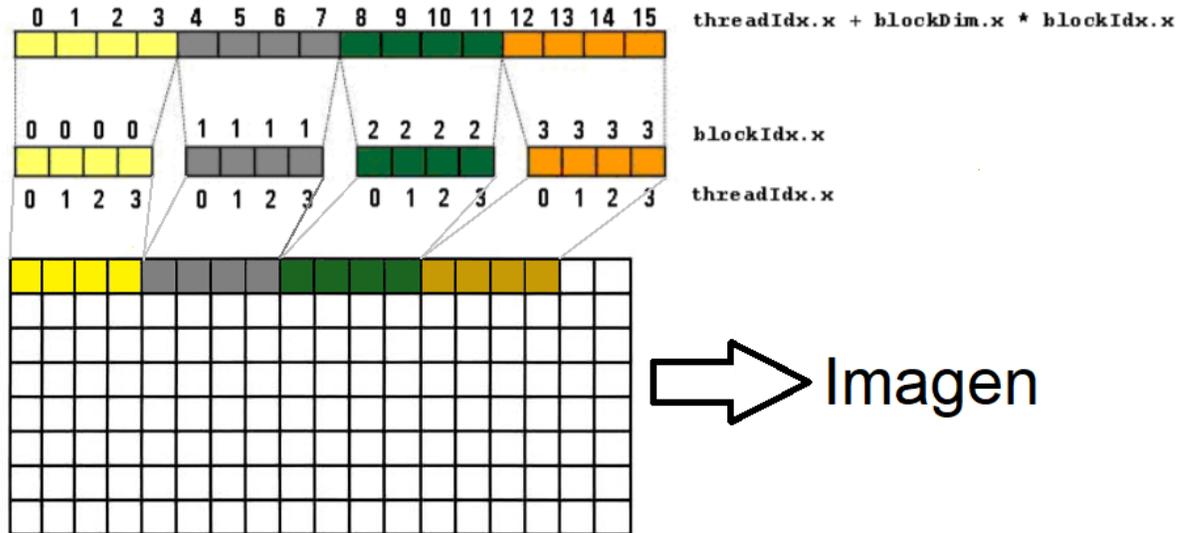


Figura 2.11: Distribución de los hilos sobre los píxeles de la imagen.

Para poder utilizar la red memristiva paralelizada en la GPU se implementó una malla formada por un único bloque y 1024 hilos paralelos. Esto le dice al sistema que queremos una malla unidimensional (el cual está formado como un vector de hilos y es considerado como vector unidimensional), en la *figura 2.12*, se muestra como cada hilo va ocupando a un píxel, de modo que es el esquema de como los hilos van ocupando los píxeles de manera ordenada, ya que por defecto se considera este esquema como la dimensión de trabajo. De este modo, las ecuaciones diferenciales se resuelven simultáneamente, los cuales cada uno de estos hilos representa 1024 nodos; por lo tanto, se puede resolver 1024 voltajes nodales simultáneamente. Los voltajes nodales son ecuaciones independientes para cada píxel. Por lo tanto, las tareas simultáneas de los cálculos de cada nodo se pueden obtener sin afectar a los nodos vecinos, y la obtención de los bordes dependería del cambio de intensidad luminosa entre los píxeles vecinos.

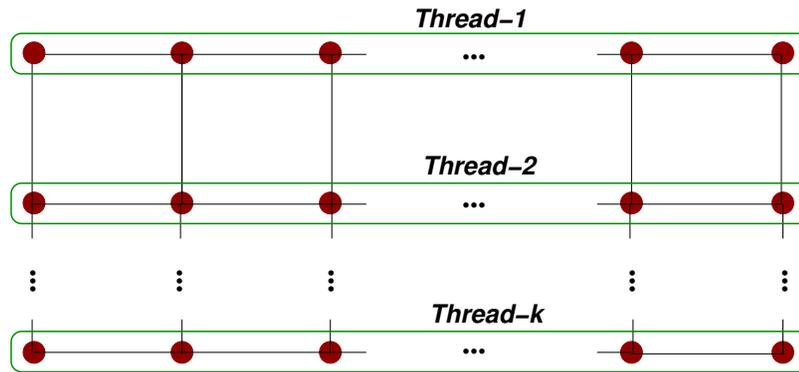


Figura 2.12: Esquema de nodos paralelizados en la red memristiva.

## 2.6. Resultados

Se ha realizado tratamiento de imágenes utilizando la red memristiva, además el algoritmo de la red es realizada en modo serie y paralela para evaluar el rendimiento de la red memristiva, la eficiencia del método de integración de la carga y la selección de un umbral ( $M_{th}$ ) para la detección de bordes con las características mencionadas; se puede decir que la red tiene un funcionamiento óptimo para procesamiento de imágenes. Son utilizadas para este proceso imágenes (obtenida de la base de datos BSD500 [17]); para el presente trabajo se utilizaron 9 imágenes en las cuales correspondían como mejores, promedios y malos resultados.

Con la implementación de la red memristiva en una GPU para poder realizar más rápido los cálculos ya mencionados, se obtiene la imagen de salida procesada por la red, dicho de otra manera obtenemos la imagen suavizada por la red. En la *figura 2.13*, se muestra la imagen suavizada a diferentes tiempos de suavizado con una red memristiva paralelizada; se puede observar entre más tiempo de suavizado tenga la imagen esta reduce en gran medida la intensidad de muchos píxeles, a su vez el suavizado de la red solo deja píxeles que presentan una gran intensidad luminosa.

a)  $t = 1 \text{ ms}$ b)  $t = 5 \text{ ms}$ c)  $t = 10 \text{ ms}$ d)  $t = 20 \text{ ms}$ **Figura 2.13: Imagen de salida para diferentes tiempos de suavizado.**

Los resultados del suavizado en paralelo en la imagen de prueba de la *figura 2.13* son idénticos a los presentados en [3].

### 2.6.1. Detección de bordes

A partir del proceso de suavizado se lleva a cabo la detección de bordes. En la *figura 2.14a* se muestra la imagen de entrada en escalas de gris. Al aplicar un suavizado con un umbral de 2% de  $M_{off}$ , se obtiene la imagen de detección de bordes mostrada en la *figura 2.14b*.

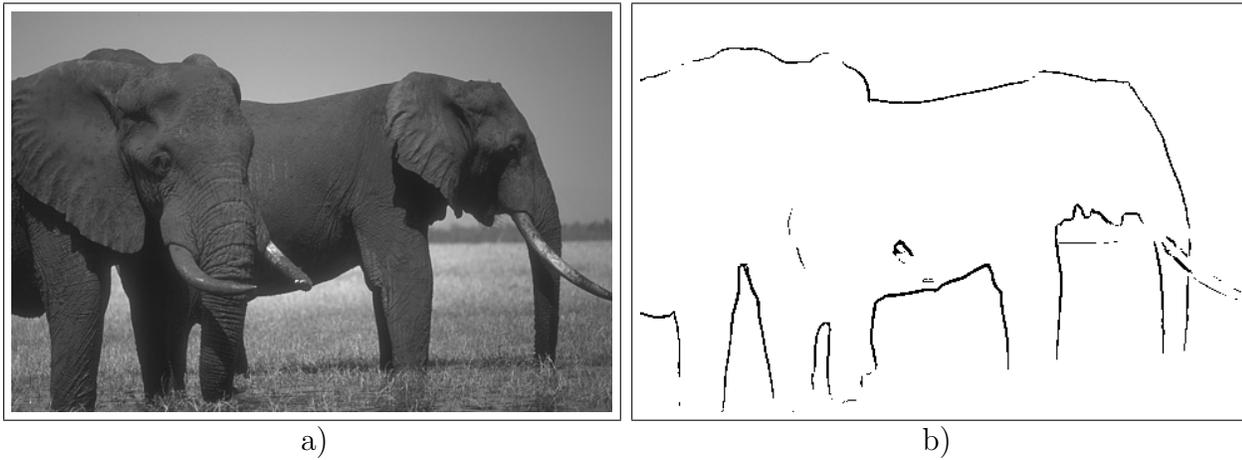


Figura 2.14: (a) Escala de gris. (b) Bordes.

Para poder comparar los resultados de los bordes, con los bordes encontrados por humanos, se realiza la curva de precisión-recuperación [17].

Donde el parámetro de recuperación es:

$$R = \frac{T_P}{T_B} \quad (2.6.1)$$

$T_P$  es el número de píxeles que pertenecen al borde evaluado, así como al borde de referencia (los cuales se les conocen como verdaderos positivos) y  $T_B$  es el número total de píxeles que pertenecen al borde en la imagen de referencia.

La precisión ( $P$ ) se define como:

$$P = \frac{T_P}{T_B + F_P} \quad (2.6.2)$$

$F_P$  es el número de píxeles que pertenecen al borde evaluado pero no al borde de referencia (conocidos como falsos positivos). La precisión representa en realidad la calidad que tiene el método de detección de bordes. Además la recuperación es la probabilidad del método para que se obtenga un borde verdadero.

Por último se tiene el parámetro  $F$ :

$$F = \frac{PR}{\beta P + (1 - \beta)R} \quad (2.6.3)$$

El parámetro  $F$ , es el promedio que se obtiene en los píxeles de bordes detectados con respecto a la imagen de referencia.

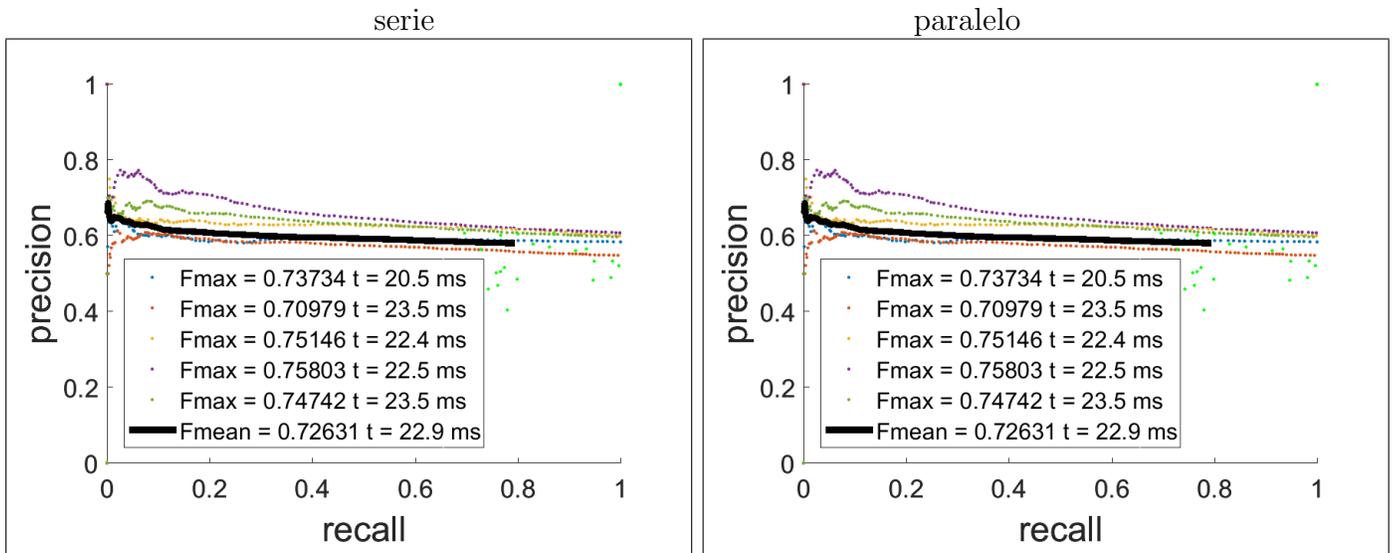


Figura 2.15: Graficas precision-recall.

Las gráficas de precisión-recuperación se muestran en la *figura 2.15*, donde el tiempo de suavizado es de  $22ms$  y con esto se puede observar de igual manera que ambas gráficas son prácticamente iguales, esto debido a que los parámetros de la red memristiva en el tratamiento de imágenes son iguales para ambos procesos (serie y paralelo), por lo cual las dos gráficas son similares; y para todo esto la ventaja que se tiene entre la programación en serie y la programación paralela son los tiempos de cómputo.

En la *figura 2.16* se muestran los tiempos de cómputo, donde se puede apreciar que el tiempo de cómputo en serie de la red memristiva es mucho mayor que el tiempo de cómputo para la red memristiva paralela.

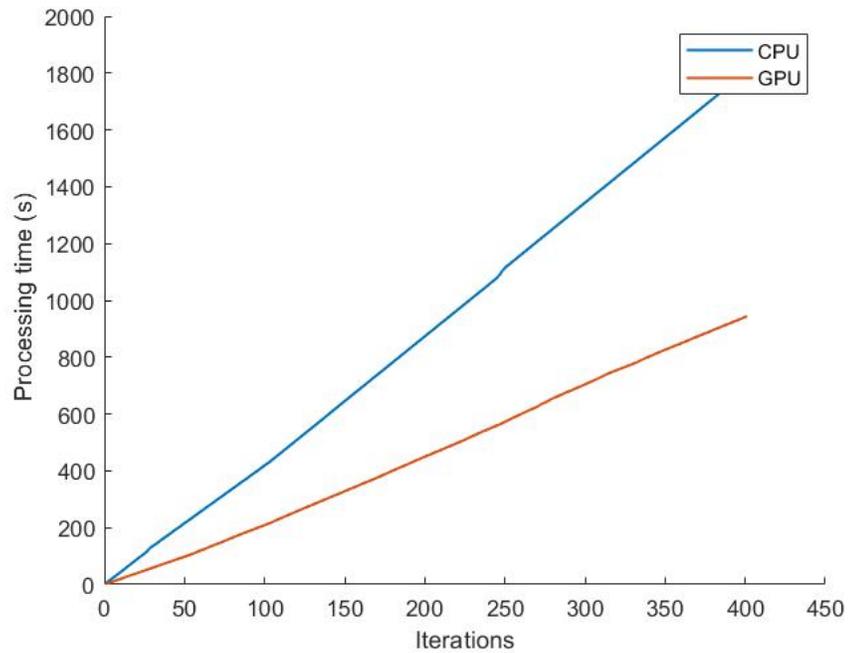
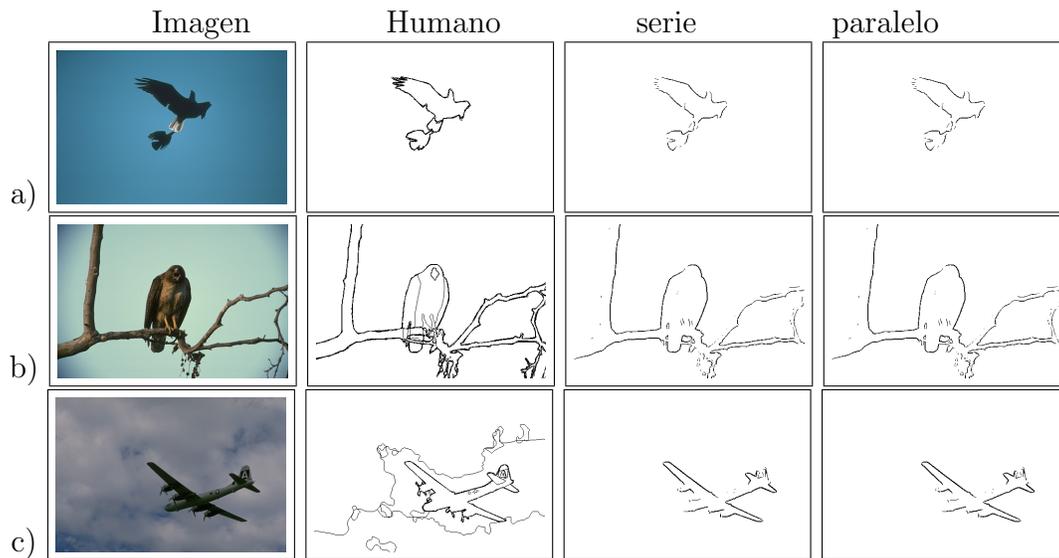


Figura 2.16: Grafica de tiempo de CPU vs GPU.

### Casos de estudio modales

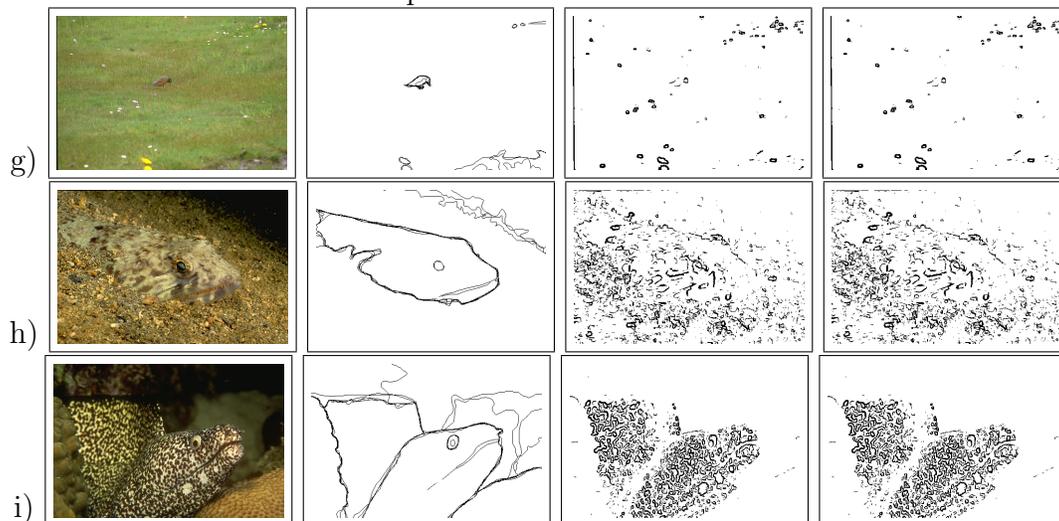
Con la finalidad de realizar un análisis comparativo del comportamiento de la red paralela con respecto a la red serial, se han seleccionado 9 imágenes de prueba agrupadas en los extremos (mejores y peores) y el promedio con respecto a la detección de bordes. La *figura 2.17* muestra los resultados de la detección de bordes para los 3 mejores, los 3 promedios y los 3 peores. Se observa que tanto la red en paralelo como la red en serie exhiben salidas casi idénticas.



Mejores resultados de la red memristiva.



Resultados promedios de la red memristiva.



Resultados malos de la red memristiva.

DETECCIÓN  
 Figura 2.17: Extracción de bordes.

### 2.6.2. Detección de bordes en función del umbral

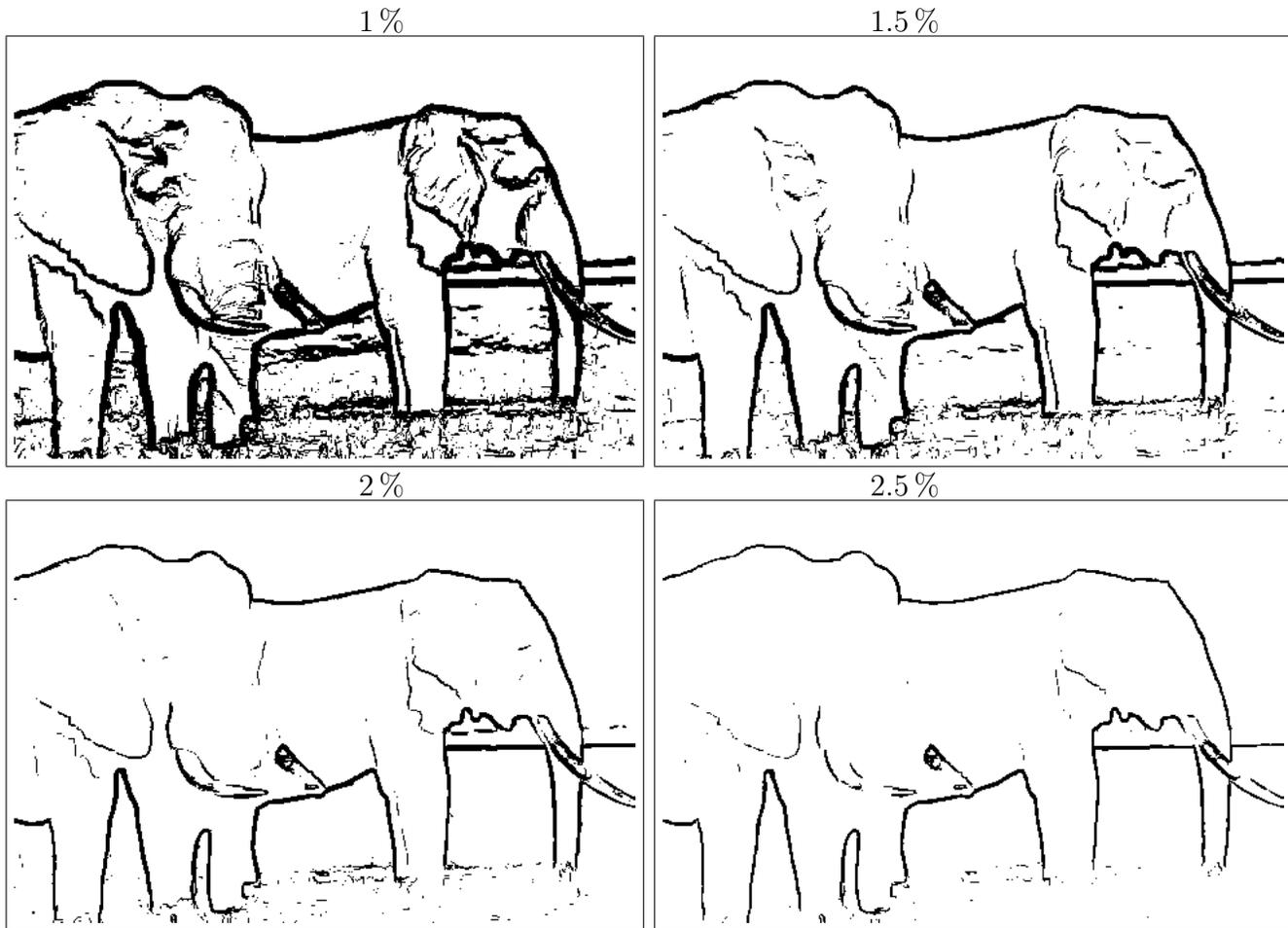


Figura 2.18: Diferentes porcentaje en umbral para la detección de bordes.

Con el objetivo de demostrar que el umbral de memristancia que determina el suavizado incide en el resultado final de la detección de bordes, hemos analizado nuevamente la imagen de prueba de la *figura 2.14a*.

La *figura 2.18* muestra la detección de bordes para umbrales de 1 %, 1.5 %, 2 % y 2.5 % de  $M_{off}$ .

## 2.7. Conclusiones

Este capítulo ha tratado sobre la utilización de la red memristiva para detectar bordes. Nos hemos enfocado a una implementación paralelizada de la red memristiva, a través de un esquema CPU-GPU. La red memristiva es capaz de realizar tanto el suavizado de la imagen como la detección final de bordes gracias a que el modelo del memristor utilizado es un modelo matemático de memristancia controlada por carga que permite establecer un valor de memristancia como el umbral que define al suavizado. Como casos de estudio, se han usado imágenes de la base de datos BSD500 [17] realizando comparaciones con la red memristiva serial. Se han llevado a cabo análisis comparativos cubriendo tanto los aspectos cuantitativos (curvas precisión-recuperación) como los aspectos cualitativos para el tratamiento de varias imágenes de la base de datos. Si bien los resultados comparativos de la implementación en paralelo y serial resultaron idénticos, al final la red en paralelo ha mostrado su ventaja en términos de un notorio ahorro en el tiempo de procesamiento total. La implementación y el análisis comparativo fueron realizados en MATLAB.



# Estudio morfológico de semillas

---

En este capítulo se lleva a cabo el estudio de las características que definen la morfología de la semilla, utilizando la red memristiva paralelizada para extraer los bordes en imágenes de las mismas. Primeramente, se presenta una breve introducción a conceptos botánicos que nos permiten establecer la importancia que tiene el estudio morfológicos de las semillas. Adicionalmente, se presenta una comparación de dos aproximaciones dadas por lugares geométricos para modelar los bordes de las semillas. El análisis de la morfología se basa en la medición de diversos parámetros geométricos sobre las imágenes con extracción de bordes de semillas. Las imágenes tratadas provienen de la base de datos [18].

## 3.1. Conceptos de Botánica

La semilla es el cuerpo reproductivo característico de muchas especies de plantas. Una semilla es esencialmente una planta en miniatura (embrión) que resulta ser extremadamente eficiente para mantenerse viva exigiendo poca demanda de nutrientes del ambiente que la rodea. Con dicha eficiencia es capaz de desarrollar una amplia cantidad de funciones orientadas a permitir su supervivencia, entre otras: multiplicación, perennizarse, inactividad y dispersión. Todas ellas con el único propósito de garantizar la germinación, la cual se puede dar en corto, mediano o largo plazo, dependiendo de la especie y de las condiciones que la rodean. Este gran nivel de resiliencia o adaptación ha sido la clave para el éxito evolutivo de un gran número de especies de plantas.

La dispersión es un mecanismo inherente en las semillas para lograr desplazarse y poblar áreas geográficas lejanas. Existen mecanismos autóctonos a la planta como son

la gravedad (dejar caer las semillas), o la emisión (expulsar las semillas). Existen también mecanismos externos tales como el viento (anemocoria), por agua (hidrocoria), por animales (zoocoria) o por actividades de humanos.

Las semillas pueden ser de gimnospermas y angiospermas. Las semillas de gimnospermas son llamadas semillas desnudas dado que no están cubiertas y se encuentran sujetas a un mecanismo externo para su polinización y posterior fertilización. Esta característica redundante en largos tiempos de germinación como es el caso de las coníferas (pinos). Las semillas de angiospermas por el contrario se encuentran confinadas a un espacio (usualmente el fruto) y provienen de plantas con flores donde la polinización y fertilización ya ha tenido lugar.

Desde el punto de vista de su estructura interna, las semillas pueden ser monocotiledóneas o dicotiledóneas, lo que en general define mecanismos diferentes de crecimiento de la nueva planta. La *figura 3.1* muestra un ejemplo de monocotiledónea y dicotiledónea. Puede observarse que la semilla dicotiledónea muestra una cicatriz llamada hilo que representa el punto donde la semilla estuvo unida a la planta y presenta claramente dos mitades que son los dos cotiledones.



**Figura 3.1:** Ejemplos de Monocotiledonea (maíz) y Dicotiledonea (frijol).

Las semillas tienen un fuerte impacto en el ámbito económico dado que muchas de ellas tienen fines alimenticios o energéticos por lo que constituyen bienes de explotación y consumo. Existe un amplio comercio mundial de semillas para alimentación (granos y cereales) lo que requiere un enorme y extenso trabajo de transportación y embarque. En resumen la semilla ha pasado de ser un ente estático atado a su planta y entorno más cercano, para convertirse en un ente con gran movilidad (dispersión).

En este contexto de alta dispersión, se encuadra el actual capítulo pues el análisis de la morfología de semillas permite distinguir las semillas *buenas* de las semillas *malas* (malezas). El material presentado en este capítulo representa el primer paso para un proyecto futuro sobre detección de malezas.

## 3.2. Modelos geométricos de semilla

Con el objetivo de simplificar el análisis morfológico de la gran cantidad de semillas, se debe recurrir a modelos geométricos que representen espacialmente los contornos provenientes de imágenes de semillas. Esta sección se enfoca a la comparación de dos modelos generados por lugares geométricos:

**Modelo ovoidal** Este modelo propone que la semilla se aproxime a un ovoide [19], el cual está dado por el sistema de ecuaciones:

$$\begin{aligned}x(\theta) &= a \cos \theta \\y(\theta) &= (b + c \cos \theta) \sin \theta\end{aligned}\tag{3.2.1}$$

donde  $\theta$  es la variable paramétrica,  $a$  es el semieje en la dirección  $x$ ,  $b$  es el semieje en la dirección  $y$ , y  $c$  define el radio en la porción más aguda del ovoide que introduce la deformación con respecto a una elipse. Resulta claro que  $c = 0$  reduce la ecuación 3.2.1 a la ecuación paramétrica de una elipse. La ecuación 3.2.1 permite definir un ovoide centrado en el origen y en posición horizontal. La *figura 3.2* muestra el ovoide y los parámetros involucrados en la formulación.

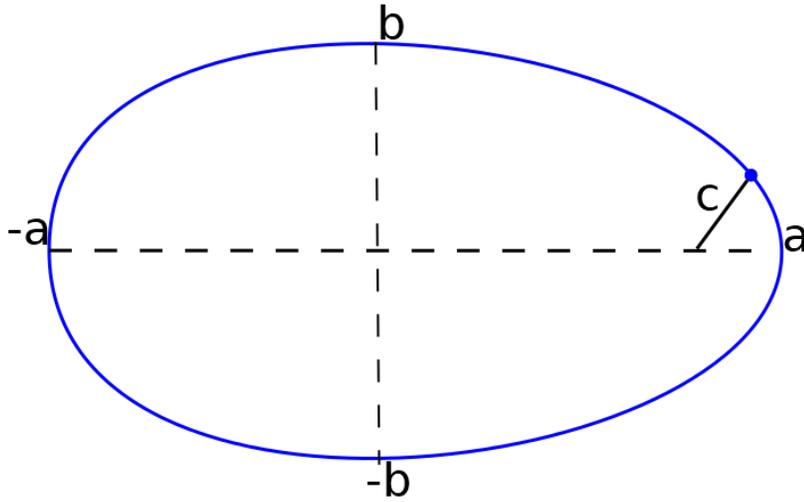


Figura 3.2: Modelo ovoidal

Una formulación alternativa que introduce mayor flexibilidad para la formulación de un ovoide está dada por:

$$\begin{bmatrix} x(\theta) \\ y(\theta) \end{bmatrix} = \begin{bmatrix} O_x \\ O_y \end{bmatrix} + \begin{bmatrix} \cos(\alpha) & -\sin(\alpha) \\ \sin(\alpha) & \cos(\alpha) \end{bmatrix} \begin{bmatrix} a \cos \theta \\ (b + c \cos \theta) \sin \theta \end{bmatrix} \quad (3.2.2)$$

donde el punto  $(O_x, O_y)$  representa el nuevo centro del ovoide y la matriz que pre-multiplica al sistema definido por la ecuación 3.2.1 representa una matriz de rotación. Es decir, la ecuación 3.2.2 puede ser aplicada para modelar un semilla en cualquier posición y orientación.

Dos características pueden ser utilizadas para aplicar el ovoide como modelo de semillas, ellas son el área y perímetro.

Por un lado, el área se calcula a través de la integral de la función paramétrica:

$$A_o = \int_0^{2\pi} y(\theta) dx \quad \text{con} \quad dx = \frac{dx}{d\theta} \quad (3.2.3)$$

la cual resulta en:

$$A_o = \pi ab \quad (3.2.4)$$

Por otro lado el perímetro se puede calcular utilizando la integral de longitud de arco:

$$P_o = \int_0^{2\pi} \sqrt{\begin{bmatrix} -a \cos \alpha \sin \theta + c \sin \alpha \sin^2(\theta) - \sin \alpha (b + c \cos \theta) \cos \theta \\ -a \sin \alpha \sin \theta - c \cos \alpha \sin^2(\theta) + \cos \alpha (b + c \cos \theta) \cos \theta \end{bmatrix}^2} d\theta \quad (3.2.5)$$

Si hacemos  $\alpha = 0$ , i.e. el ovoide no está rotado, entonces la expresión para el perímetro se reduce a:

$$P_o = \int_0^{2\pi} \sqrt{a^2 \sin^2 \theta + [-c \sin^2 \theta + (b + c \cos \theta) \cos \theta]^2} d\theta \quad (3.2.6)$$

**Modelo elipsoidal** La propuesta de este modelo es la aproximación de la semilla por una elipse. El lugar geométrico está definido por la conocida ecuación de la elipse [20]:

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} = 1 \quad (3.2.7)$$

donde  $a$  es el semieje en la dirección  $x$ ,  $b$  es el semieje en la dirección  $y$ . La forma paramétrica de la ecuación 3.2.7 está dada por:

$$\begin{aligned} x &= a \sin \theta \\ y &= b \cos \theta \end{aligned} \quad \theta, \in [0, 2\pi] \quad (3.2.8)$$

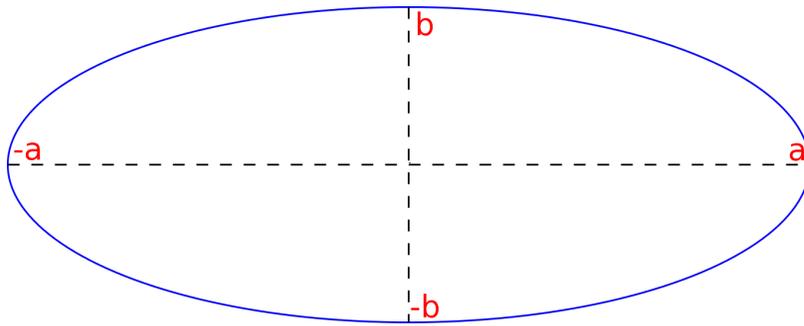


Figura 3.3: Modelo elipsoidal

La figura 3.3 muestra la elipse y sus parámetros. A partir de ellos, se obtiene la fórmula del área:

$$A_e = \pi ab \quad (3.2.9)$$

Para obtener el perímetro de la elipse, se recurre a la forma aproximada [21] dada por:

$$P_e \approx \pi \left[ 3(a+b) - \sqrt{(3a+b)(a+3b)} \right] \quad (3.2.10)$$

### 3.2.1. Redondez y Compacticidad

La redondez y la compacticidad son dos parámetros importantes que sirven para definir que tan cercana a un círculo es la forma de un objeto en 2D.

Por una lado, la redondez está definida como:

$$Rd = \frac{4\pi A}{P^2} \quad (3.2.11)$$

donde  $A$  es el área y  $P$  es el perímetro del objeto.

Por otro lado, la compacticidad se define como:

$$C = \frac{P^2}{A} \quad (3.2.12)$$

De las ecuaciones 3.2.11 y 3.2.12 es fácil comprobar que el producto compacticidad-redondez es igual a:

$$CRdProduct = 4\pi \quad (3.2.13)$$

Resulta claro que para un círculo de radio  $r$  se obtienen los siguientes valores de redondez y compacticidad:

$$Rd_{circ} = \frac{4\pi(\pi r^2)}{(2\pi r)^2} = \mathbf{1} \quad C_{circ} = \frac{(2\pi r)^2}{(\pi r^2)} = \mathbf{4\pi} \quad (3.2.14)$$

Este resultado indica que  $Rd = 1$  es el valor ideal de redondez y que corresponde a un círculo. Correspondientemente, la compacticidad ideal es  $4\pi$ . Adicionalmente, ambas medidas son invariantes a la traslación, rotación y escalamiento.

## Análisis

A continuación, se realiza un breve estudio de los resultados de redondez y compacticidad para una elipse con semieje horizontal  $a = 10$  y semieje vertical  $b$  que toma diversos valores dados por la secuencia  $b = [1, 2, 3, 5, 10, 20, 30, 50, 100]$ .

$b$	Área	Perímetro	Compacticidad	Redondez	(3.2.15)
100	$1000.0 \pi$	$129.2514010 \pi$	$16.70592466 \pi$	$0.2394360134$	
50	$500.0 \pi$	$66.8629150 \pi$	$8.941298804 \pi$	$0.4473623004$	
30	$300.0 \pi$	$42.54033308 \pi$	$6.032266463 \pi$	$0.6631006812$	
20	$200.0 \pi$	$30.83920217 \pi$	$4.755281952 \pi$	$0.8411698908$	
10 circle	$100.0 \pi$	$20.0 \pi$	$4.0 \pi$	$1.0$	
5	$50.0 \pi$	$15.41960108 \pi$	$4.755281950 \pi$	$0.8411698912$	
3	$30.0 \pi$	$13.96003195 \pi$	$6.496083067 \pi$	$0.6157556728$	
2	$20.0 \pi$	$13.37258300 \pi$	$8.941298805 \pi$	$0.4473623000$	
1	$10.0 \pi$	$12.92514010 \pi$	$16.70592466 \pi$	$0.2394360134$	

Se puede verificar que el producto redondez-compacticidad permanece igual a  $4\pi$  en todos los casos. Adicionalmente  $Rd = 1$  implica un círculo y a la vez es el valor máximo que puede tomar. Para la compacticidad – asociada al círculo – toma el valor de  $4\pi$  y corresponde a su valor mínimo.

### 3.2.2. Comparación entre ovoide y elipse

A continuación se realizará un estudio comparativo entre los modelos utilizando para ello las características de área, perímetro, redondez y compacticidad. Los parámetros a utilizar se muestran en la *Tabla 3.1*.

Modelo	$a$	$b$	$c$
Ovoide	600	250	60
Elipse	600	250	

**Tabla 3.1:** Parámetros usados para la descripción del ovoide y elipse.

Primeramente, es importante notar de las ecuaciones 3.2.4 y 3.2.9, que el valor de las áreas de ambos modelos son idénticas. La *figura 3.4* permite ver claramente esto pues el ovoide gana superficie respecto a la elipse a la derecha y pierde la misma cantidad de superficie en la izquierda. Los resultados comparativos se muestran en la *Tabla 3.2*.

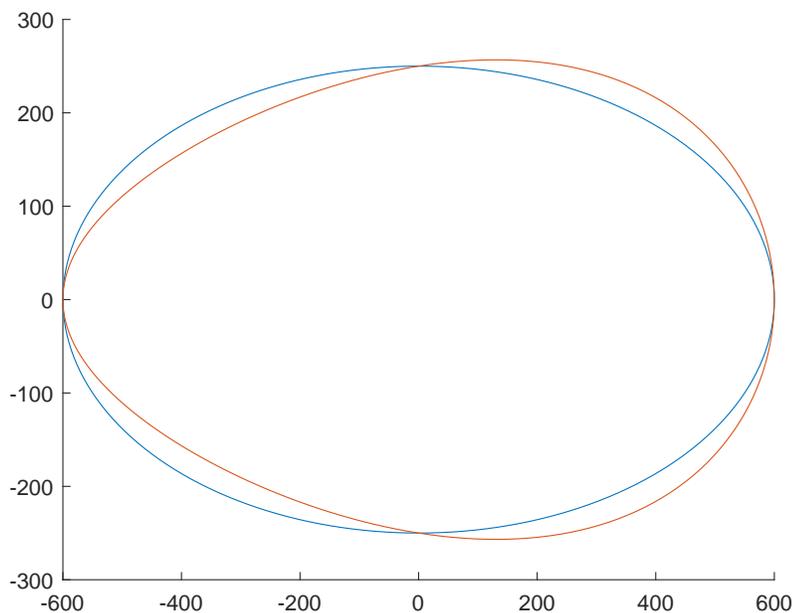


Figura 3.4: Modelo elipsoidal (azul) y modelo ovoide (rojo).

Característica	Elipse	Ovoide
Área	$1.5 \times 10^5 \pi$	$1.5 \times 10^5 \pi$
Perímetro	2784.769065	2791.904607
Compacticidad	16.45649113	16.54093362
Redondez	0.763611788	0.7597135025

Tabla 3.2: Características morfológicas de ambos modelos para los valores de la *tabla 3.1*.

Si se toma como referencia el modelo elíptico, entonces se puede definir una figura de error como:

$$\epsilon = \left\| \frac{\mathcal{C}_e - \mathcal{C}_o}{\mathcal{C}_e} \right\| \quad (3.2.16)$$

donde  $\mathcal{C}_e$  y  $\mathcal{C}_o$  representan una característica específica del modelo elíptico y ovoide respectivamente.

La *Tabla 3.3* resume el error en porcentaje de cada una de las características. Los errores reportados son menores a 0.5 %.

Característica	Error
Área	0
Perímetro	0.2562346045
Compacticidad	0.5131257285
Redondez	0.5105062364

Tabla 3.3: Error en porcentaje entre ambos modelos.

Como conclusión del análisis comparativo de los modelos, se elige el modelo elíptico dado que requiere solamente dos parámetros, los semiejes mayor y menor.

### 3.3. Extracción de parámetros morfológicos

A continuación se toma como punto de partida la elipse para modelar el cuerpo de la semilla y del hilo, tal como lo muestra la *figura 3.5*, procederemos a la extracción de los parámetros morfológicos de la semilla. En [22] y [23], se ha definido una serie de características para este fin, las cuales son invariantes a la rotación, escalamiento y traslación. En este trabajo se utilizan dichas características, además de la compacticidad y redondez.

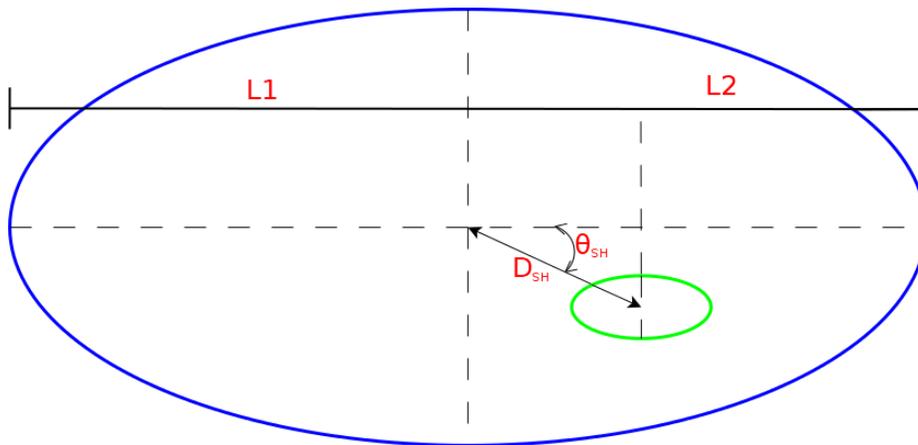


Figura 3.5: Elipse con hilo para modelar la semilla con hilo.

Los parámetros morfológicos son los siguientes:

$f_1$  Relación entre la longitud del eje mayor de la semilla ( $L_{XS}$ ) y la distancia entre el centro de la semilla y del hilo ( $D_{SH}$ ).

$$f_1 = \frac{L_{XS}}{D_{SH}} \quad (3.3.1)$$

$f_2$  Representa el ángulo de la línea que une los centros con el eje mayor de la semilla.

$$f_2 = \Theta_{SH} \quad (3.3.2)$$

$f_3$  Relación de áreas.

$$f_3 = \frac{A_S}{A_H} \quad (3.3.3)$$

donde  $A_S$  y  $A_H$  son las áreas de las semillas y del hilo respectivamente.

$f_4$  Relación de perímetros.

$$f_4 = \frac{P_S}{P_H} \quad (3.3.4)$$

donde  $P_S$  y  $P_H$  son los perímetros de las semillas y del hilo respectivamente.

$f_5$  Compacticidad de la semilla.

$$f_5 = C_S = \frac{P_S^2}{A_S} \quad (3.3.5)$$

$f_6$  Compacticidad del hilo.

$$f_6 = C_H = \frac{P_H^2}{A_H} \quad (3.3.6)$$

$f_7$  Relación entre el eje mayor de la semilla ( $L_{XS}$ ) y el eje mayor del hilo ( $L_{XH}$ ).

$$f_7 = \frac{L_{XS}}{L_{XH}} \quad (3.3.7)$$

$f_8$  Relación entre el perímetro de la semilla ( $P_S$ ) y el eje mayor del hilo ( $L_{XH}$ ).

$$f_8 = \frac{P_S}{L_{XH}} \quad (3.3.8)$$

$f_9$  Relación entre las distancias de  $L_1$  y  $L_2$  que son mostradas en la *figura 3.5*.

$$f_9 = \frac{L_1}{L_2} \quad (3.3.9)$$

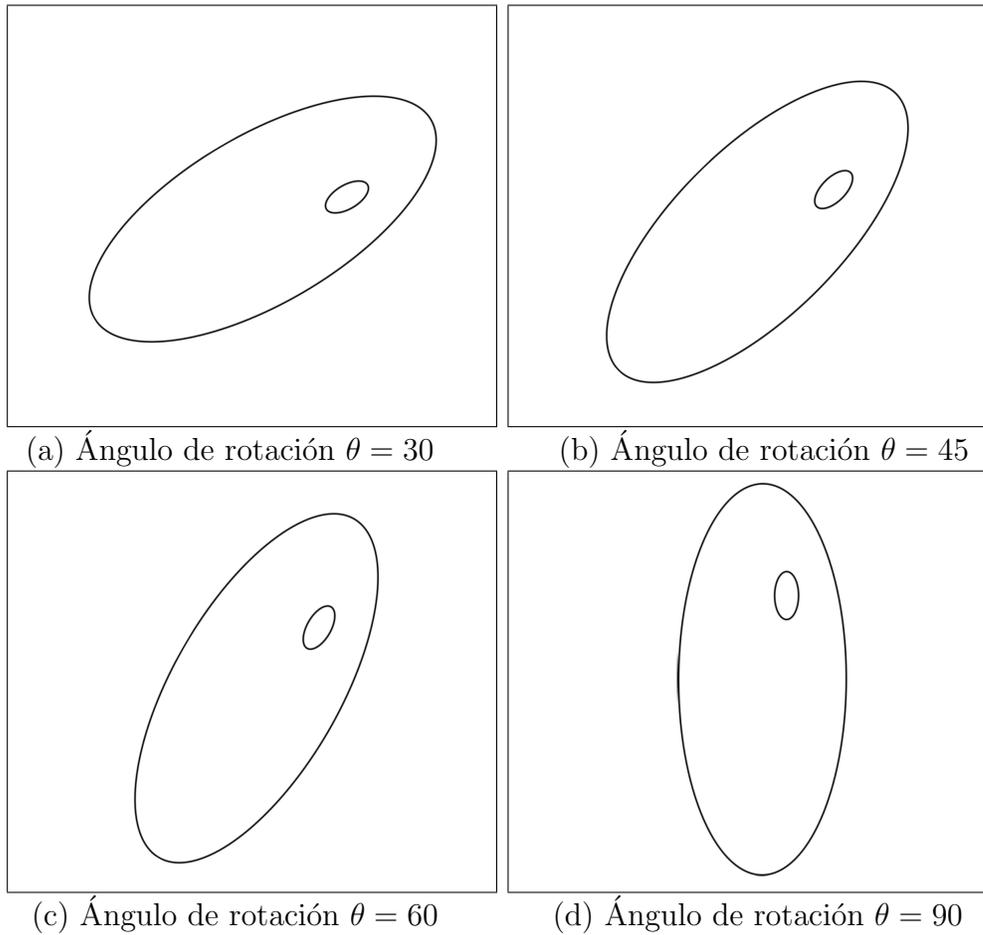
Las características morfológicas listadas son aplicables a semillas con hilo o bien a imágenes de semillas donde el hilo es visible. Para el caso de semillas sin hilo o con hilo no visible, solo  $f_5$  (compacticidad de la semilla) es aplicable. La *Tabla 3.4* lista las características utilizables para cada caso.

Parámetro	Con hilo	Sin hilo
$f_1$	✓	✗
$f_2$	✓	✗
$f_3$	✓	✗
$f_4$	✓	✗
$f_5$	✓	✓
$f_6$	✓	✗
$f_7$	✓	✗
$f_8$	✓	✗
$f_9$	✓	✗

Tabla 3.4: Parámetros usados para semillas.

### 3.3.1. Demostración para diferentes orientaciones

Con la finalidad de demostrar que las características morfológicas anteriormente mencionadas son invariantes a la orientación, en este párrafo se realiza el análisis morfológico para una semilla con hilo descrita idealmente por elipses. La *figura 3.6* muestra los casos de estudio a analizar.



**Figura 3.6: Semillas rotadas con diferentes ángulos ( $\theta$ ).**

Las imágenes ( $1187 \times 1017$  píxeles) fueron tratadas en MATLAB para realizar primeramente el cálculo de áreas ( $mm^2$ ) y perímetros ( $mm$ ) de las semillas y del hilo por separado para ser comparados con los resultados provenientes de fórmulas 3.2.9 y 3.2.10. Las *Tablas 3.5* y *3.6* muestran estos resultados para semilla e hilo respectivamente, en las diversas rotaciones.

Ángulo de rotación	MATLAB		Fórmula	
	Área	Perímetro	Área	Perímetro
$\theta = 0$	13153	523.43	13195	524.27
$\theta = 30$	13155	524.31	13195	524.27
$\theta = 45$	13153	523.92	13195	524.27
$\theta = 60$	13153	524.38	13195	524.27
$\theta = 90$	13146	523.43	13195	524.27

Tabla 3.5: Resultados de la semilla obtenidos por MATLAB y fórmula.

Ángulo de rotación	MATLAB		Fórmula	
	Área	Perímetro	Área	Perímetro
$\theta = 0$	560.42	93.901	565.49	94.413
$\theta = 30$	560.52	94.233	565.49	94.413
$\theta = 45$	560.43	94.028	565.49	94.413
$\theta = 60$	560.423	93.979	565.49	94.413
$\theta = 90$	560.12	93.92	565.49	94.413

Tabla 3.6: Resultados del hilo obtenidos por MATLAB y fórmula.

Tomando como referencia los cálculos de área y perímetros obtenidos por fórmula, se obtiene el error relativo de las mediciones hechas por MATLAB. Las *Tablas 3.7* y *3.8* muestran los valores de error (en porcentaje) para semilla e hilo respectivamente, en las diversas rotaciones.

Ángulo de rotación	Área	Perímetro
$\theta = 0$	0.31424	0.15964
$\theta = 30$	0.29748	0.006537
$\theta = 45$	0.31731	0.067665
$\theta = 60$	0.31941	0.020191
$\theta = 90$	0.37142	0.16082

Tabla 3.7: Error relativo de área y perímetro de la semilla.

Ángulo de rotación	Área	Perímetro
$\theta = 0$	0.89617	0.54205
$\theta = 30$	0.87893	0.19042
$\theta = 45$	0.89352	0.40809
$\theta = 60$	0.8944	0.45977
$\theta = 90$	0.94876	0.52278

Tabla 3.8: Error relativo de área y perímetro del hilo.

Los resultados mostrados permiten concluir que área y perímetro son invariantes a la rotación o bien tienen un error con respecto a los valores obtenidos por fórmulas menores al 1 %.

Finalmente, las características morfológicas  $f_1 - f_9$  fueron calculadas para todas las orientaciones. La *Tabla 3.9* resume estos resultados, los cuales demuestran que las características son invariantes a la rotación.

Rotación	$f_1$	$f_2$	$f_3$	$f_4$	$f_5$	$f_6$	$f_7$	$f_8$	$f_9$
$\theta = 0$	4.3698	0.32696	22.47	5.5743	21.757	15.734	6.0474	13.125	2.431
$\theta = 30$	4.3698	0.33983	22.47	5.5639	21.826	15.842	6.0475	13.146	2.4268
$\theta = 45$	4.3703	0.33989	22.469	5.5719	21.798	15.776	6.0467	13.136	2.4263
$\theta = 60$	4.37	0.33982	22.469	5.5797	21.837	15.759	6.047	13.148	2.4271
$\theta = 90$	4.3702	0.32754	22.469	5.5732	21.769	15.748	6.0475	13.126	2.4297

Tabla 3.9: Parámetros morfológicos para modelos elípticos rotados.

### 3.4. Casos de estudio

En esta sección se presentan diferentes casos de estudio, consistentes en imágenes de semillas provenientes de la base de datos PaDIL [18]. En todos los casos, se tomaron dos versiones de las imágenes bajo estudio, la imagen original y una imagen ampliada al doble. Se traron imágenes con 3 vistas de la semilla, lateral, dorsal y ventral. Es preciso señalar que el hilo de la semilla resulta oculto o semi-oculto en alguna de las vistas. A partir de las imágenes de bordes resultantes del procesamiento en paralelo de la red memristiva, se obtuvieron las diversas características morfológicas  $f_1 - f_9$ , área, perímetro, redondez, eje mayor y eje menor.

### 3.4.1. Vicia Sativa

La Vicia Sativa es una planta con flores de la familia Fabaceae (leguminosa) considerada maleza en varios países; sin embargo, la norma mexicana sobre malezas [24] no la reporta como tal. La 3.7 muestra tres vistas de la semilla.



Figura 3.7: Vicia Sativa.

El resultado de detección de bordes en las imágenes se muestra en la figura 3.8. Dado que la única vista que muestra el hilo es la vista ventral, los parámetros morfológicos  $f_1 - f_9$  son obtenidos a partir de ella. Los resultados se muestran en la Tabla 3.10 para las imágenes de tamaño normal y doble.

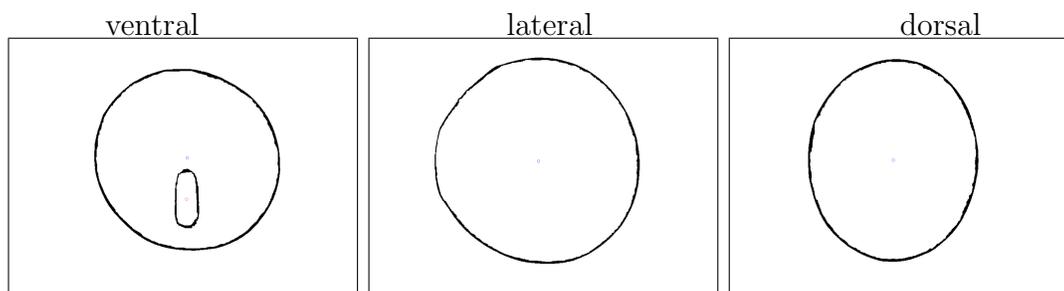


Figura 3.8: Vicia Sativa: bordes.

	$f_1$	$f_2$	$f_3$	$f_4$	$f_5$	$f_6$	$f_7$	$f_8$	$f_9$
ventral×1	4.5871	1.1025	24.831	4.32	13.79	18.347	3.2996	10.06	1.5557
ventral×2	4.5861	1.10245	24.831	4.4721	16.275	20.206	3.2996	10.929	1.5557

Tabla 3.10: Parámetros morfológicos de Vicia Sativa.

Adicionalmente, las características de compacticidad y redondez se resumen en la *Tabla 3.11*, donde además, a manera de verificación, se agregan columnas para el producto  $CRdProduct$  y para  $\pi$ .

	Redondez	Compacticidad	$CRdProduct$	$RC\pi$
ventral $\times$ 1	0.9113	13.79	12.566	3.1416
ventral $\times$ 2	0.77213	16.275	12.566	3.1416
lateral $\times$ 1	0.92397	13.6	12.566	3.1416
lateral $\times$ 2	0.78129	16.084	12.566	3.1416
dorsal $\times$ 1	0.92505	13.584	12.566	3.1416
dorsal $\times$ 2	0.78269	16.055	12.566	3.1416

**Tabla 3.11: Redondez y compacticidad para Vicia Sativa.**

Por último, los principales parámetros geométricos de los modelos elípticos del hilo y de la semilla en sus diferentes vistas se muestran en las *Tablas 3.12* y *3.13*, respectivamente.

	Área ( $mm^2$ )	Perímetro( $mm$ )	Eje Mayor( $mm$ )	Eje Menor ( $mm$ )
ventral $\times$ 1	0.45285	2.8824	1.2378	0.4736
ventral $\times$ 2	0.4452	2.9993	1.2273	0.469595

**Tabla 3.12: Características geométricas del hilo de Vicia Sativa.**

	Área ( $mm^2$ )	Perímetro( $mm$ )	Eje Mayor( $mm$ )	Eje Menor ( $mm$ )
ventral $\times$ 1	11.697	12.452	4.0843	3.7815
ventral $\times$ 2	11.5	13.413	4.0497	3.7495
lateral $\times$ 1	14.836	14.205	4.4003	4.2972
lateral $\times$ 2	14.586	15.317	4.363	4.2608
dorsal $\times$ 1	12.029	12.783	4.2772	3.5815
dorsal $\times$ 2	11.826	13.78	4.2409	3.5512

**Tabla 3.13: Características geométricas de la semilla de Vicia Sativa.**

### 3.4.2. Trifolium Subterraneum.

Esta es una planta originaria del sur de Europa de hábitos rastreros. Pertenecce a la familia de las fabáceas, es de color morado o rojizo y su longitud es de  $2mm$  a  $4mm$ . Para esta semilla se tienen 4 tipos de vistas de la base de datos [18], las cuales se muestran en la *figura 3.9*, donde además de las tres vistas ya mencionadas se ha agregado una vista perpendicular al hilo.



Figura 3.9: *Trifolium Subterraneum*.

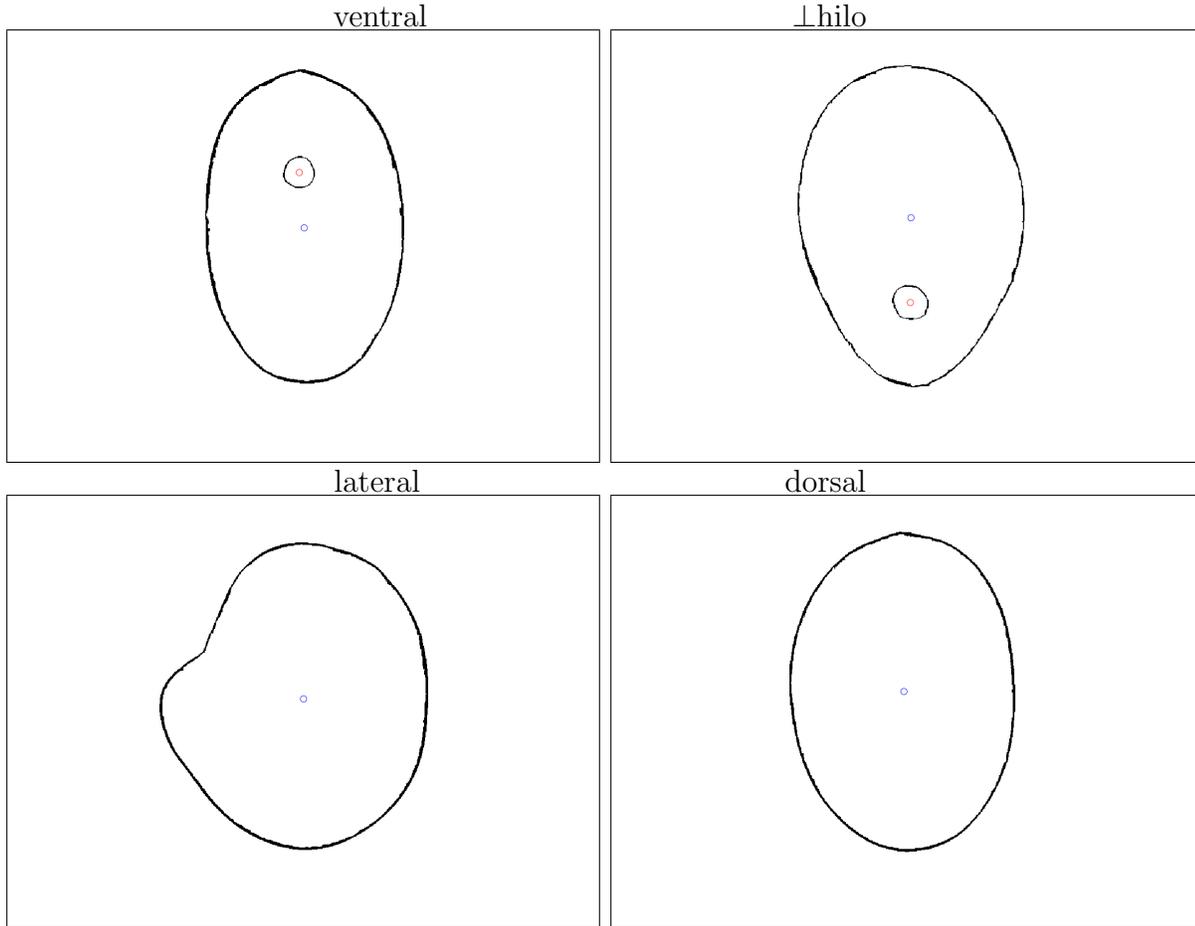


Figura 3.10: *Trifolium Subterraneum*: bordes.

Por un lado, los resultados de las características morfológicas  $f_1 - f_9$  se muestran en la *Tabla 3.14*; mientras que la compacticidad y la redondez se muestran en la *Tabla 3.15*.

	$f_1$	$f_2$	$f_3$	$f_4$	$f_5$	$f_6$	$f_7$	$f_8$	$f_9$
ventral×1	5.6017	0.002079	70.492	9.1081	14.873	12.638	10.545	28.278	2.0918
ventral×2	5.6016	0.002079	70.492	9.0176	17.565	15.227	10.545	30.731	2.099
⊥hilo×1	3.7046	0.054087	62.72	8.4352	14.591	12.862	9.0238	25.268	3.8012
⊥hilo×2	3.7043	0.054087	62.72	8.4135	17.348	15.371	9.0238	27.552	3.8012

Tabla 3.14: Parámetros morfológicos de *Trifolium Subterraneum*.

	Redondez	Compactidad	$CRdProduct$	$RC\pi$
ventral×1	0.84492	14.873	12.566	3.1416
ventral×2	0.71542	17.565	12.566	3.1416
⊥hilo×1	0.86123	14.591	12.566	3.1416
⊥hilo×2	0.72437	17.348	12.566	3.1416
lateral×1	0.90162	13.938	12.566	3.1416
lateral×2	0.75317	16.685	12.566	3.1416
dorsal×1	0.90594	13.871	12.566	3.1416
dorsal×2	0.76778	16.367	12.566	3.1416

Tabla 3.15: Redondez y compactidad para *Trifolium Subterraneum*.

Las características geométricas de los modelos elípticas se muestran en las *Tablas 3.16* y *3.17* para el hilo (en dos vistas) y la semilla, respectivamente.

	Área ( $mm^2$ )	Perímetro( $mm$ )	Eje Mayor( $mm$ )	Eje Menor ( $mm$ )
ventral×1	0.058805	0.86208	0.27767	0.27056
ventral×2	0.057421	0.93505	0.27438	0.26735
⊥hilo×1	0.05718	0.85759	0.28629	0.25532
⊥hilo×2	0.056002	0.92779	0.28332	0.25268

Tabla 3.16: Características geométricas del hilo de *Trifolium Subterraneum* (para dos vistas).

	Área ( $mm^2$ )	Perímetro( $mm$ )	Eje Mayor( $mm$ )	Eje Menor ( $mm$ )
ventral×1	4.2041	7.8519	2.9281	1.8559
ventral×2	4.1051	8.432	2.8935	1.8339
⊥hilo×1	3.6435	7.2339	2.5834	1.8202
⊥hilo×2	3.5684	7.806	2.5566	1.8014
lateral×1	6.4181	9.4579	3.1432	2.625
lateral×2	6.3334	10.28	3.1224	2.6076
dorsal×1	5.0444	8.3649	3.0174	2.1299
dorsal×2	4.9236	8.9769	2.9811	2.1042

Tabla 3.17: Características geométricas de la semilla de *Trifolium Subterraneum*.

### 3.4.3. *Crotalaria Zanzibarica*

La *Crotalaria Zanzibarica* es una planta de la familia Fabaceae que consiste en un arbusto con racimo alargado de pétalos amarillos, cuyas semillas son de color naranja a marrón rojizo. En algunos países se le considera especie invasora mientras que la norma mexicana no la cataloga como tal. Las imágenes originales y su tratamiento con detección de bordes se muestran en las *figuras 3.11* y *3.12*, respectivamente.



Figura 3.11: *Crotalaria Zanzibarica*.

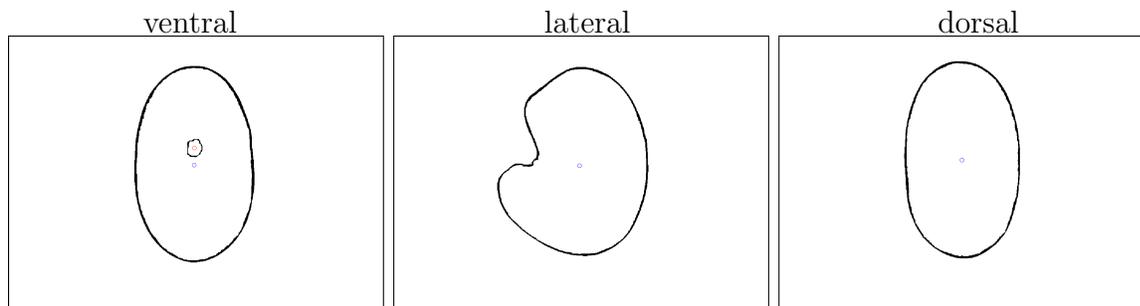


Figura 3.12: *Crotalaria Zanzibarica*: bordes.

En este caso de estudio igual se presentan los mismos parámetros ya mencionados con anterioridad, los resultados se muestran en las *Tablas 3.18*, *3.19*, *3.20* y *3.21*.

<i>Crotalaria</i>	$f_1$	$f_2$	$f_3$	$f_4$	$f_5$	$f_6$	$f_7$	$f_8$	$f_9$
ventral×1	11.497	0.002715	102.62	10.375	14.502	13.824	11.864	30.681	1.5346
ventral×2	11.4965	0.00271505	102.62	10.311	16.884	16.297	11.864	33.105	1.5346

Tabla 3.18: Parámetros morfológicos de *Crotalaria Zanzibarica*.

Crotalaria	Redondez	Compacticidad	$CRdProduct$	$RC\pi$
ventral×1	0.86654	14.502	12.566	3.1416
ventral×2	0.74429	16.884	12.566	3.1416
lateral×1	0.79013	15.904	12.566	3.1416
lateral×2	0.66456	18.909	12.566	3.1416
dorsal×1	0.85446	14.707	12.566	3.1416
dorsal×2	0.73274	17.15	12.566	3.1416

Tabla 3.19: Redondez y compacticidad para Crotalaria Zanzibarica.

Crotalaria	Área ( $mm^2$ )	Perímetro( $mm$ )	Eje Mayor( $mm$ )	Eje Menor ( $mm$ )
ventral×1	0.026836	0.60908	0.20597	0.16804
ventral×2	0.026441	0.65644	0.20445	0.1668

Tabla 3.20: Características geométricas del hilo de Crotalaria Zanzibarica.

Crotalaria	Área ( $mm^2$ )	Perímetro( $mm$ )	Eje Mayor( $mm$ )	Eje Menor ( $mm$ )
ventral×1	2.7806	6.3194	2.4436	1.4719
ventral×2	2.7398	6.7684	2.4256	1.461
lateral×1	3.69	7.6607	2.5974	1.8765
lateral×2	3.6111	8.2633	2.5694	1.8563
dorsal×1	3.6605	7.3372	2.8309	1.6483
dorsal×2	3.5986	7.8558	2.8068	1.6343

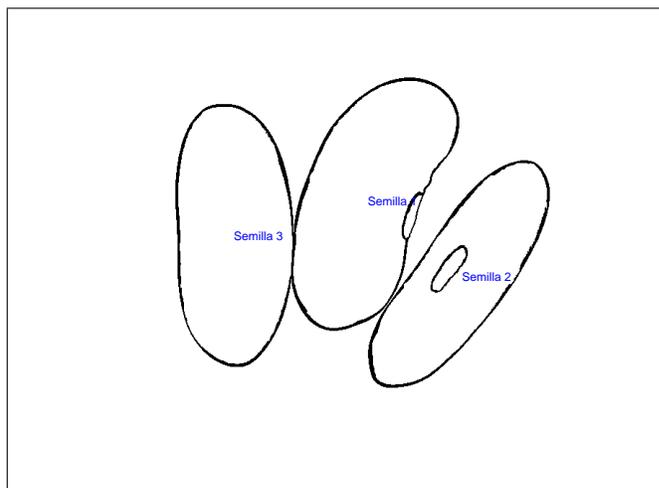
Tabla 3.21: Características geométricas de la semilla de Crotalaria Zanzibarica.

### 3.4.4. Frijol

El frijol es una leguminosa ampliamente extendida y constituye una importante fuente de alimento en varias regiones del mundo. Existen múltiples especies del mismo, en este párrafo tratamos la especie *Phaseolus vulgaris* var. belleza marrón. Adicionalmente, con la finalidad de realizar un tratamiento de imágenes que contienen un aglomerado de semillas, la imagen a tratar se muestra en la *figura 3.13*. la correspondiente imagen con bordes detectados se muestra en la *figura 3.14*.



**Figura 3.13: Frijol.**



**Figura 3.14: Frijol: bordes.**

En este caso de estudio se tienen 3 semillas a analizar. Por la disposición de las mismas, resulta claro que en la semilla 3 no puede realizarse análisis que involucre al hilo dado que este no es visible. Adicionalmente, si bien las semillas 1 y 2 muestran el hilo, por la perspectiva de cada una de ellas, es obvio que los resultados del análisis morfológico arrojan cifras muy diferentes. Las *Tablas 3.22* y *3.23* muestran estos resultados.

Semilla	$f_1$	$f_2$	$f_3$	$f_4$	$f_5$	$f_6$	$f_7$	$f_8$	$f_9$
1	5.8405	1.5524	86.882	6.9942	17.259	30.653	6.0042	14.303	1.0535
2	19.123	1.6875	32.461	5.5196	21.074	22.454	5.2288	11.62	1.0415

Tabla 3.22: Parámetros morfológicos del frijol.

Semilla	Redondez	Compactidad	$CRdProduct$	$RC\pi$
1	0.72812	17.259	12.566	3.1416
2	0.59628	21.074	12.566	3.1416
3	0.77317	16.253	12.566	3.1416

Tabla 3.23: Redondez y compactidad para el frijol.

Los parámetros geométricos del modelo elíptico para el hilo y la semilla se muestran en las *Tablas 3.24* y *3.25*, respectivamente.

Semilla	Área ( $mm^2$ )	Perímetro( $mm$ )	Eje Mayor( $mm$ )	Eje Menor ( $mm$ )
1	0.95451	5.4091	2.645	0.48032
2	1.8288	6.4081	3.0439	0.7761

Tabla 3.24: Características geométricas del hilo de frijol.

Semilla	Área ( $mm^2$ )	Perímetro( $mm$ )	Eje Mayor( $mm$ )	Eje Menor ( $mm$ )
1	83.884	37.832	15.881	6.9094
2	61.193	35.37	15.916	5.1602
3	78.656	35.755	15.103	6.6593

Tabla 3.25: Características geométricas de la semilla de frijol.

### 3.4.5. *Vaccaria Hispanica*

Esta hierba sí está catalogada como maleza por la norma mexicana [24]. Sus semillas son de color negro y de 2 a 2.5  $mm$  de tamaño. En este caso de estudio se analiza un aglomerado de estas semillas que se muestra en la *figura 3.15*. Los correspondientes bordes se muestran en la *figura 3.16*.



Figura 3.15: Vaccaria Hispanica.

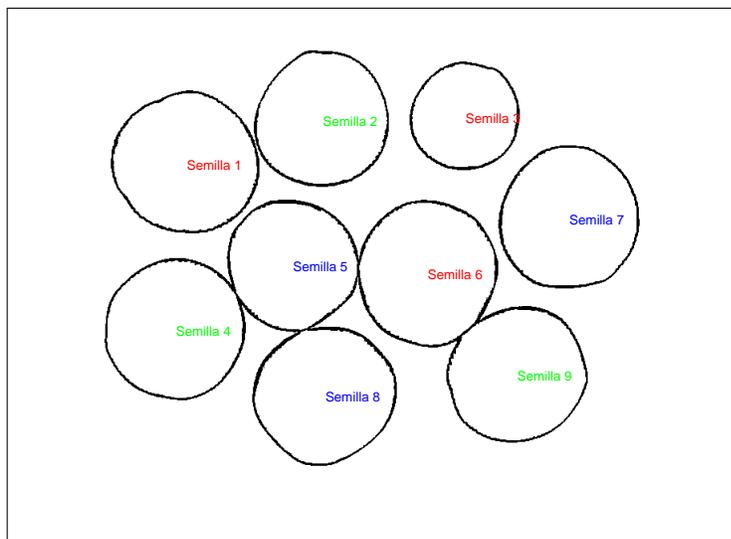


Figura 3.16: Vaccaria Hispanica: bordes.

Por último, las *Tablas 3.26* y *3.27* resumen los resultados morfológicos.

Semilla	Redondez	Compactidad	$CRdProduct$	$RC\pi$
1	0.91006	13.808	12.566	3.1416
2	0.93198	13.484	12.566	3.1416
3	0.96004	13.089	12.566	3.1416
4	0.93698	13.412	12.566	3.1416
5	0.87404	14.377	12.566	3.1416
6	0.87525	14.358	12.566	3.1416
7	0.95471	13.162	12.566	3.1416
8	0.92189	13.631	12.566	3.1416
9	0.91824	13.685	12.566	3.1416

Tabla 3.26: Redondez y compactidad para *Vaccaria Hispanica*.

Semilla	Área ( $mm^2$ )	Perímetro( $mm$ )	Eje Mayor( $mm$ )	Eje Menor ( $mm$ )
1	4.2356	7.6476	2.3823	2.3823
2	4.2356	7.0828	2.2222	2.2222
3	2.3781	5.5793	1.7761	1.7761
4	4.0613	7.3803	2.2941	2.2941
5	3.4622	7.0553	2.1355	2.1355
6	4.1374	7.7073	2.3618	2.3618
7	4.1253	7.3688	2.3407	2.3407
8	3.9495	7.3373	2.3064	2.3064
9	3.8638	7.2717	2.2868	2.2868

Tabla 3.27: Características geométricas de la semilla de *Vaccaria Hispanica*.



# Conclusiones y Trabajo Futuro

---

En la presente tesis de maestría se ha demostrado la viabilidad para utilizar la red memristiva como procesador analógico para detectar bordes en imágenes. En específico, se ha demostrado que la implementación en paralelo, con ayuda de GPU, no sólo preserva la calidad en la detección de bordes, sino también reduce los tiempos de cómputo.

El tamaño de la red memristiva es igual al número de píxeles ( $n \times m$ ) de la imagen original y el proceso de solución exige que las ecuaciones nodales asociadas sean resueltas en un análisis transitorio. Los memristores que conforman la red son modelados por funciones controladas por carga  $M(q)$  que se hallan expresadas en forma analítica, lo que conlleva a que la evaluación repetitiva durante el proceso de solución sea muy eficiente.

Se ha realizado el tratamiento de imágenes provenientes de una base de prueba con la finalidad de comparar la implementación paralelo (GPU) con la implementación normal. Los resultados cualitativos y cuantitativos son similares para ambas implementaciones, con una sensible reducción en el tiempo de ejecución para la versión en paralelo.

Por último, se han tratado imágenes de semillas con la finalidad de determinar sus características morfológicas. Para este fin, se ha recurrido a un modelo geométrico de la semilla basado en la elipse, después de compararlo con la alternativa del ovoide.

## Trabajo Futuro

El presente trabajo constituye el primer paso para obtener un conocimiento más certero sobre las características morfológicas de semillas que pueda ser utilizado en el futuro en aplicaciones específicas. En particular, uno de los problemas más apremian-

tes en el manejo de semillas es la detección de semillas de malezas, las cuales pueden ser de especies invasoras con alto riesgo para cultivos, bosques y flora autóctonas. Los siguientes puntos son algunos de los tópicos futuros de investigación:

- Detección de malezas.
- Añadir clasificadores a la actual implementación de la red memristiva.
- Generar una base de datos morfológicos de semillas.
- Ampliar el procesamiento de la red memristiva al tratamiento de otro tipo de imágenes, e.g. aplicaciones médicas, industriales, etc.

# Bibliografía

---

- [1] A. Sarmiento-Reyes, J. M. Ugalde Franco, Y. Rodríguez-Velásquez, y J. L. F. Palomeque Loyo, “Development of an operator-based fully analytical charge-controlled memristor model,” en *2018 15th International Conference on Electrical Engineering, Computing Science and Automatic Control (CCE)*, pp. 1–4, Sep. 2018.
- [2] L. Chua, “Memristor-the missing circuit element,” *IEEE Transactions on Circuit Theory*, vol. 18, pp. 507–519, Sep. 1971.
- [3] Y. Rodríguez-Velásquez, “Development of an analytical model for a charge-controlled memristor and its applications,” Master’s thesis, Instituto Nacional de Astrofísica, Óptica y Electrónica, Agosto 2019.
- [4] J. Hutchinson, C. Koch, J. Luo, y C. Mead, “Computing motion using analog and binary resistive networks,” *Computer*, vol. 21, pp. 52–63, March 1988.
- [5] F. Jiang y B. E. Shi, “The memristive grid outperforms the resistive grid for edge preserving smoothing,” en *2009 European Conference on Circuit Theory and Design*, pp. 181–184, Aug 2009.
- [6] A. Sarmiento-Reyes y Y. R. Velásquez, “A charge-controlled memristor model for image edge detection with a memristive grid,” en *2017 International Caribbean Conference on Devices, Circuits and Systems (ICCDCS)*, pp. 49–52, June 2017.
- [7] D. Strukov, G. Snider, D. Stewart, y S. Williams, “The missing memristor found,” *Nature*, vol. 453, pp. 80–3, 06 2008.
- [8] L. Chua y P. Lin, *Computational Methods in CAD*. Academic Press, 1973.

- 
- [9] J. Vlach y K. Singhal, *Computer Methods for Circuit Analysis and Design*. Van Nostrand Reinhold Company, 1983.
- [10] L. Chua, C. Desoer, y E. Kuh, *Linear and Nonlinear Circuits*. McGraw-Hill International Editions, 1987.
- [11] A. Schwarz, *Computer-Aided Design of Microelectronic circuits and Systems: Volume 1*. Academic Press, 1987. ISBN 0-12-632431-X.
- [12] Shane Cook, *CUDA programming: a developer's guide to parallel computing with GPU*. Elsevier., 2013.
- [13] J. Cheng y T. M. Max Grossman, *Professional CUDA C programming*. John Wiley & Sons, 2014.
- [14] Tolga Soyata, *GPU parallel program development using CUDA*. Chapman & Hall/CRC Computational Science Series, 2018.
- [15] Jason Sanders and Edward Kandrot, *CUDA by example an introduction to general-purpose gpu programming*. Boston: Addison-wesley, 2010.
- [16] D. Storti y M. Yurtoglu, *CUDA for engineers and introduction to high-performance parallel computing*. Addison-wesley, 2015.
- [17] D. R. Martin, C. C. Fowlkes, y J. Malik, "Learning to detect natural image boundaries using local brightness, color, and texture cues," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, pp. 530–549, May 2004.
- [18] Australian Government Department of Agriculture, "Padil – high quality images and information tools designed for biosecurity and biodiversity." Available: <http://www.padil.gov.au/>, [en línea]. [Accedido: 18-Nov- 2019].
- [19] N. Yamamoto, "Tdcc laboratory)." Disponible en línea: <http://nyjp07.com/index-egg5-E.html/>. [Accedido: 1-May-2019].
- [20] Charles H. Lehmann, *Geometría analítica*. Limusa., 1989.
- [21] M. Villarino, "Ramanujan's perimeter of an ellipse," 07 2005.

- 
- [22] C. Shi y G. Ji, "Recognition method of weed seeds based on computer vision," en *2009 2nd International Congress on Image and Signal Processing*, pp. 1–4, Oct 2009.
- [23] C. Shi y G. Ji, "Study of recognition method of leguminous weed seeds image," en *2009 International Workshop on Intelligent Systems and Applications*, pp. 1–4, May 2009.
- [24] SAGARPA, "Nom-043-fito-1995. norma oficial mexicana, especificaciones para prevenir la introducción de malezas cuarentenarias a México," Mar 2000.