



Universidad Autónoma de Querétaro
Facultad de Ingeniería
Especialidad Sistemas Mecatrónicos

NOMBRE DE LA TESIS

Medición de parámetros y características geométricas en especies de hojas mediante procesamiento de imágenes y redes neuronales

Que como parte de los requisitos para obtener el grado de
Ingeniero en Sistemas Mecatrónicos

Presenta:

Romero González Julio Alejandro

Dirigido por:

Dr. Damián Vargas Vázquez

SINODALES

Dr. Damián Vargas Vázquez
Presidente

Dra. Ma. Del Carmen Espino Gudiño
Secretario

Dra. Angélica Rosario Jiménez Sánchez
Vocal

M. en C. Aurora Femat Díaz
Suplente


Firma


Firma


Firma


Firma

Centro Universitario
Querétaro, Qro.
09 de enero de 2012
México

Resumen

En la actualidad existen distintas clases de programas informáticos para identificar especies de plantas o flora, pero este tipo de software usa claves dicotómicas que dependen de respuestas humanas para poder llevar a cabo la identificación. Por otra parte, existen herramientas informáticas para evaluar características morfológicas de los elementos del follaje de forma automatizada, así como equipo comercial que ofrece mediciones aproximadas de área foliar. Sin embargo, hasta el momento existen pocas herramientas disponibles no comerciales que ofrezcan información completa acerca de los distintos parámetros de las hojas (área foliar, longitud, anchura, perímetro, circularidad, etc.); además, en la actualidad aún no se reporta una alternativa para clasificar especies de hojas de forma automática y sin intervención humana, basándose en las características morfológicas distintivas de cada especie. Debido a lo anterior, la importancia de desarrollar un algoritmo capaz de ofrecer de manera fiable los distintos parámetros de las hojas, además de poder identificar distintas especies basándose en dichos parámetros y en la estructura geométrica distintiva de cada especie, la cual pueda ser ampliada de manera gradual con la incorporación de nuevas especies, y que ofrezca una alternativa económica a los productos existentes en el mercado.

Palabras clave: Redes neuronales, segmentación, clasificación, especies.

Summary

At present there are different types of software to identify species of plants and flora, but this type of software uses dichotomous keys that depend on human responses in order to perform the identification. On the other hand, there are tools to evaluate the morphological characteristics of the foliage elements automatically, and including commercial equipment that provides approximate measurements of leaf area. However, up to now there are few non-commercial tools available that provide comprehensive information about the various parameters of the leaves (leaf area, length, width, perimeter, circularity, etc.). Besides, at present there is not reported an alternative for classifying leaves species automatically without human intervention, based on morphological characteristics of each species. Because of this, the importance of developing an algorithm that can reliably provide the various parameters of the leaves, besides being able to identify different species based on these parameters and the distinctive geometric structure of each species, which can be expanded gradually with the addition of new species, and that it can offer an economical alternative to existing products on the market.

Keywords: neural networks, segmentation, classification, species.

AGRADECIMIENTOS

Al término de esta etapa de mi vida, quiero expresar un profundo agradecimiento a quienes con su ayuda, apoyo, comprensión, amistad, compañía, ánimo, me han ayudado a cruzar con firmeza el camino de la superación, porque con su apoyo y aliento hoy he logrado uno de mis más grandes anhelos. Es cierto que cuando se presenta el momento de decir gracias, hacen falta palabras que realmente siente el corazón, agradezco de todo corazón a Dios y a mis Padres porque a través de ellos me concedió la vida en este mundo, y porque gracias a su apoyo y consejos han guiado mi camino. Así como a mis abuelos que para mí han sido como mis padres y quienes me han formado moralmente como ser humano y profesional, quiero agradecerles tantos sacrificios, esfuerzos de su vida para educarme, el amor con el que han entregado su vida para convertirme en un hombre de provecho. A mi tía quien con su ayuda, apoyo y comprensión me ha alentado y motivado a buscar un mejor futuro y edificación de mi vida, por ese cariño que siempre he recibido en los momentos buenos y malos de mi vida, hago este triunfo compartido, esperando que comprendas que mis ideales son inspirados en tu nobleza. A mis amigos quienes directa o indirectamente han sido inspiradores, a mi mejor amiga a quien jamás encontraré la forma de agradecer su apoyo, comprensión, confianza, sonrisas, espero que comprendas que muchos de mis logros han sido por la fortaleza de tu apoyo.

ÍNDICE

Resumen	ii
Summary	iii
Agradecimientos	iv
Índice	v
1. INTRODUCCIÓN.....	1
1.1. Hipótesis.....	3
1.2. Objetivo General.....	3
1.3. Objetivos específicos.....	3
1.4. Antecedentes.....	4
2. REVISIÓN DE LITERATURA.....	13
2.1. Entorno Builder C++.....	13
2.2. Momentos Invariantes.....	14
2.2.1. Momentos de segundo orden.....	15
a) Momento de inercia de fila.....	15
b) Momento de inercia de columna.....	15
c) Momento de inercia cruzado.....	16
2.2.2. Rotación.....	17
2.2.3. Momentos de tercer orden.....	17
2.3. Filtros pasa bajas.....	18
2.4. Método de OTSU para umbral optimo.....	20
2.5. Características morfológicas.....	22

2.5.1. Área foliar.....	22
2.5.2. Circularidad.....	22
2.5.3. Rectangularidad.....	23
2.5.4. Relación perímetro-longitud.....	23
2.5.5. Relación perímetro-ancho y longitud.....	23
2.5.6. Factor de elongación.....	23
2.6. Erosión.....	24
3. Redes neuronales Artificiales.....	25
3.1. Elementos de una RNA.....	28
3.2. Diseño de una RNA.....	29
3.3. Estructura de una RNA.....	33
3.4. Estructura de una Neurona Artificial.....	33
3.5. Clasificación de las RNA.....	37
3.5.1. Adeline-Madeline.....	37
3.5.2. Estructura Adaline.....	38
3.5.3. Estructura Madaline.....	39
3.5.4. Red propagación hacia adelante (feedforward)	40
3.5.5. Red propagación hacia atrás (Backpropagation)	41
3.6. Aprendizaje.....	42
4. Identificación de especies	43
4.1. Identificación y planteamiento del problema.....	43
4.2. Estructura del programa.....	47
4.2.1. Interfaz grafica.....	47
4.3. Adquisición de la imagen.....	49

4.4. Procesamiento.....	53
4.4.1. Segmentación por OTSU.....	55
4.4.2. Filtro mediana.....	56
4.5. Extracción de características.....	58
4.6. RNA Aplicación/Entrenamiento.....	58
4.7. Identificación.....	65
5. RESULTADOS.....	67
5.1. Pruebas sobre especie Durazno.....	67
5.2. Pruebas sobre especie Herdera.....	69
5.3. Pruebas sobre especie Melia.....	72
5.4. Pruebas sobre especie Melissa.....	74
5.5. Pruebas sobre especie Guayaba.....	76
5.6. Descriptor de forma: Firmas de Forma.....	79
6. Conclusiones.....	85
LITERATURA CITADA.....	86

1. INTRODUCCIÓN

La consecución de un criterio racional de clasificación de los vegetales ha sido una preocupación sobresaliente en la historia de la botánica. En 1576, el flamenco Matthias Lobelius agrupaba las plantas por su aspecto externo; pero puede considerarse a Cesalpino en 1583, como el primer inventor de un verdadero sistema botánico con su *plantis libri*. Pero fue en el siglo XVIII cuando se establecieron definitivamente las normas de nomenclatura y taxonomía, y se empezó a utilizar el moderno sistema de clasificación con el sueco Carl von Linné. Ningún método de clasificación tuvo el éxito incomparable del sistema sexual de Linné, gracias sobre todo a su nomenclatura binaria, la precisión, claridad de las descripciones, de la distinción de especies, variedades, y a la reunión de todas las conocidas en sus géneros (Arber, 1987).

En la actualidad los programas informáticos que existen se basan en la utilización de claves dicotómicas (se basa en definiciones de los caracteres morfológicos, macroscópicos o microscópicos) para identificar especies de plantas o flora, pero dependen de respuestas humanas para poder llevar a cabo la identificación. El sistema DELTA es un método que usa este método tenemos el sistema DELTA (lenguaje de descripción de la taxonomía) que es un método flexible para la codificación de las descripciones taxonómicas para el tratamiento informático, siendo un conjunto de programas desarrollados para la identificación interactiva y recuperación de información (Askevold et al., 1994). Así mismo ActKey usa claves y bases de datos que proporcionan una estrategia más amplia para la identificación de una planta desconocida; siendo un programa que está basado en una web, los visitantes pueden usar dichas

claves interactivas como herramientas para la identificación (Kuoh y Song, 2005).

Por otra parte, existen herramientas informáticas para evaluar características morfológicas de los elementos del follaje de forma automatizada, así como equipo comercial que ofrece mediciones aproximadas de área foliar. Sin embargo, hasta el momento existen pocas herramientas disponibles no comerciales que ofrezcan información completa acerca de los distintos parámetros de las hojas (área foliar, longitud, anchura, perímetro, circularidad, etc.); además, en la actualidad aún no se reporta una alternativa para clasificar especies de hojas de forma automática y sin intervención humana, basándose en las características morfológicas distintivas de cada especie.

Para el ser humano es fácil poder distinguir entre bordes, superficies, colores, formas, etc., con la desventaja de su velocidad y la tendencia a cometer errores, sin embargo un sistema computarizado es veloz y bien definido sus procesos además de que su error tiende a ser cero, pero le es difícil identificar y clasificar formas similares específicas (un borde liso o uno dentado, una figura ovalada, elíptica, acorazonada, lanceolada, etc., una superficie circular o un ondulada), puesto que representa un problema más complejo que implica juicio y discernimiento. Como principal objetivo de este proyecto es proponer un método basado en redes neuronales artificiales para la identificación de especies de hojas, un problema no muy sencillo debido a la gran cantidad de especies existentes y/o clasificadas hasta el momento. El problema a resolver es la similitud entre las especies, debido a que muchas de ellas comparten muchas características similares que inducen a una

equivocación en la identificación; parte de estas características se encuentra en la forma de la hoja, aun siendo tan distinto su borde o contorno.

1.1. Hipótesis

Mediante el diseño de un algoritmo basado en técnicas de procesamiento de imágenes y de redes neuronales, para identificar y clasificar distintos tipos de especies de hojas, basándose en características únicas de forma que comparte cada especie.

1.2. Objetivo General

Desarrollar un programa para la medición de parámetros en distintas especies de hojas para su evaluación y clasificación mediante el uso de técnicas de procesamiento de imágenes y de redes neuronales, el cual pueda ser actualizado y entrenado para identificar nuevos especímenes.

1.3. Objetivos específicos

- ✓ Proponer una herramienta de bajo costo
- ✓ Diseñar un programa que pueda ser actualizado para añadir nuevas especies y parámetros de comparación mediante una interfaz gráfica.
- ✓ Clasificar cierto número de especies mediante el procesamiento de imágenes

1.4. Antecedentes

Steinbach (2003) y Brickell (2003) describen las hojas como apéndices caulinares, en general verdes y aplanados, que nacen y se expanden lateralmente en los nudos de los tallos y ramificaciones. La función principal de las hojas es realizar la fotosíntesis en los cloroplastos de las células; debido a lo cual, los vegetales superiores son, junto a los otros organismos fotosintéticos, los productores primarios en la biosfera. Las hojas realizan el intercambio de gases (fotosíntesis y respiración) a través de sus estomas aeríferas, por los que además transpira el vapor de agua (evapotranspiración). A través de las estomas de las hojas, la planta toma el dióxido de carbono, CO_2 , de la atmósfera, y expulsa el O_2 procedente de la fotólisis del H_2O , proceso incluido en la fotosíntesis. Este oxígeno es fundamental para la vida en nuestro planeta. Típicamente, en la hoja se distinguen tres partes:

- Limbo
- Pecíolo
- Vaina

Limbo: el limbo o lámina, es la parte generalmente laminar plana, verde y ancha de la hoja; la cara superior se llama haz y la inferior envés; el haz suele ser de color oscuro y el envés algo más claro. La base del limbo se agranda a veces para albergar la yema, siempre presente en la axila de la hoja (yema axilar). El limbo está surcado por una serie de líneas o cordones, perfectamente visibles al trasluz y salientes por el envés, llamadas nerviaciones, nervaduras o nervios. Son hacecillos de conductos vasculares prolongación y ramificación de los del pecíolo, cuya misión es aportar la savia bruta y retirar la elaborada. En muchas hojas el nervio principal es central y finaliza en la punta del limbo (el ápice); del nervio principal suelen partir otros nervios secundarios, ver fig. 1.1.

Mediante las nervaduras del limbo se puede realizar clasificaciones de las hojas.



Figura 1.1 Partes de la hoja (botanical-online.com 2011)

Pecíolo: el pecíolo o pedúnculo foliar, es el filamento, en general delgado y de color verde, que une el limbo al tallo. Su haz suele ser plano o cóncavo, mientras que su envés suele ser convexo. Sus tejidos vasculares, que comunican la hoja con el tallo, permiten la llegada del agua y los minerales absorbidos por la raíz. Tiene además la capacidad de orientar a la hoja en la dirección de la luz solar.

Vaina: la vaina es la terminación ensanchada del pecíolo en el punto de unión con el tallo. Puede rodear al tallo muy claramente, como es el caso de la vaina cilíndrica de las gramíneas, o no existir. Algunas vainas llevan una prolongación membranosa en su parte superior llamada lígula. En la base del pecíolo, en ciertas especies, suelen encontrarse unas pequeñas laminillas o apéndices de distintos tipos, que pueden ser glandulares, espinosas, foliáceas o escamiformes, que reciben el nombre de estípulas. Las hojas sin pecíolo se llaman sentadas o sésiles.

La enorme variabilidad de las hojas permite clasificarlas en diversos tipos atendiendo a diferentes criterios:

- Por su nervadura
- Por el número y disposición de los folíolos
- Por su forma general
- Por la forma del borde
- Por la forma del limbo
- Por la forma del ápice
- Por la forma del margen
- Por la forma de la base
- Etc.

Durante el desarrollo de este proyecto por ahora nos enfocaremos en clasificarlas por la forma del limbo y por su forma de borde, debido a que abarcar las otras formas de clasificación implica mayor tiempo en el desarrollo y más herramientas, posiblemente podrán ser considerados los demás tipos para una actualización futura.

Tipos de hojas según las características y aspecto del limbo

Según la composición de las hojas en base a las características y aspecto del limbo, se las clasifica en simples o sencillas y compuestas. En las primeras, el pecíolo no se ramifica, siendo el limbo de una sola pieza. Las hojas compuestas presentan el limbo dividido en hojitas (los folíolos) que a su vez pueden subdividirse. La diferencia entre una hoja verdadera y un folíolo (que pueden ser grandes), reside en que en la axila de la primera hay una yema, de la que carecen los folíolos.

Las hojas simples pueden:

- Paripinnadas: cuando disponen de un número par de folíolos.
- Imparipinnadas: cuando se muestra un folíolo terminal, y por tanto existe un número impar de folíolos.

Según la forma del limbo, como puede apreciarse en la fig. 1.2, las hojas se pueden clasificar en:

- a) Acorazonadas: cuando su forma recuerda la de un corazón.
- b) Lanceoladas: cuando presenta una forma de lanza.
- c) Sagitadas: cuando su forma recuerda la de una saeta.
- d) Bilobuladas: cuando está partida o hendida en dos lóbulos, ejemplo del Ginkgo biloba.
- e) Elípticas: cuando presenta la forma de una elipse.
- f) Ovaladas: cuando tiene forma de óvalo.
- g) Compuestas palmeadas: cuando una hoja compuesta presenta divisiones o folíolos dispuestos como los dedos de una mano.
- h) Compuestas trifoliadas: cuando una hoja compuesta presenta tres divisiones o folíolos.



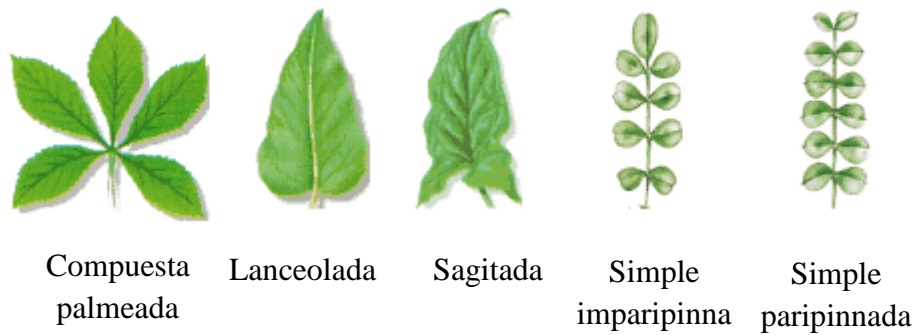


Figura 1.2 Algunos tipos de hojas clasificadas por las características y aspecto del limbo

Tipos de hojas según su borde

Las hojas también se pueden clasificar por su borde (ver fig. 1.3). Éste puede ser liso (entero); presentar indentaciones (borde dentado o con pequeños entrantes); hendiduras más o menos acusadas (borde lobulado, festoneado, partido).

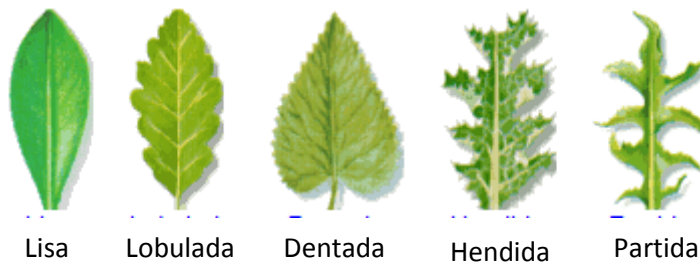


Figura 1.3 Algunos tipos de hojas clasificadas por su borde

Existe software que identifica especies de plantas o flora, pero este tipo de software usa claves dicotómicas que dependen de respuestas humanas para

poder llevar a cabo la identificación. Una clave dicotómica se basa en definiciones de los caracteres morfológicos, macroscópicos o microscópicos; de ella parten dos soluciones posibles, en función de si tienen o no tienen determinado carácter, repitiéndose el proceso de definiciones de características, hasta llegar al organismo en cuestión. Como ejemplo de software que usa este método tenemos:

Sistema Delta: el formato DELTA (lenguaje de descripción de la taxonomía) es un método flexible para la codificación de las descripciones taxonómicas para el tratamiento informático. Los datos del formato DELTA pueden ser utilizados para producir descripciones en lenguaje natural, claves convencionales e interactivas, y clasificaciones cladistas y fenético.

El sistema DELTA es un conjunto de programas desarrollados originalmente en Entomología de CSIRO. Se compone de: Intkey, un programa para la identificación interactiva y recuperación de información, el editor de Delta, un programa para crear y editar datos DELTA; Confor, un programa para traducir los datos DELTA en otros formatos. El sistema DELTA y varios conjuntos de datos están disponibles a través de la página de inicio DELTA: <http://delta-intkey.com>

Actkey: Es un programa de claves y bases de datos interactivo en línea para la identificación de los organismos. Se diferencia de las tradicionales claves dicotómicas en la prestación de los puntos de entrada de acceso múltiple. En lugar de responder a las preguntas de claves dísticas mientras se sigue un camino predefinido, ActKey proporciona una estrategia más amplia para la identificación de una planta desconocida. El programa está basado en

una web y es compatible con los navegadores de Internet más populares. Los visitantes pueden usar claves interactivas en el sitio web www.efloras.org como herramientas para la identificación, y una versión anterior de ActKey con más de 70 claves interactivas se puede encontrar en <http://flora.huh.harvard.edu:8080/actkey/>. Por otra parte, ActKey permite al taxónomo crear y editar una clave en línea y publicarlo instantáneamente. El sitio web www.efloras.org diseñado por Hong Song (2005) se utiliza para alojar la información de cada taxón y la muestra de espécimen, y ActKey es una de sus muchas características.

Hebestigma: Es un software que permite desarrollar las **habilidades identificar clasificar** los organismos vegetales correspondientes a la división Magnoliophyta. Este brinda varias alternativas como son: búsqueda a través de su ubicación taxonómica y caracteres diagnósticos, además de una galería de imágenes, nombres científicos y vernáculos, y un glosario de términos más utilizados, que actúan de forma hipervinculada permitiendo llegar a identificar y clasificar cada uno de los organismos vegetales.

WinFOLIA 2011a: Es un sistema comercial de análisis de imágenes a través de una computadora, la cual realiza mediciones morfológicas precisas sobre las hojas; comprende un hardware para la adquisición de imágenes (escáner o cámara y accesorios) y un programa de computadora (WinFOLIA), específicamente diseñado para calcular el área, morfología y enfermedades de las hojas. Ofrece diferentes configuraciones en función de la medición, velocidad precisión y portabilidad. La licencia para el uso del software de WinFolia está dividida en tres versiones con diferentes costos:

- WinFolia Básico es una solución de nivel de entrada de bajo costo para el análisis de las hojas. Se mide el área foliar, largo y ancho (en la dirección horizontal y vertical), perímetro, área y algunos agujeros de otras medidas.

- WinFolia Reg tiene todas las mediciones de la versión básica además de algunas mediciones morfológicas que son específicos de las hojas, como diferenciar la hoja del pecíolo, elegir si incluirlo o no en las medidas de área y la longitud, medir el ancho de distintas.

- WinFolia añade algunas mediciones morfológicas extendidas (como fractales) algunas de las cuales son específicas de las hojas. Esta versión también cuenta con el análisis de color que pueden ser utilizados para la enfermedad, la necrosis y la cuantificación de los insectos.

Otros métodos no están basados en claves dicotómicas, y se basan en las características morfológicas de los elementos del follaje. El área foliar, perímetro, circularidad, largo, anchura, circularidad, etc. son variables importantes para muchos estudios fisiológicos y agronómicos en donde se involucra el crecimiento de la planta, eficiencia fotosintética, e incluso puede darnos una clave para la clasificación y diferenciación de algunas especies.

Daughtry (1990) describió diferentes métodos directos de medición del índice de área foliar (IAF). Los métodos destructivos consisten en recoger las hojas directamente del árbol, midiendo la superficie foliar de todas las hojas de

la muestra, mientras que con los no destructivos la recolección de las hojas se hace mediante la colocación de trampas bajo la copa, midiendo IAF a partir del peso de las hojas recogidas en dichas trampas. Las técnicas planimétricas están basadas en la correlación existente entre la superficie de la hoja individual y el número de unidades que cubren un plano horizontal, mientras que las técnicas gravimétricas relacionan peso seco de hoja y superficie de hoja utilizando relaciones predeterminadas de superficie de hoja verde y peso seco (peso específico foliar, o masa de hoja por área).

En Bylesjö et al. (2008), se diseñó una herramienta informática (LAMINA) para proporcionar indicadores clásicos de la forma de la hoja (dimensiones de la hoja) y el tamaño (área), que normalmente se requiere para el análisis de correlación con la biomasa de la productividad, así como las medidas que indican la asimetría en la forma de la hoja, los rasgos de la hoja de dientes de sierra, y las medidas de daños herbivoría (falta de área foliar).

En O'Neal et al. (2002), se reporta un método para medir el área foliar y la defoliación mediante análisis digital de imágenes usando un escáner común y software de dominio público. Se comparó la exactitud y la precisión de este método a la de un medidor de área foliar estándar. Ambos métodos se utilizaron para medir los discos metálicos de un área conocida, el área de la soja (*Glycine max L.*) hojas, y la zona removido mediante la simulación de la alimentación de la hoja perforada.

Por otra parte, Femat et al. (2011) desarrolló un método para la medición de algunos parámetros en hojas basado en procesamiento de

imágenes para la medición de área y búsqueda de longitud y anchura en distintas especies de hojas. Lo anterior mediante el uso de un escáner convencional y software propio, comparando los resultados con un aparato comercial (LICOR LI-3000C) para su validación.

Antes de iniciar con el desarrollo del proyecto, es necesario tomar en cuenta conceptos, bases y estructuras que nos ayudarán en la comprensión de cómo se conforma la propuesta para la solución a este problema.

2. REVISIÓN DE LITERATURA

2.1. Programa de desarrollo C++ builder

C++ Builder es una aplicación Windows que proporciona un entorno de trabajo visual para construir aplicaciones Windows que integra distintos aspectos de la programación en un entorno unificado o integrado. La integración y facilidad de manejo hace que sea una herramienta indispensable para el desarrollo rápido de aplicaciones o RAD (Rapid Application Development). Guarda una gran similitud con el IDE de Visual Basic, aunque existen ciertas diferencias que veremos. El IDE de C++ Builder es una aplicación Windows 95 y como tal cumple con los estándares de aspecto, diseño y comportamiento que aconseja Microsoft a los desarrolladores de aplicaciones. En consecuencia, cualquiera que esté familiarizado con el manejo a nivel de usuario de Windows 95 no le supondrá ningún esfuerzo manejarlo con soltura. El entorno de desarrollo se divide, básicamente, en tres partes. Una serie de ventanas, que pueden estar visibles u ocultas, constituyen la base de C++ Builder.

2.2. Momentos invariantes de Hu

El momento de orden $(p+q)$ de una imagen digital $f(x,y)$ de tamaño $M \times N$ está definido como

$$\mu_{pq} = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} (x - \bar{x})^p (y - \bar{y})^q f(x, y) \quad (2.1)$$

Donde $p = 0,1,2,\dots$ y $q = 0,1,2,\dots$ siendo enteros, el momento central correspondiente de orden $(p+q)$ está definido como

$$\mu_{pq} = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} (x - \bar{x})^p (y - \bar{y})^q f(x, y) \quad (2.2)$$

Para $p = 0,1,2,\dots$ y $q = 0,1,2,3$ donde

$$\bar{x} = \frac{m_{10}}{m_{00}} \quad (2.3)$$

$$\bar{y} = \frac{m_{01}}{m_{00}} \quad (2.4)$$

Los momentos centrales normalizados denotados por η_{pq} están definidos como

$$\eta_{pq} = \frac{\mu_{pq}}{\mu_{00}^{\gamma}} \quad (2.5)$$

Donde

$$\bar{\gamma} = \frac{p+q}{2} + 1 \quad (2.6)$$

Para $p+q = 2,3,\dots$

2.2.1. Momentos de segundo orden (p+q=2)

Con los momentos de orden 2 comienza realmente el análisis y reconocimiento de formas; dichos momentos contienen términos en los que el valor de la función, esto es, la densidad del objeto, se multiplica por distancias al cuadrado desde el centro de masas. Tales expresiones se conocen como inercia en la mecánica.

a) Momento de inercia de fila

$$U_{20} = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} (x - \bar{x})^2 (y - \bar{y})^0 f(x, y) \quad (2.7)$$

Este momento aumenta cuanto mayor sea la componente horizontal de la figura. Así, será muy grande para una recta horizontal y muy pequeño para una recta vertical.

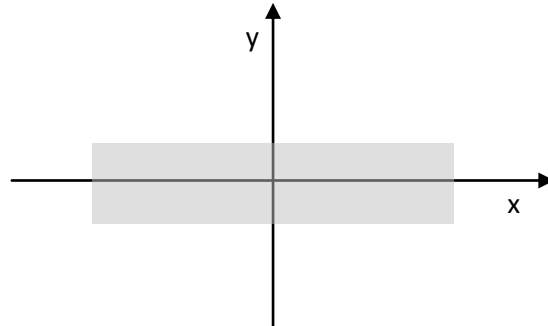


Figura 2.1 Representación gráfica del momento de inercia de fila

b) Momento de inercia de columna

$$U_{02} = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} (x - \bar{x})^0 (y - \bar{y})^2 f(x, y) \quad (2.8)$$

Este momento aumenta cuanto mayor sea la componente vertical de la figura. Así, será muy grande para una recta vertical y muy pequeño para una recta horizontal.

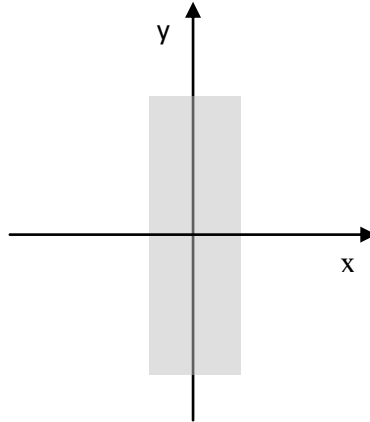


Figura 2.2 Representación gráfica del momento de inercia de columna

c) Momento de inercia cruzado

$$U_{02} = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} (x - \bar{x})^1 (y - \bar{y})^1 f(x, y) \quad (2.9)$$

Este momento tiene en cuenta conjuntamente tanto la componente horizontal como la vertical. Será positivo cuando la componente vertical se concentre en los cuadrantes 1º y 3º. Será negativo cuando la componente vertical se concentre en los cuadrantes 2º y 4º. Una figura simétrica respecto a los dos ejes tiene un momento igual a 0.

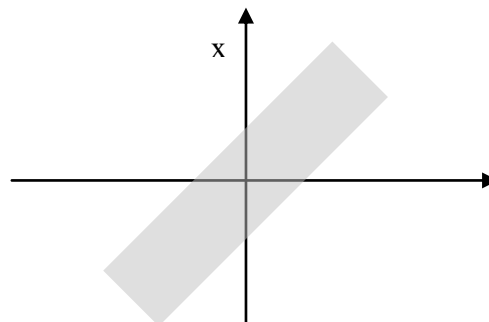


Figura 2.3 Representación gráfica del momento de inercia cruzado

2.2.2. Rotación

$$\varphi = \frac{1}{2} \tan^{-1} \left(\frac{2U_{11}}{U_{20} - U_{02}} \right) \quad (2.10)$$

La orientación o ángulo de rotación de la figura se define como el ángulo entre el eje de abscisas y el eje alrededor del cual la figura puede rotar con mínima inercia

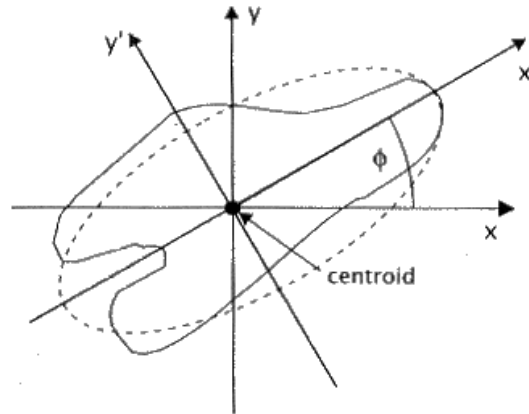


Figura 2.4 Representación gráfica del ángulo de rotación

2.2.3. Momentos de tercer orden (p+q=3)

Los momentos de orden 3 sirven para calcular los momentos invariantes

$$U_{30} = M_{30} - 3\bar{x}M_{20} + 2\bar{x}^2M_{10} \quad (2.11)$$

$$U_{21} = M_{21} - 2\bar{x}M_{11} - \bar{y}M_{20} + 2\bar{x}\bar{y}M_{01} \quad (2.12)$$

$$U_{12} = M_{12} - 2\bar{y}M_{11} - \bar{x}M_{02} + 2y^{-2}M_{10} \quad (2.13)$$

$$U_{03} = M_{03} - 3\bar{y}M_{02} + 2y^{-2}M_{01} \quad (2.14)$$

Un conjunto de siete momentos invariantes pueden ser derivados de los momentos de segundo y tercer orden.

$$\phi_1 = \eta_{20} + \eta_{02} \quad (2.15)$$

$$\phi_2 = (\eta_{20} + \eta_{02})^2 + 4\eta_{11}^2 \quad (2.16)$$

$$\phi_3 = (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - \eta_{03})^2 \quad (2.17)$$

$$\phi_4 = (\eta_{30} + \eta_{12})^2 + (\eta_{21} + \eta_{03})^2 \quad (2.18)$$

$$\begin{aligned} \phi_5 = & (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] \\ & + (3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] \end{aligned} \quad (2.19)$$

$$\begin{aligned} \phi_6 = & (\eta_{20} - \eta_{02})[(\eta_{03} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] \\ & + 4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{21} + \eta_{03}) \end{aligned} \quad (2.20)$$

$$\begin{aligned} \phi_7 = & (3\eta_{21} - \eta_{03})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} - \eta_{03})^2] \\ & + (3\eta_{12} - \eta_{30})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] \end{aligned} \quad (2.21)$$

Este conjunto de momentos es invariante a la traslación, cambio de escala, reflejo (en un signo menos) y a la rotación (Hu, 1962).

2.3. Filtros pasa bajas

Su objetivo es suavizar la imagen y son útiles cuando se supone que la imagen tiene gran cantidad de ruido que se quiere eliminar. También pueden utilizarse para resaltar la información correspondiente a una determinada escala (tamaño de la matriz de filtrado). Existen varias posibilidades:

- Filtro de la media, asigna al pixel central la media de todos los pixeles incluidos en la ventana. La matriz de filtrado estaría compuesta por unos y el divisor sería el número total de elementos en la matriz.
- Filtro de media ponderada, los elementos de la matriz de filtrado no son todos 1 sino que se da más peso a uno de ellos (generalmente el central) para obtener un resultado más parecido a la imagen original y evitar que aparezca borrosa.
- Filtro de la mediana tiene la ventaja de que el valor final del pixel es un valor real presente en la imagen y no un promedio, de este modo se reduce el efecto borroso que tienen las imágenes que han sufrido un filtro de media. Además el filtro de la mediana es menos sensible a valores (Jaramillo).

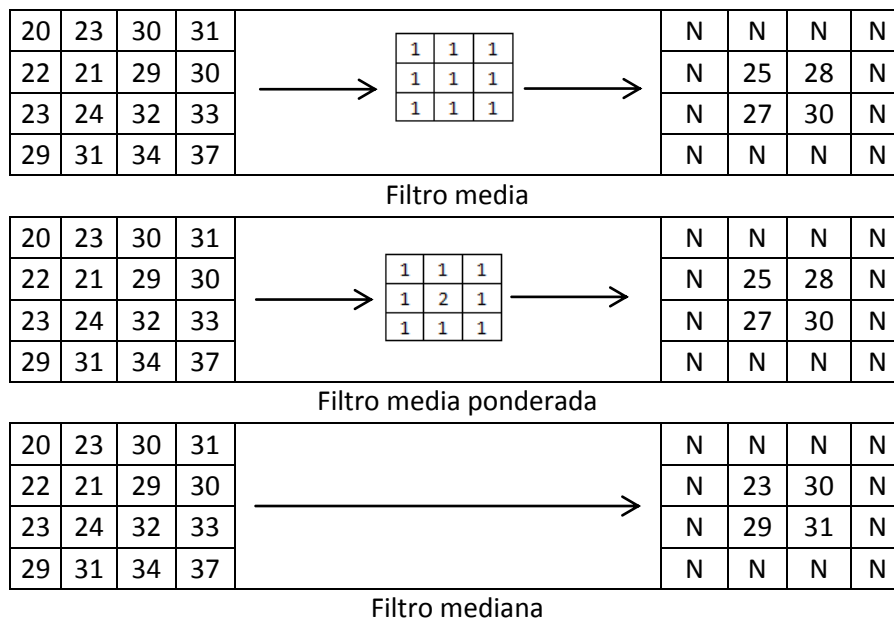


Figura 2.5 Tipos de filtros y formas de aplicación

2.4. Método de OTSU para umbral optimo

La segmentación es una parte importante en el procesamiento y análisis de imágenes, pues simplifica la imagen al dividirla en partes resaltando las regiones de interés para facilitar el proceso de identificación y análisis. La mayoría de los algoritmos de segmentación se basan en una de las dos propiedades básicas de los valores de intensidad: la discontinuidad y similitud (Young, 1986). Los principios que rigen son la similitud entre los píxeles pertenecientes a un objeto y sus diferencias respecto al resto. Por tanto, la escena debe caracterizarse por un fondo uniforme y por objetos parecidos. Al aplicar un umbral, T , la imagen en escala de grises, $f(x,y)$, quedará binarizada; etiquetando con '1' los píxeles correspondientes al objeto y con '0' aquellos que son del fondo.

Por ejemplo, si los objetos son claros respecto del fondo, se aplicará

$$g(x,y) = \begin{cases} 1 & \Leftrightarrow f(x,y) > T \\ 0 & \Leftrightarrow f(x,y) \leq T \end{cases} \quad (2.22)$$

La mayoría de las técnicas de umbralización se basan en estadísticas sobre el histograma unidimensional de una imagen. A medida que el número de clases de una imagen aumenta, el método de Otsu necesita mucho más tiempo para seleccionar un umbral multinivel adecuado. La importancia del método de Otsu radica en que es automático, es decir, no necesita supervisión humana ni información previa de la imagen antes de su procesamiento.

Una imagen es una función bidimensional de la intensidad del nivel de gris, y contiene N píxeles cuyos niveles de gris se encuentran entre 0 y L , para este caso L tomara el valor de 255, debido a que usamos una imagen de 256

colores, el nivel de gris i se denota como f_i , y la cantidad de gris i (probabilidad de ocurrencia) en la imagen está dada por:

$$p_i = \frac{f_i}{N} \quad (2.23)$$

En el caso de la umbralización en dos niveles de una imagen (a veces llamada binarización), los píxeles son divididos en dos clases: C1, con niveles de gris $[1, \dots, t]$; y C2, con niveles de gris $[t+1, \dots, L]$.

$$C_1 = \frac{p_1}{\omega_1(t)}, \dots, \frac{p_t}{\omega_1(t)} \quad (2.24)$$

$$C_2 = \frac{p_{t+1}}{\omega_2(t)}, \frac{p_{t+2}}{\omega_2(t)}, \dots, \frac{p_L}{\omega_2(t)} \quad (2.25)$$

Donde la distribución de la cantidad de gris para las dos clases es:

$$\omega_1(t) = \sum_{i=1}^t p_i \quad (2.26)$$

$$\omega_2(t) = \sum_{i=t+1}^L p_i \quad (2.27)$$

También, la media para la clase C1 y la clase C2 es:

$$\mu_1 = \sum_{i=1}^t \frac{i * p_i}{\omega_1(t)} \quad (2.28)$$

$$\mu_2 = \sum_{i=t+1}^L \frac{i * p_i}{\omega_2(t)} \quad (2.29)$$

Sea μ_T la intensidad media de toda la imagen. Entonces:

$$\omega_1 \cdot \mu_1 + \omega_2 \cdot \mu_2 = \mu_T \quad (2.30)$$

$$\omega_1 + \omega_2 = 1 \quad (2.31)$$

Usando análisis discriminante, Otsu (1979) definió la variancia entre clases de una imagen umbralizada como:

$$\sigma_B^2 = \omega_1 (\mu_1 - \mu_T)^2 + \omega_2 (\mu_2 - \mu_T)^2 \quad (2.32)$$

Para una umbralización de dos niveles, Otsu verificó que el umbral óptimo t^* se elige de manera que σ_B^2 sea máxima; esto es

$$t^* = \text{MAX}_t \{ \sigma_B^2(t) \}, \quad i \leq t \leq L \quad (2.33)$$

2.5. Características morfológicas

2.5.1. Área foliar

Astegiano et al (2002), realizó un estudio sobre el área foliar de una planta y este se refiere a la cantidad de superficie de hoja que ella posee, el cual es un parámetro ampliamente usado en estudios de ecofisiología de cultivos. Para su determinación se utilizan diferentes métodos ya sean destructivos, no destructivos o indirectos

2.5.2. Circularidad

La circularidad es una medida compacta de una forma, un círculo es la forma más compacta, así que cuanto más compacta es una forma, más se

asemeja a un círculo. La circularidad es una relación y por lo tanto un número sin dimensiones. La relación de circularidad C será mayor que 0 y menor o igual a 1. Una forma larga y delgada tiene una circularidad que se aproxima a 0 y un círculo tiene una circularidad de 1.

2.5.3. Rectangularidad

La medida de la rectangularidad es un descriptor de forma, aunque es un método intuitivo, ha sido muy poco mencionado y usado, la rectangularidad está definida como la media entre el área de la región y el área del mínimo rectángulo que limita a la figura, y está dada por la siguiente ecuación:

$$\text{Rectangularidad} = \frac{P}{L * A} \quad (2.34)$$

2.5.4. Relación perímetro-longitud

$$R_{PL} = \frac{P}{L} \quad (2.35)$$

2.5.5. Relación perímetro-ancho y longitud

$$R_{PLA} = \frac{P}{L + A} \quad (2.36)$$

2.5.6. Factor de elongación

$$R_{LA} = \frac{L}{A} \quad (2.37)$$

2.6. Erosión

La erosión es la operación dual de la dilatación: la erosión de una imagen A por un elemento estructurante B es el conjunto de elementos para los que la traslación de B está contenida en A (Soile, 2003).

$$A \ominus B = \{x \in E^N \mid B_x \subseteq A\} \quad (2.38)$$

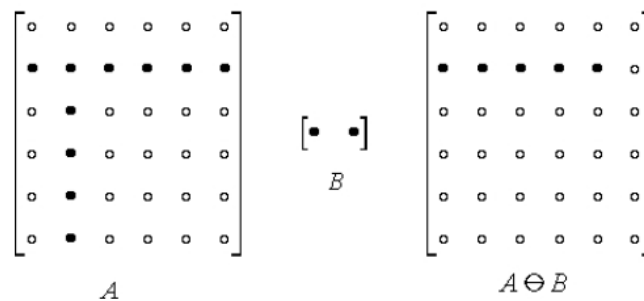


Figura 2.6 Erosionado de A con elemento estructurante B

La erosión puede verse como la intersección de las traslaciones negativas. Si B trasladado a x puede estar contenido en A (poniendo el origen de B en x), entonces x pertenece a la erosión

$$A \ominus B = \bigcap_{b \in B} A_{-b} \quad (2.39)$$

El efecto de la erosión es la disminución de la talla de las partículas y la desaparición de los elementos de talla inferior al elemento estructurante:

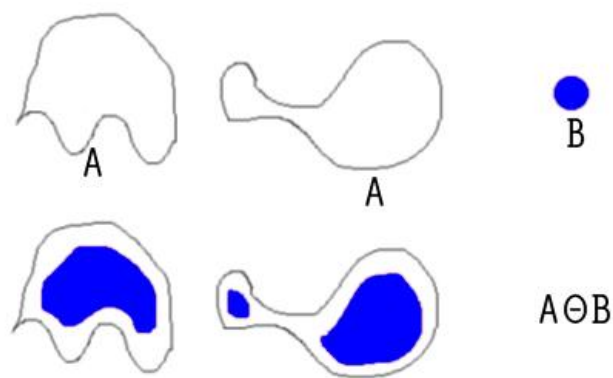


Figura 2.7 Efecto de la erosión con elemento estructurante B

Propiedades

La erosión se puede considerar como un “encogimiento” de la imagen original. En términos de la teoría de conjuntos esto significa que el conjunto erosionado está contenido dentro del conjunto original, lo que implica que una transformación de este tipo sea Antiextensiva.

$$X \supset X \ominus B \quad (2.40)$$

3. REDES NEURONALES ARTIFICIALES

En los años 50 dos neurocientíficos, Warren McCulloch y Walter Pitts (1943) propusieron un modelo básico de neurona. Aunque McCulloch y Pitts no fueron los primeros en considerar las neuronas como sistemas para realizar cálculos, fueron los pioneros en el intento de definir formalmente a las neuronas como elementos computacionales y explorar las consecuencias de las propiedades neuronales.

McCulloch y Pitts parten de cinco consideraciones acerca del comportamiento de las neuronas.

Dichas consideraciones las plantean de la siguiente forma:

- La actividad neuronal es un proceso "todo o nada".
- Un cierto número fijo de sinapsis debe ser excitado dentro de un período de adición latente en orden de excitar una neurona en cualquier intervalo de tiempo, y este número es independiente de la actividad previa y la posición de la neurona.
- El único retardo significativo dentro del sistema es el retardo sináptico.

- La actividad de cualquier sinapsis inhibitoria previene absolutamente la excitación de la neurona en ese intervalo de tiempo.
- La estructura de la red no cambia con el tiempo.

Estas consideraciones describen a lo que se ha conocido como la neurona "McCulloch-Pitts". De las consideraciones mostradas podemos extraer y resaltar lo siguiente. La neurona McCulloch-Pitts es un dispositivo binario, es decir, solo puede estar en uno de dos posibles estados. Cada neurona puede recibir entradas de sinapsis excitadoras, las cuales tienen todos unos mismos pesos. También pueden recibir entradas de sinapsis inhibitorias, cuya acción es absoluta; es decir, si la sinapsis inhibitoria está activa, la neurona no puede encender. Hay un lapso de tiempo dado fijo para la integración de las entradas sinápticas, basado en el retardo sináptico observado; esto le da a la neurona su carácter de trabajo en tiempo discreto (McCulloch y Pitts 1943) .

Al hacer estas consideraciones, los autores comienzan trabajando con arreglos lineales de neuronas, es decir, consideran que no hay vías de retroalimentación entre neuronas, y parten de una serie de simplificaciones en las que consideran que el umbral de disparo en las neuronas adopta valores discretos, y que estos umbrales también se mantienen inalterados. Estas últimas consideraciones les permiten demostrar el funcionamiento de su neurona, para posteriormente demostrar que mediante la realización de diversos arreglos y conexiones de neuronas de este tipo, entre los que se encuentran algunos arreglos cíclicos de neuronas, es posible producir los mismos resultados que con neuronas con umbrales cambiantes con el tiempo.

El modo de operación de la neurona McCulloch-Pitts es simple. Durante el tiempo de integración, la neurona responde a la actividad de su sinapsis, la cual refleja el estado de las células pre sinápticas. Si no hay sinapsis inhibitorias activas, la neurona suma sus entradas sinápticas y verifica si la suma de ellas alcanza o excede su nivel de umbral. Si lo hace, entonces la neurona se vuelve activa. Si no, la neurona permanece inactiva. En caso de que exista alguna sinapsis inhibitoria activa, la neurona permanece inactiva. Para ilustrar la funcionalidad del modelo McCulloch-Pitts vamos a revisar el comportamiento de algunas neuronas. Para ilustrar esto vamos a utilizar la representación gráfica que utilizaron sus autores en el artículo que describe al modelo, el cual se muestra en la fig. 3.1.

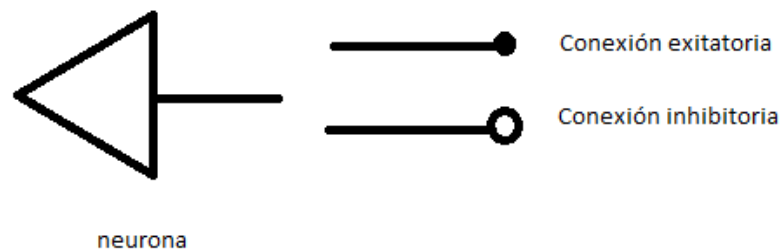


Figura 3.1 Representación gráfica de una neurona y sus tipos de conexiones

Un ejemplo de su funcionamiento lo podemos tener si consideramos una neurona de este tipo con dos sinapsis excitatorias a su entrada, considerando unitario el peso de cada una de estas sinapsis, y supongamos que la neurona tiene un umbral de dos unidades. Si ninguna de las dos entradas se encuentra activa en un tiempo dado, o incluso si solamente una de ellas se encuentra activa y la otra no, la neurona no responderá; solamente cuando ambas sinapsis de entrada se encuentren activas la neurona podrá

responder. Si revisamos cuidadosamente este comportamiento podremos identificar que corresponde al comportamiento de la función lógica AND. De la misma manera, si consideramos que la neurona ahora posee un umbral unitario, entonces ahora tendremos el comportamiento lógico OR.

3.1. Elementos y de una RNA

En las Redes Neuronales Artificiales, RNA, la unidad análoga a la neurona biológica es el elemento procesador, PE (process element). Un elemento procesador tiene varias entradas y las combina, normalmente con una suma básica. La suma de las entradas es modificada por una función de transferencia y el valor de la salida de esta función de transferencia se pasa directamente a la salida del elemento procesador (Olabe, 2008).

Los elementos básicos de un sistema neuronal biológico son las neuronas, que se agrupan en conjuntos compuestos por millones de ellas organizadas en capas, constituyendo un sistema con funcionalidad propia. Un conjunto de estos subsistemas da lugar a un sistema global. En la figura 3.2 podemos ver una RNA como una colección de procesadores paralelos conectados entre ellos en forma de un grafo dirigido, organizados de tal manera que la estructura de la red nos lleva por sí misma a considerarla como una característica a tener en cuenta a la hora de crear una RNA.

Podemos representar de manera sistemática cada elemento de proceso (unidad) de la red como un nodo, con conexiones entre unidades representadas mediante arcos, además estos arcos nos indican la dirección en la que fluye la información.

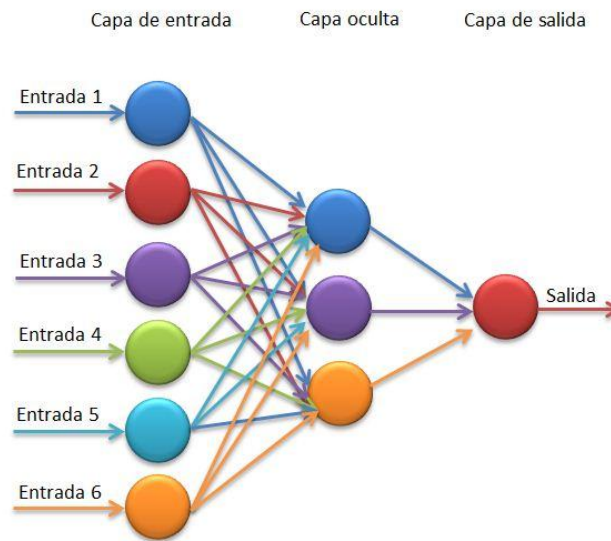


Figura 3.2 Ejemplo de Red Neuronal Artificial (perceptrón múltiple)

De manera formal Rumelhart definió los componentes de un sistema neuronal:

1. Un conjunto de procesadores elementales o neuronas
2. Un patrón de conectividad entre las neuronas o arquitectura
3. Una dinámica de activaciones.
4. Una regla de aprendizaje
5. El entorno donde opera

3.2. Diseño de una RNA

Con un paradigma convencional de programación en ingeniería del software, el objetivo del programador es modelar matemáticamente (con distintos grados de formalismo) el problema en cuestión y posteriormente

formular una solución (programa) mediante un algoritmo codificado que tenga una serie de propiedades que permitan resolver dicho problema. El objetivo es conseguir que la red aprenda automáticamente las propiedades deseadas, en este sentido, el diseño de la red tiene más que ver con cuestiones tales como la selección del modelo de red, la de las variables a incorporar y el preprocesamiento de la información que formará el conjunto de entrenamiento. Asimismo, el proceso por el que los parámetros de la red se adecuan a la resolución de cada problema no se denomina genéricamente programación sino que se suele denominar entrenamiento neuronal (Russel, 2004).

La selección de una red se realiza en función de las características del problema a resolver. La mayoría de éstos se pueden clasificar en aplicaciones de Predicción, Clasificación, Asociación, Conceptualización, Filtrado y Optimización. Los tres primeros tipos de aplicaciones requieren un entrenamiento supervisado (Olabe, 2008).

RED	AÑO	TIPO
Adaline y Madaline Diseñador: Bernard Widrow Características: Técnicas de Adaptación para el Reconocimiento de Patrones.	1960	Predicción
Adaptive Resonance Theory Networks (ART) Diseñador: Carpenter, Grossberg Características: Reconocimiento de Patrones y Modelo del Sistema Neuronal. Concepto de Resonancia Adaptativa.	1960-86	Conceptualización
Back-Propagation Diseñador: Rumelhart y Parker Características: Solución a las limitaciones de su red predecesora el perceptron.	1985	Clasificación
Bi-Directional Associative Memory (BAM) Networks		

Diseñador: Bart Kosko	1987	Asociación
Características: Inspirada en la red ART.		
The Boltzmann Machine		
Diseñador: Ackley, Hinton y Sejnowski	1985	Asociación
Características: Similar a la red Hopfield.		
Brain-State-in a Box		
Diseñador: James Anderson	1970-86	Asociación
Características: Red Asociativa Lineal.		
Cascade-Correlation-Networks		
Diseñador: Fahhman y Lebiere	1990	Asociación
Características: Adición de nuevas capas ocultas en cascada.		
Counter-Propagation		
Diseñador: Hecht-Nielsen	1987	Clasificación
Características: Clasificación Adaptativa de Patrones.		
Delta-Bar-Delta (DBD) Networks		
Diseñador: Jacobb	1988	Clasificación
Características: Métodos Heurísticos para Acelerar la Convergencia.		
Digital Neural Network Architecture (DNNA) Networks		
Diseñador: Neural Semiconductor Inc.	1990	Predicción
Características: Implementación Hardware de la función Sigmoid.		
Directed Random Search (DRS) Networks		
Diseñador: Maytas y Solis	1965-81	Clasificación
Características: Técnica de valores Random en el mecanismo de Ajuste de Pesos.		
Functional-link Networks (FLN)		
Diseñador: Pao	1989	Clasificación
Características: Versión mejorada de la red Backpropagation.		
Hamming Networks		
Diseñador: Lippman	1987	Asociación
Características: Clasificador de vectores binarios utilizando la Distancia Hamming.		

Hopfield Networks		
Diseñador: Hopfield	1982	Optimización
Características: Concepto de la red en términos de energía.		
Learning Vector Quantization (LVQ) Networks		
Diseñador: Kohonen	1988	Clasificación
Características: Red Clasificadora.		
Perceptron Networks		
Diseñador: Rosenblatt	1950	Predicción
Características: Primer modelo de sistema Neuronal Artificial.		
Probabilistic Neural Network (PNN)		
Diseñador: Spetcht	1988	Asociación
Características: Clasificación de Patrones utilizando métodos estadísticos.		
Recirculation Networks		
Diseñador: Hinton y McClelland	1988	Filtrado
Características: Alternativa a la red Backpropagation.		
Self-Organizing Maps (SOM)		
Diseñador: Kohonen	1979-82	Conceptualización
Características: Aprendizaje sin supervisión.		
Spatio-Temporal-Pattern Recognition (SPR)		
Diseñador: Grossberg	1960-70	Asociación
Características: Red clasificadora Invariante en el espacio y tiempo.		

Tabla 3.1 Tipos de redes neuronales

3.3. Estructura de la RNA

Biológicamente, un cerebro aprende mediante la reorganización de las conexiones sinápticas entre las neuronas que lo componen. De la misma manera, las RNA tienen un gran número de procesadores virtuales interconectados que de forma simplificada simulan la funcionalidad de las neuronas biológicas. En esta simulación, la reorganización de las conexiones sinápticas biológicas se modela mediante un mecanismo de pesos, que son ajustados durante la fase de aprendizaje (Haykin, 2001). En una RNA entrenada, el conjunto de los pesos determina el conocimiento de esa RNA y tiene la propiedad de resolver el problema para el que la RNA ha sido entrenada.

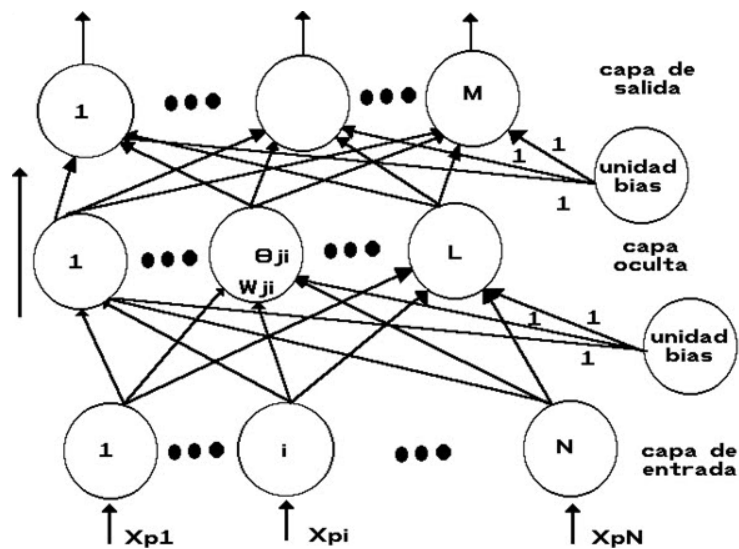


Figura 3.3 Ejemplo de formas de conexión entre neuronas (Freeman 1991)

3.4. Estructura de una Neurona Artificial

La arquitectura del Perceptron, llamada mapeo de patrones (pattern-mapping), aprende a clasificar modelos mediante un aprendizaje supervisado. Los modelos que clasifica suelen ser generalmente vectores con valores

binarios (0,1) y las categorías de la clasificación se expresan mediante vectores binarios.

El Perceptron presenta dos capas de unidades procesadoras (PE) y sólo una de ellas presenta la capacidad de adaptar o modificar los pesos de las conexiones. La arquitectura del Perceptron admite capas adicionales pero éstas no disponen la capacidad de modificar sus propias conexiones (Freeman, 1991).

La Fig. 3.4 muestra la unidad procesadora básica del Perceptron. Las entradas x_i llegan por la parte izquierda, y cada conexión con la neurona j tiene asignada un peso de valor w_{ij} .

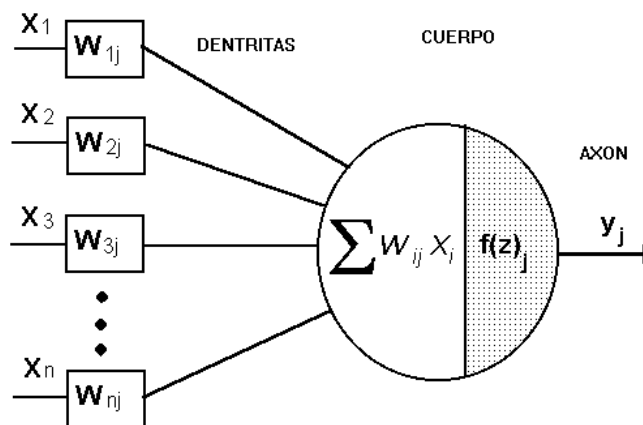


Figura 3.4 Elementos básicos de una neurona artificial

Ahora describiremos la mayor parte de componentes de los que está compuesta una red neuronal. Estos componentes son válidos aun cuando la neurona sea usada en la entrada, salida, o capa oculta.

Una red neuronal se compone de unidades llamadas neuronas. Cada neurona recibe una serie de entradas a través de interconexiones y emite una salida. Esta salida viene dada por tres funciones:

Una función de propagación (también conocida como función de excitación), que por lo general consiste en el sumatorio de cada entrada multiplicada por el peso de su interconexión (valor neto). Si el peso es positivo, la conexión se denomina excitatoria; si es negativo, se denomina inhibitoria.

$$f_{pi}(t) = \sigma(t)(w_{ij}, x_i) \quad (3.1)$$

La función más habitual es la suma ponderada de todas las entradas. Podemos agrupar las entradas y pesos en dos vectores (x_1, x_2, \dots, x_n) y ($w_{1j}, w_{2j}, \dots, w_{nj}$), podemos calcular esta suma realizando el producto escalar sobre estos dos vectores.

$$f_{pi}(t) = \sum w_{ij} \cdot x_i \quad (3.2)$$

Una función de activación, que modifica a la anterior. Puede no existir, siendo en este caso la salida la misma función de propagación.

$$f_T(t) = h_i(a_i(t-1), f_{pi}(t)) \quad (3.3)$$

En este caso la función de activación depende del potencial postsináptico $f_{pi}(t)$ y del propio estado de activación de anterior. Sin embargo, en muchos modelos de A se considera que el estado actual de la neurona no depende de su estado anterior $a_i(t - 1)$, sino únicamente del actual

$$f_T(t) = h_i(f_{pi}(t)) \quad (3.4)$$

Una función de transferencia, que se aplica al valor devuelto por la función de activación. Se utiliza para acotar la salida de la neurona y generalmente viene dada por la interpretación que queremos darle a dichas salidas. Algunas de las más utilizadas son la función sigmoidea (para obtener valores en el intervalo $[0,1]$) y la tangente hiperbólica (para obtener valores en el intervalo $[-1,1]$).

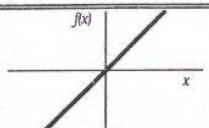
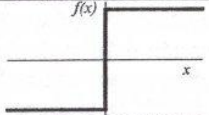
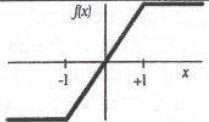
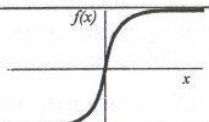
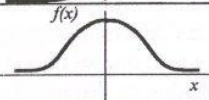
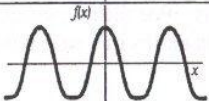
	Función	Rango	Gráfica
Identidad	$y = x$	$[-\infty, +\infty]$	
Escalón	$y = \text{sign}(x)$ $y = H(x)$	$\{-1, +1\}$ $\{0, +1\}$	
Lineal a tramos	$y = \begin{cases} -1, & \text{si } x < -l \\ x, & \text{si } -l \leq x \leq l \\ +1, & \text{si } x > l \end{cases}$	$[-1, +1]$	
Sigmoidea	$y = \frac{1}{1 + e^{-x}}$ $y = \text{tgh}(x)$	$[0, +1]$ $[-1, +1]$	
Gaussiana	$y = Ae^{-Bx^2}$	$[0, +1]$	
Sinusoidal	$y = A \text{sen}(ax + \varphi)$	$[-1, +1]$	

Figura 3.5 Ejemplos de funciones de activación (Freeman 1991)

3.5. Clasificación de las Redes Neuronales Artificiales

Una primera clasificación de las redes de neuronas artificiales que se suele hacer es en función del patrón de conexiones que presenta. Así se definen tres tipos básicos de redes:

* Dos tipos de redes de propagación hacia delante o acíclicas en las que todas las señales van desde la capa de entrada hacia la salida sin existir ciclos, ni conexiones entre neuronas de la misma capa.

- Monocapa. Ejemplos: perceptrón, Adaline.
- Multicapa. Ejemplos: perceptrón multicapa.

* Las redes recurrentes que presentan al menos un ciclo cerrado de activación neuronal. Ejemplos: Elman, Hopfield, máquina de Boltzmann.

3.5.1. Adeline-Madeline

La arquitectura de Adaline (Adaptive Linear Neuron) fue creada por Bernard Widrow en 1959. Utiliza un dispositivo lógico que realiza una suma lineal de las entradas y genera una función umbral para el resultado de dicha suma. La arquitectura Madaline (Multilayer Adaline) creada también por Widrow presenta una configuración constituida por dos o más unidades Adaline. A lo largo del tiempo se han estudiado diferentes variaciones de los algoritmos de aprendizaje de la Adaline, y Madaline, y entre las aplicaciones investigadas destacan entre otras, filtros adaptativos de eliminación de ruido y reconocimiento de patrones de señales. No obstante, desde los primeros experimentos con la Adaline y Madaline se constató la capacidad de clasificar patrones linealmente separables, presentando la misma limitación que el

Perceptron: la carencia de un método que ajuste más de una capa de pesos (Neelakanta, 1994).

3.5.2. Estructura Adaline.

La fig. 3.6 a, muestra una Adaline básica. La unidad procesadora representada por un círculo con el símbolo sumatorio implementa una función umbral. Las conexiones de cada una de las entradas tienen asociadas un valor de ponderación llamado también peso w_i . El mecanismo de ajuste de los pesos representado en la fig. 3.6 b, consiste en utilizar la diferencia entre el valor de la salida y el valor esperado.

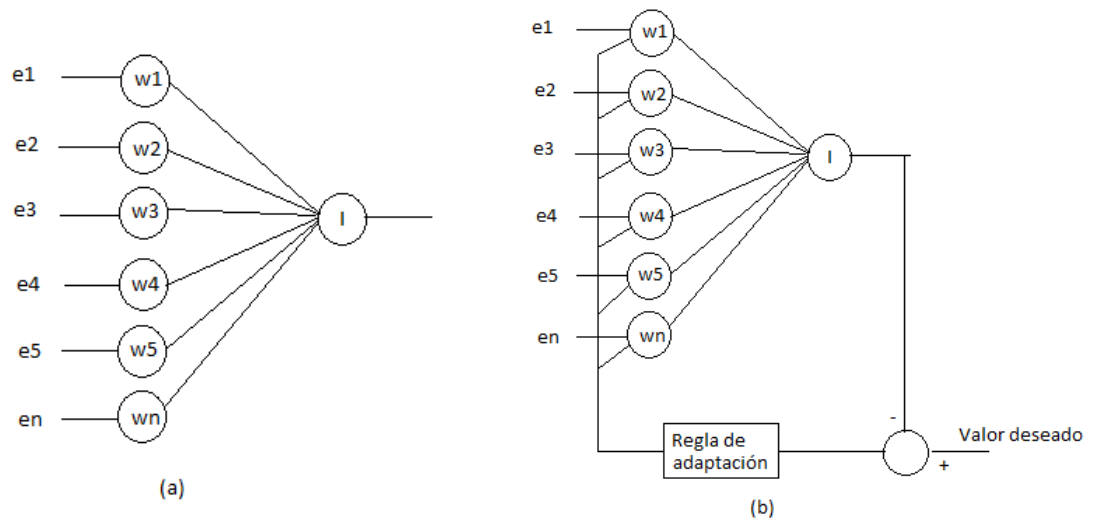


Figura 3.6 Red Adaline

La unidad procesadora actúa como un sumador y después realiza la función umbral según la ecuación

$$x_j = \begin{cases} 1 & \text{si } \sum_i w_i * a_i \geq 0 \\ -1 & \text{si } \sum_i w_i * a_i < 0 \end{cases} \quad (3.5)$$

La salida de la unidad Adaline es ± 1 a diferencia de la arquitectura del Perceptron que sólo permite los valores 0 y 1.

3.5.3. Estructura Madaline.

El sistema Madaline tiene una capa de unidades Adaline que están conectadas a una simple unidad Madaline. La Fig. 1.5.1.2 muestra cuatro unidades en la capa de entrada, tres unidades Adaline en la segunda capa y una unidad Madaline en la tercera capa.

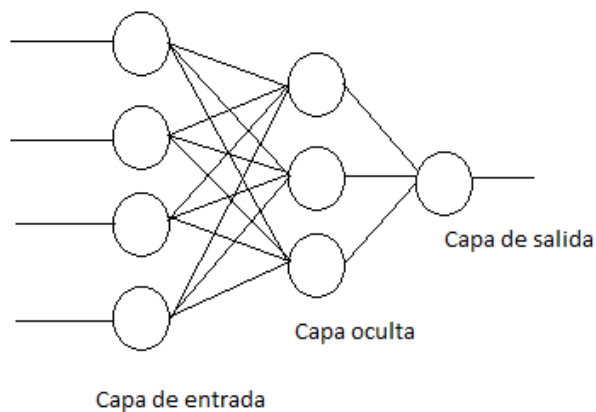


Figura 3.7 Sistema Madaline

Las conexiones entre la capa de entrada y la capa de las unidades Adaline tienen asociadas un peso ajustable por cada una de ellas. Sin embargo, las conexiones entre la capa de Adaline y la unidad Madaline no tienen asociado ningún peso. Cada unidad Adaline transmite su salida (-1 ó +1) a la unidad Madaline. La Madaline emplea una regla de mayorías para obtener su

salida: si la mitad o más de las unidades Adaline presentan un valor de salida +1, entonces la salida de la Madaline es +1. En caso contrario el valor de salida de la red Madaline es -1.

3.5.4. Red de propagación hacia adelante (feedforward)

Esta fase de propagación hacia adelante se inicia cuando se presentan las entradas en la capa de la red, las unidades de entrada toman el valor correspondiente y se calcula el valor de activación o nivel de salida de la primera capa. A continuación las demás capas realizarán la fase de propagación hacia adelante que determina el nivel de activación de las otras capas. La función f es una función umbral genérica, entre las que cabe destacar la función Sigmoid y la función Hiperbólica.

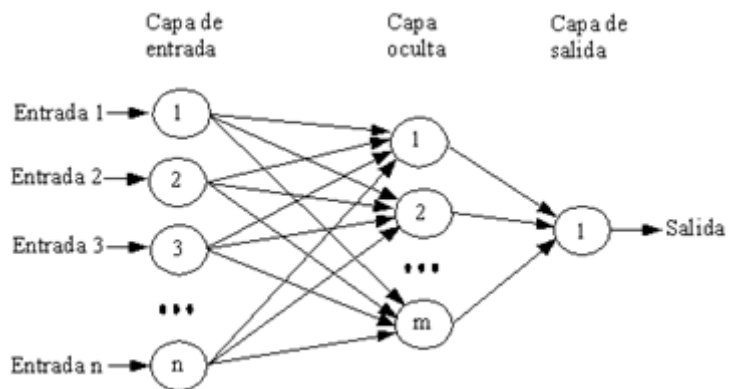


Figura 3.8 Perceptrón múltiple con propagación hacia adelante

3.5.5. Red de propagación hacia atrás (Backpropagation)

El algoritmo que se emplea para entrenar una RNA es la Regla Delta Generalizada. La RNA está compuesta por capas múltiples, con propagación de las entradas hacia adelante y completamente interconectada en sus capas. Esto significa que no existen conexiones hacia atrás ni conexiones que alimenten la salida de una capa hacia otra que no sea la siguiente. Puede existir más de una capa oculta en una RNA.

Una RNA es llamada red representativa (mapping network) si esta es capaz de calcular alguna función de relación entre las entradas y las salidas. Estas son útiles en situaciones donde se desconoce la función de relación entre las entradas y salidas, en estos casos la alta capacidad de la RNA para descubrir su propia representación es extremadamente útil. Si se cuenta con un conjunto de P vectores de parejas, $(x_1, u_1), \dots, (x_n, u_n)$, los cuales son ejemplo de una función representativa $u = \varphi(x)$: $x \in \mathbb{R}^N$, $u \in \mathbb{R}^N$. Se puede entrenar una RNA para obtener una aproximación $O = u' = \varphi(x)'$. Para emplear el método aquí expuesto es necesario que los vectores de parejas hayan sido adecuadamente seleccionados y que exista suficiente cantidad de ellos. El algoritmo asemeja al problema de encontrar la ecuación de una línea que mejor aproxima un conjunto de puntos. Como este caso se asemeja a contar con no linealidades y dimensiones múltiples, se emplea una versión interactiva del método simple de los mínimos cuadrados, llamada técnica de gradiente descendente. Para iniciar se considera un vector de entradas, $x_i = (x_{i1}, x_{i2}, \dots, x_{in})^T$, que se aplica a la capa de entrada de una RNA. Las unidades de la entrada distribuyen los valores a las unidades de las capas ocultas. La entrada a la red de la unidad oculta j es:

$$Net_{pj}^k = \sum_{i=1}^N w_{jp}^k * x_{pj} + \theta_j^k \quad (3.6)$$

3.6. Aprendizaje

Una segunda clasificación que se suele hacer es en función del tipo de aprendizaje de que es capaz (si necesita o no un conjunto de entrenamiento supervisado). Para cada tipo de aprendizaje encontramos varios modelos propuestos por diferentes autores:

- * Aprendizaje supervisado: necesitan un conjunto de datos de entrada previamente clasificado o cuya respuesta objetivo se conoce.

- * Aprendizaje no supervisado o auto organizado: no necesitan de tal conjunto previo.

- * Redes híbridas: son un enfoque mixto en el que se utiliza una función de mejora para facilitar la convergencia.

- * Aprendizaje reforzado: se sitúa a medio camino entre el supervisado y el auto organizado.

4. IDENTIFICACIÓN DE ESPECIES

Como se ha dicho en la hipótesis se desarrollará un programa por el cual sea fácil identificar ciertas especies de hojas y que pueda ser actualizable para el entrenamiento de nuevas especies. Principalmente se trabajó con 13 especies, de las cuales algunas comparten características similares (en cuanto a la forma y/o contorno), empleando una red neuronal, se pretende que dicha red entregue a la salida un único valor que corresponda al conjunto de una sola especie; así, se obtiene como resultado un valor neural para cada especie, para que de este modo pueda realizarse una clasificación adecuada. Cabe destacar que las muestras obtenidas de las hojas, deben encontrarse en el mejor estado posible, es decir que la hoja no presente enfermedades que deformen el contorno de la hoja, que no se encuentre doblada, arrugada o maltratada a la hora de capturar la imagen de la hoja, todo esto con el fin de minimizar el error en el procesamiento que se le aplicara a cada especie. Como se ha mencionado anteriormente se trabajará con 13 especies distintas, de las cuales se identificó el nombre de 7 de ellas: Durazno, Lima, Guayaba, Malva, Melissa, Melia azedarach, Hedera, El resto de las especies (6) no se pudo conocer o comprobar su nombre; a estas especies se les asigno un nombre para identificación del programa: Alargada, Corazón, Fpica, Estrella, Ovalada y Flama. De este tipo de hojas, se tomaron 50 muestras de las primeras 4 especies mencionadas anteriormente y 10 muestras por especie de las restantes.

4.1. Identificación y planteamiento del problema

Actualmente hay muchos procesos en los cuales sistemas computarizados intervienen de manera efectiva, ya sea para la identificación de

piezas, reconocimiento de objetos, (clasificados por su forma, colores, diseños, etc.), seguimiento de objetos, identificación de rostros, etc. A pesar de estos avances tecnológicos, las maquinas que controlan este tipo de procesos aún carecen de la habilidad del ser humano para reconocer múltiples características y detalles (formas, tamaños, colores, tramas, grietas, estructuras de una forma, tipos de contorno, etc.), y asociarlas para asignarles una identificación. De ahí, el principal problema a la hora de desarrollar este proyecto, puesto que para una maquina presenta un problema complejo la identificación y distinción de estas características.

En particular hay varias formas o propuestas para resolver este problema; basándonos en modelos matemáticos como los momentos invariantes de Hu (1962), estos presentan datos importantes que son invariantes al tamaño y rotación de las hojas; de esta manera se obtendrían valores similares para una misma especie y distintivos en comparación con otras especies. Sin embargo dichos momentos son insuficientes para la correcta clasificación, por lo que también se ha basado este trabajo en la medición de varios parámetros que nos proporcionen alguna información de la especie, como tal la excentricidad, la circularidad, la rectangularidad, la relación entre perímetro y longitud, la relación entre el largo y el ancho, la relación entre el perímetro y la longitud en adición con el ancho, etc.

Cada uno de estos parámetros propuestos nos proporciona un dato de ayuda, como es el caso de la excentricidad de la hoja, la cual es un parámetro que determina el grado de desviación de una sección cónica (elipse) con respecto a una circunferencia; en otras palabras, la excentricidad nos dará un valor 0 para un objeto circular y conforme ese objeto se vuelva menos redondo

y más ovalado el valor se irá incrementado, hasta que llegue a ser plano y la excentricidad tendrá un valor de 1. De acuerdo a lo anterior, la excentricidad nos ayudara a identificar el grado de redondez de la hoja; cabe mencionar que muchas de las hojas -o las más comunes- tienden a presentar una forma ovalada. Dentro de la base de datos, se cuenta con cuatro especies con forma ovalada, y dos especies que se acercan más a la forma redonda (una más distintiva que otra). Por tanto, se requieren de más herramientas para distinguir entre ellas; debido a lo anterior, se integra un parámetro más al análisis: la rectangularidad, la cual nos proporciona la similitud entre un rectángulo y una hoja. Por otro lado, se tiene que ocho de las especies tienden a ser ovaladas y su rectangularidad puede caer dentro de un rango similar, por lo que se propone también incluir la relación de perímetro y longitud, dicha relación nos proporciona un factor que nos indica cuantas veces es proporcional la longitud física al perímetro de la hoja. Sumando lo anterior también se medirá cuantas veces es mayor el perímetro en relación a la suma del ancho y largo de la hoja, y como una proporción adicional la relación entre la longitud y ancho físicos de la hoja; cada uno de estos parámetros proporcionará de forma independiente un dato significativo que nos permita acercarnos más a la identificación de las especies. Sin embargo, no podemos descartar que en cierto porcentaje de los casos, estos valores presenten gran similitud; para ello, primeramente debemos de cerciorarnos qué relaciones son adecuadas y de qué forma deberán ser usadas.

Todos estos valores, factores o relaciones obtenidas, por sí solas carecen de significado alguno; por ello se propone el uso de una Red Neuronal Artificial (Artificial Neural System), debido a su gran parecido con las neuronas del ser humano, que pueden conmutar, manipular, modificar, procesar, grandes cantidades de información en fracciones de segundos; la RNA se plantea como

un método que permita que dichos valores se relacionen proporcionando significado a ello, proponiendo así el diseño de un programa que sea capaz de distinguir entre cada especie basándose principalmente en los modelos matemáticos de los parámetros o relaciones mencionadas. Se ha planteado entonces un esquema básico para la resolución a este problema, que se resume con el siguiente diagrama:

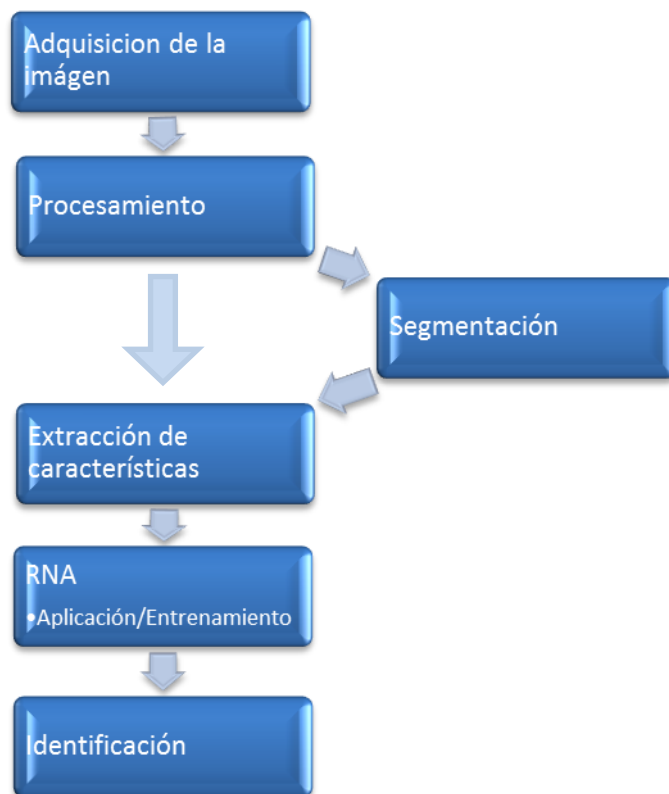


Figura 4.1 Diagrama de bloques a la solución del problema propuesto

4.2. ESTRUCTURA DEL PROGRAMA

4.2.1. Interfaz del programa

Para facilitar el manejo del este programa la interfaz propuesta ha sido simplificada; esta interfaz contiene una barra de menú sencilla (con los siguientes menús: Archivo, Filtrar, Identificar, Red Neuronal), un segmento para mostrar la imagen de la especie de hoja que se analiza, un segmento donde nos muestra las características de las hojas y un segmento para la identificación de la especie (ver fig. 4.2).

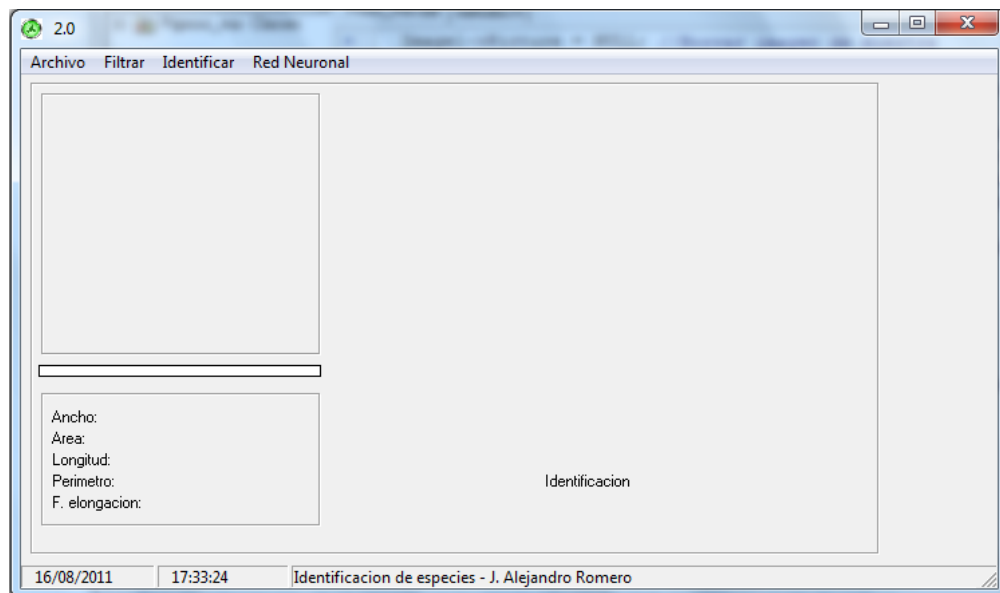


Figura 4.2 Interfaz del programa

En el menú archivo podemos encontrar los submenús Abrir (el cual nos permite abrir una imagen nueva en formato BMP), Guardar (que nos permite almacenar una imagen en formato BMP), y Salir (que cierra y detiene la ejecución del programa) como se puede apreciar en la fig. 4.3



Figura 4.3 Submenús del menú archivo

Para trabajar con la imagen de la hoja, es necesario que se encuentre en forma binaria, por lo que se ha implementado una etapa de pre-procesamiento en la que dos filtros se encargan de transformar una imagen en RGB a una imagen binaria; para ello el menú filtro cuentan con dos submenús Otsu (método otsu para seleccionar umbral optimo) y mediana (para reducir el ruido en la imagen), ver fig. 4.4.

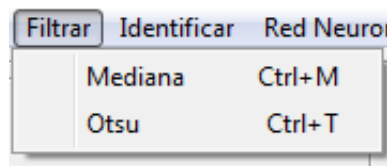


Figura 4.4 Submenús del menú filtrar

En la fig. 4.4 también podemos observar el menú identificar, el cual se encarga de realizar todos los procesos de identificación de la especie que se está analizando.

El menú red neuronal cuenta con los siguientes submenús Ajustar pesos donde se pueden modificar los pesos de los valores predeterminados (ver fig. 4.5), y submenú entrenamiento en el cual se realizan los

entrenamientos necesarios para cuando se desea incluir una nueva especie, ver fig. 4.6.

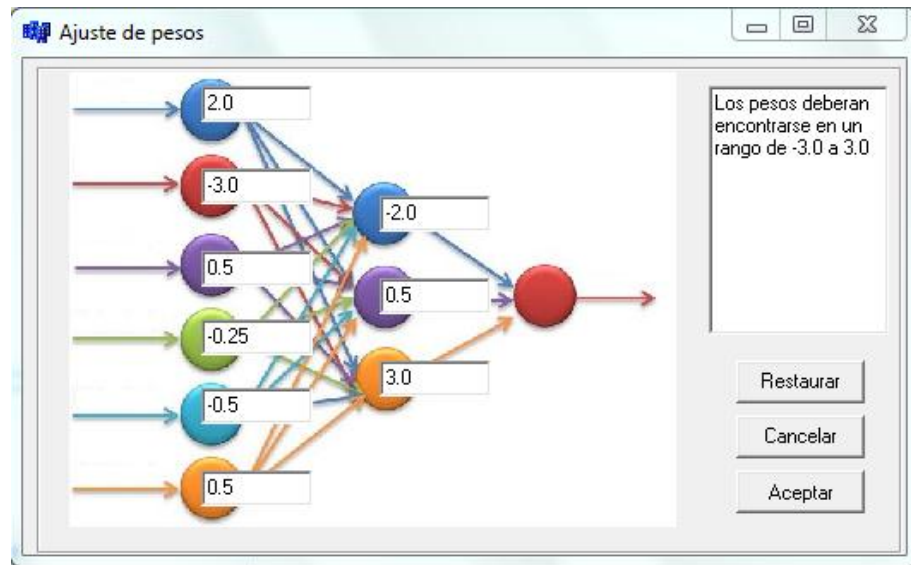


Figura 4.5 Ventana para modificación de pesos

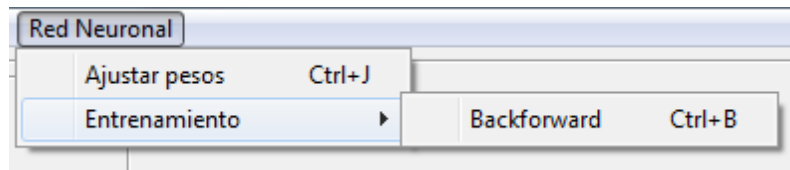


Figura 4.6 Submenús del menú Red neuronal

4.3. Adquisición de la Imagen

Para la adquisición de la imagen se ha empleado un escáner convencional, en el cual se recorta manualmente la imagen al tamaño aproximado de la hoja (para evitar aumentar el tiempo de procesamiento en el análisis de porciones innecesarias), dejando un margen para obtener la hoja completa; este dispositivo se ha configurado para que proporcione las imágenes

de la hoja en formato BMP, debido a su fácil lectura y manejo a través del entorno de programación C++ Builder. Cada imagen fue etiquetada con un nombre para su identificación, ver fig. 4.8 y 4.9.

Como primer grupo de las muestras de las distintas especies se muestran todas aquellas de las cuales pudimos obtener su nombre común, no obstante no se afirma que el nombre asignado sea el correcto ya que algunos de los nombres fueron encontrados a través de foros especializados y dedicados al trato con la flora en los cuales en distintos países o regiones se les conoce por un nombre distinto; algunas de estas especies pudieron confirmarse en algunos viveros locales dedicados al cultivo de flora silvestre.

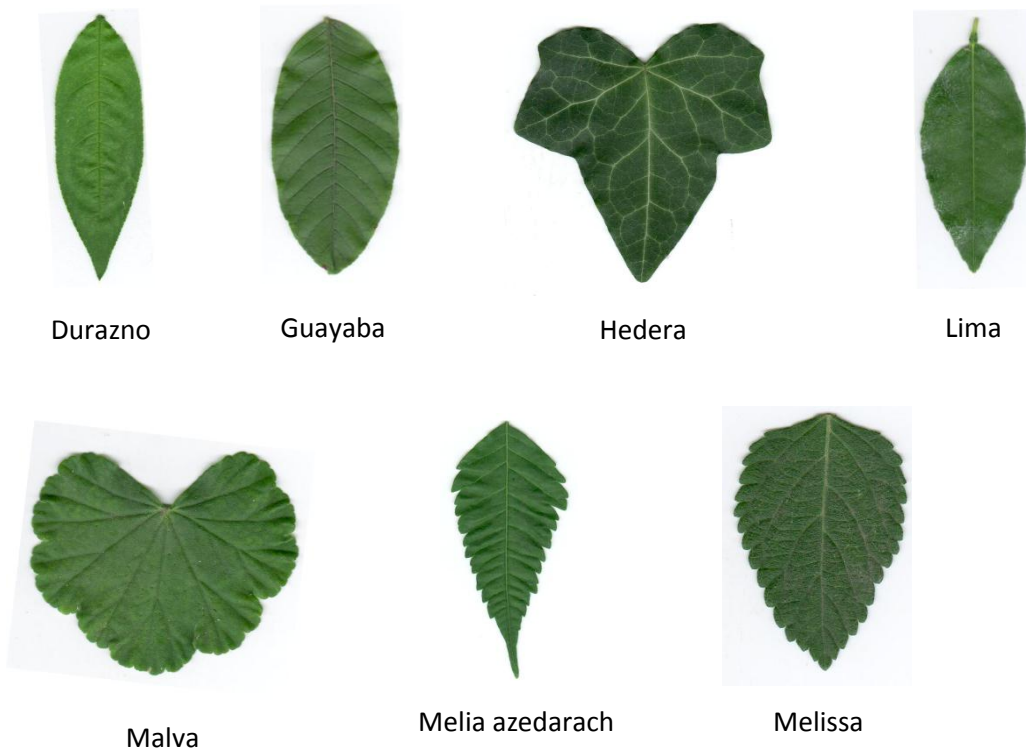


Figura 4.8 Especies con nombre común

Como segundo grupo se muestran las especies de las cuales no se pudo obtener su nombre, y por lo cual se les asignó una etiqueta o sobrenombre para poder identificarlas.

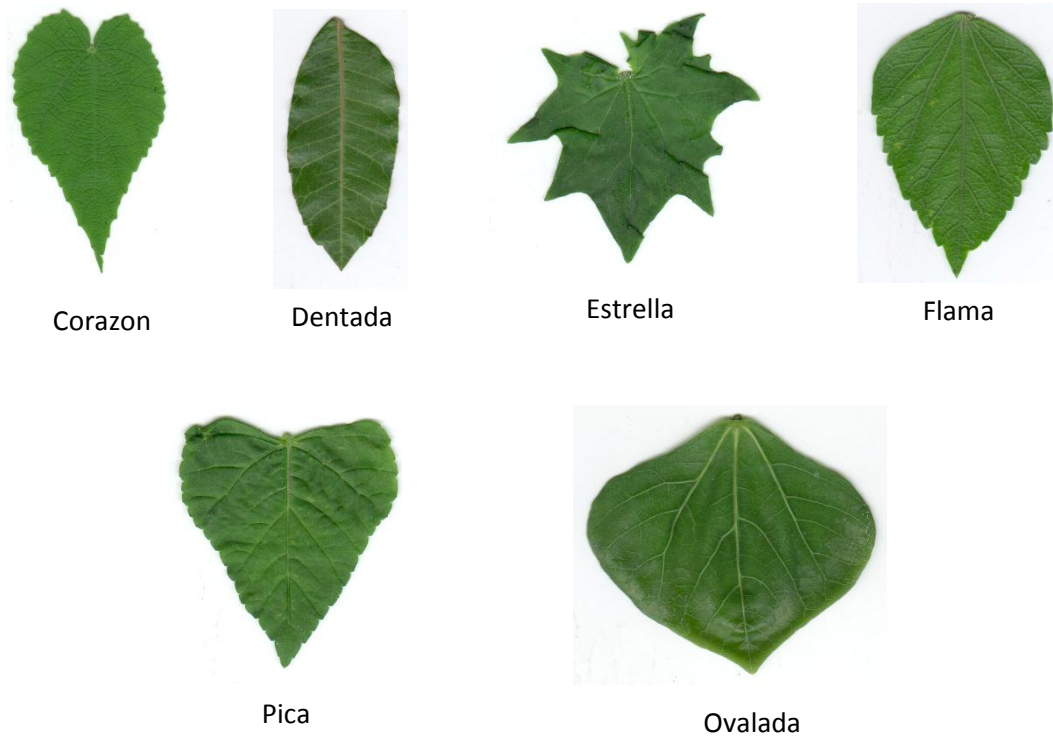


Figura 4.9 Especies con etiqueta como nombre

Para fines de este proyecto todas las imágenes capturadas fueron convertidas de RGB a binarias y etiquetadas por medio del mismo algoritmo desarrollado; a continuación se presentan las mismas hojas tomadas como ejemplo en forma binaria, ver la fig. 4.10.



a) Durazno_H1_003



b) Guayaba_H1_002



c) Lima_H1_006

Figura 4.10 Ejemplos de imágenes escaneadas y binarizadas: a) corresponde a la tercera hoja de Durazno, la figura b) corresponde a la segunda hoja de Guayaba, la figura c) corresponde a la sexta hoja de Lima.

La estructura de bloques para la lectura de la imagen en formato BMP es presentada a continuación:

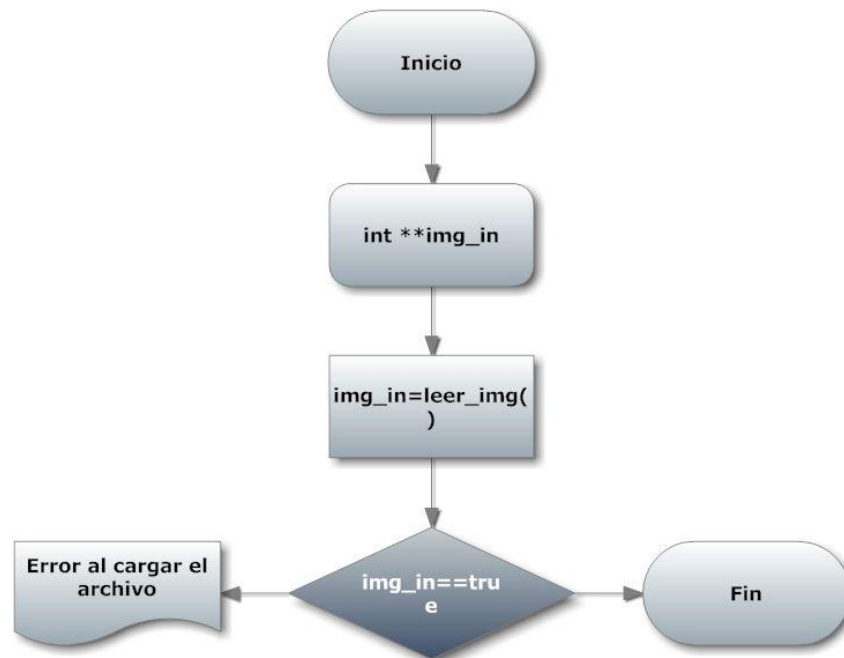


Figura 4.11 Diagrama de bloques para la lectura del BMP

Observando la fig. 4.11, en el primer bloque se hace la declaración para un arreglo bidimensional del tipo entero el cual contendrá los valores de la imagen; en el segundo bloque un método encargado de la lectura de la imagen, el cual consta de un filtro para leer exclusivamente imágenes en el formato BMP debido a su facilidad de manejo; en el tercer bloque se realiza la búsqueda de la imagen a procesar, finalmente en el último bloque se verifica que la imagen haya sido cargada correctamente, y en caso de que no haya sucedido así se presenta un mensaje al usuario a través de la pantalla.

4.4. Procesamiento

Para poder manipular la imagen es necesaria la transformación de la imagen RGB a un modo binario; primeramente, en esta etapa convertiremos una imagen RGB a una en escala de grises, dado que un color hace referencia a la mezcla de los colores primarios con que se forma: el rojo, el verde y el azul. Estos colores pueden ser representados por 8 bits, es decir el rojo se obtiene con (255,0,0), el verde con (0,255,0) y el azul con (0,0,255), obteniendo, en cada caso un color resultante monocromático. La ausencia de color —lo que nosotros conocemos como color negro— se obtiene cuando las tres componentes son 0, (0,0,0); obviamente, el color blanco se forma con los tres colores primarios a su máximo nivel (255,255,255).

Una vez entendido este concepto podemos realizar el cambio de RGB a grises de una forma sencilla, ya que la trama está compuesta de #RRVVAA, donde RR representa los 8 bits para el color rojo, VV los 8 bits para el color verde y AA los 8 bits para el color azul, así de esta forma generaremos el diagrama de flujo para obtener la transformación de RGB a escala de grises.

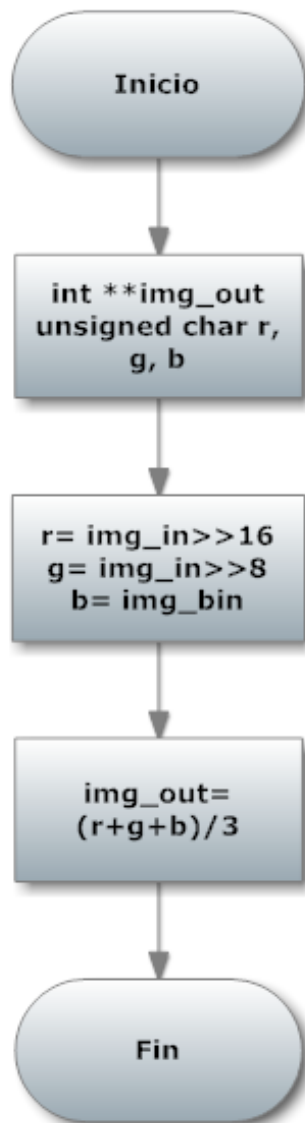


Figura 4.12 Diagrama de bloques para pasar de RGB a escala de grises

De la fig. 4.12 en el primer bloque se declaran las variables para obtener una copia de la imagen y las variables para obtener los colores, en el segundo bloque se presentan la obtención de los colores por medio de corrimientos, en el último bloque se hace la obtención del color de gris, promediando los tres colores básicos. Los resultados son mostrados en la fig. 4.13.

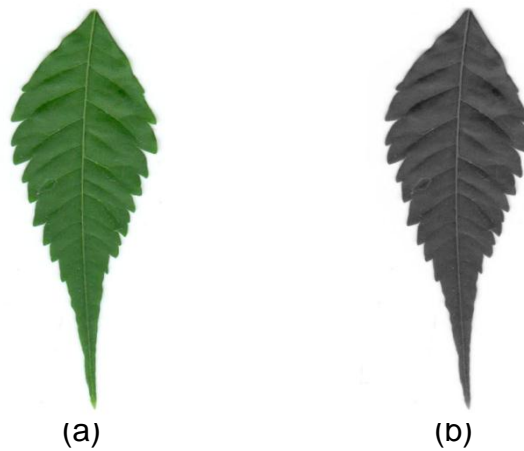


Figura 4.13 Conversión de imagen RGB a escala de grises. (a) Imagen en color RGB, (b) Imagen en escala de grises

4.4.1. Segmentación por OTSU

El método que se utiliza para realizar la segmentación en las imágenes, está basado en el primer grupo, este método es la umbralización por OTSU; se utiliza esta técnica debido a que la imagen muestra una clara diferencia entre nuestro objeto a extraer (la forma de la hoja) y el fondo. Para observar el resultado con el método de otsu ver fig. 4.14.

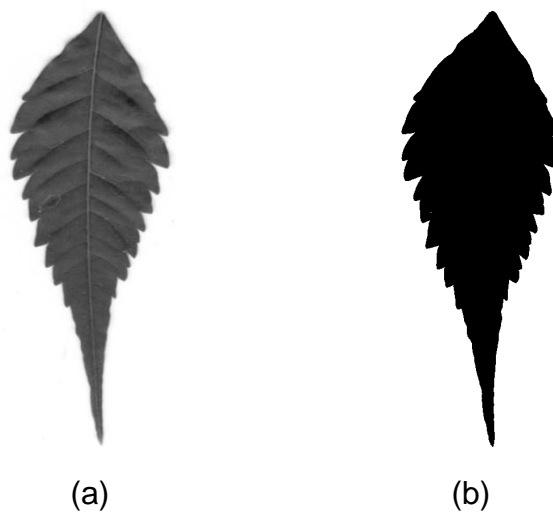


Figura 4.14 Conversión de imagen RGB a escala de grises. (a) Imagen en escala de grises, (b) Imagen aplicando el método OTSU

A pesar de que se ha logrado la transformación de una imagen a color a una binaria (donde el fondo puede quedar en blanco y el cuerpo de la hoja en negro o viceversa), esta imagen presenta algunas correcciones que deben realizarse; ampliando la sección media de la hoja (ver fig. 4.15) podemos notar claramente la presencia de ruido visualizada como un agujero, que puede ser corregido por medio de un filtro.

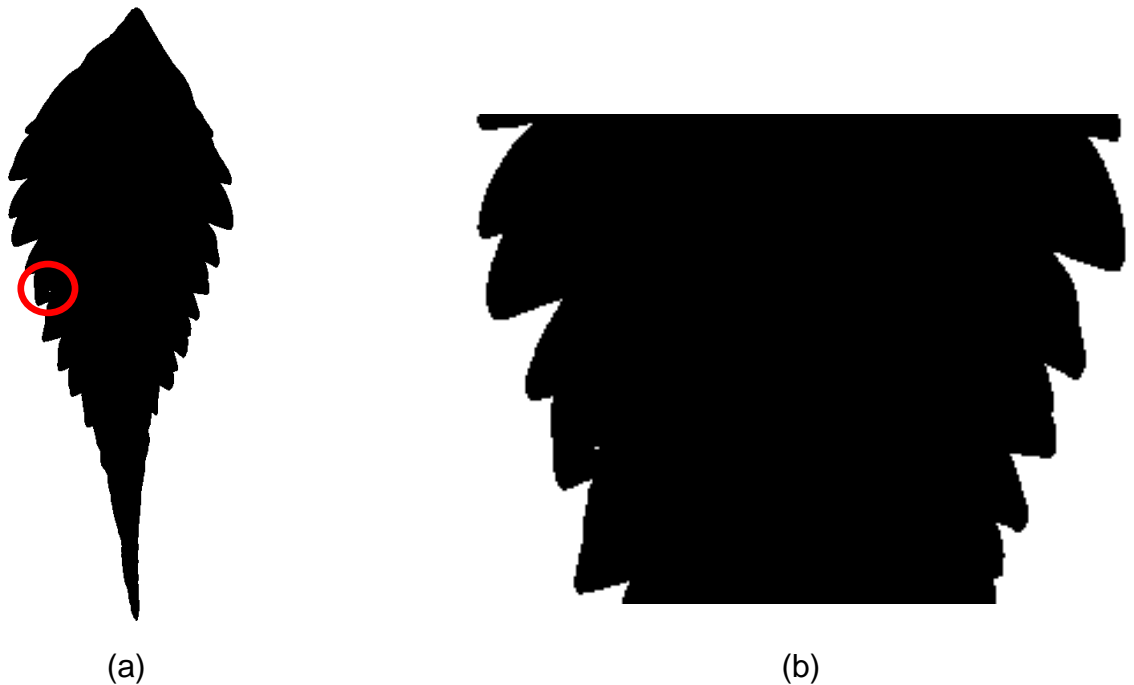


Figura 4.15 (a) Imagen tras aplicación de otsu, (b) Ampliación de la imagen a

4.4.2. Filtro mediana

Como se mencionó en el capítulo 2, existen varios tipos de filtros, pero para este proyecto se usa el filtro de mediana al presentar mejores resultados al eliminar de manera efectiva el ruido, sin alterar la imagen original y sus bordes.

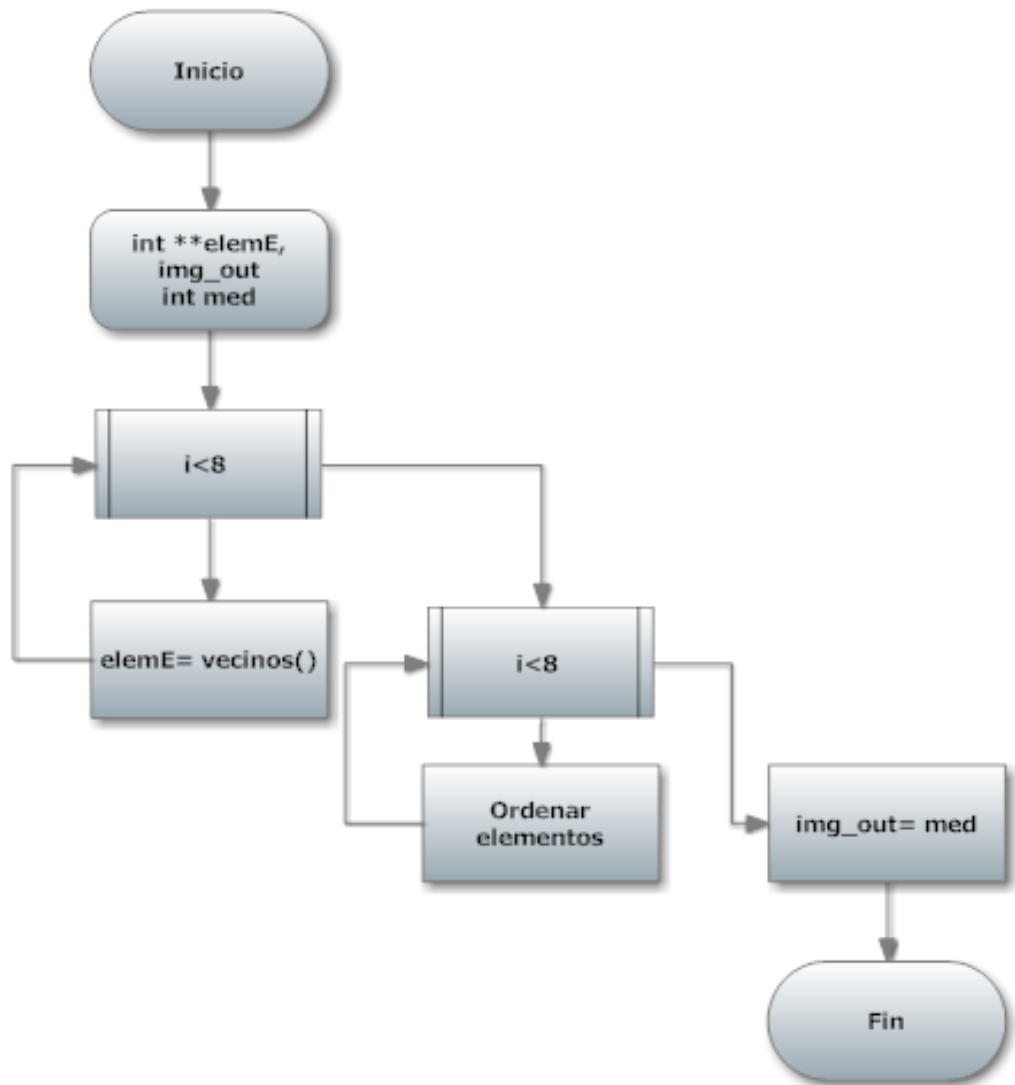


Figura 4.16 Diagrama de flujo del filtro de mediana

La fig. 4.16 nos muestra el diagrama de flujo de la mediana; en el primer bloque se hace referencia a las variables usadas como dos arreglos bidimensionales enteros, uno para el tamaño de la ventana del filtro (elemenE) y otro para la copia de la imagen (img_out). El primer ciclo dentro del diagrama es para obtener los elementos de la ventana del filtro correspondientes a las coordenadas de la imagen que se analiza; una vez llena la ventana del filtro, se

ordenan los elementos por medio del método de burbuja, y se saca la mediana de dichos valores, asignando este resultado a un elemento de la matriz correspondiente como copia de la imagen. Los resultados de la aplicación de este filtro podemos verlos a continuación en la fig. 4.17.

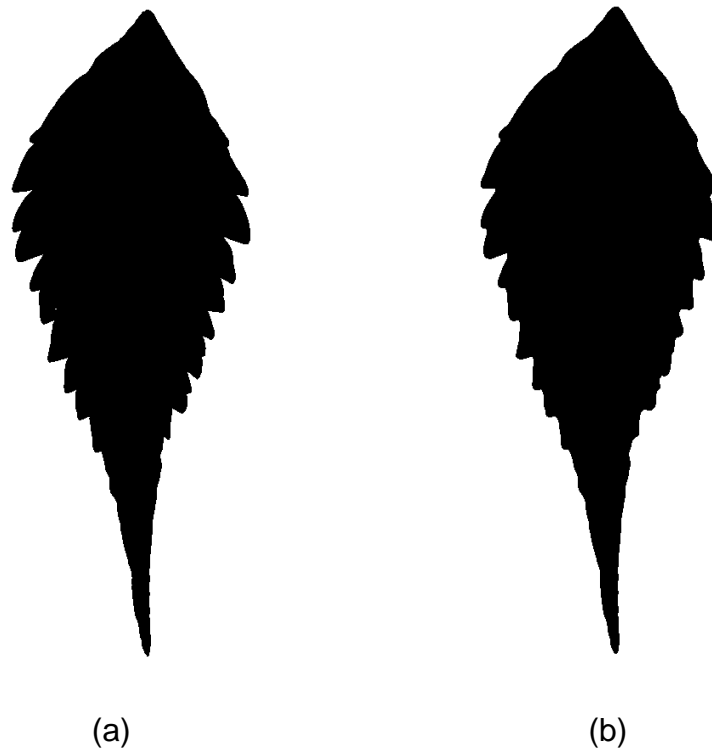


Figura 4.17 (a) imagen tras aplicar el método de OTSU,
(b) Imagen aplicando OTSU y filtro de mediana

4.5. Extracción de características

Una vez procesada y segmentada la imagen, de ella se obtendrán las características antes mencionadas, para definir los parámetros que nos darán mayor información sobre las hojas. Los primeros parámetros a definir serán los momentos invariantes, puesto que es necesario analizar cuáles de estos momentos no presentan grandes variaciones dentro de la misma especie. En

este capítulo se limitará a mostrar los resultados del procedimiento de las ecuaciones mostradas en el capítulo 2, así como la selección de parámetros; además se efectuará el cálculo de la excentricidad y la circularidad (también mencionadas en el segundo capítulo).

Obtención del borde de la imagen

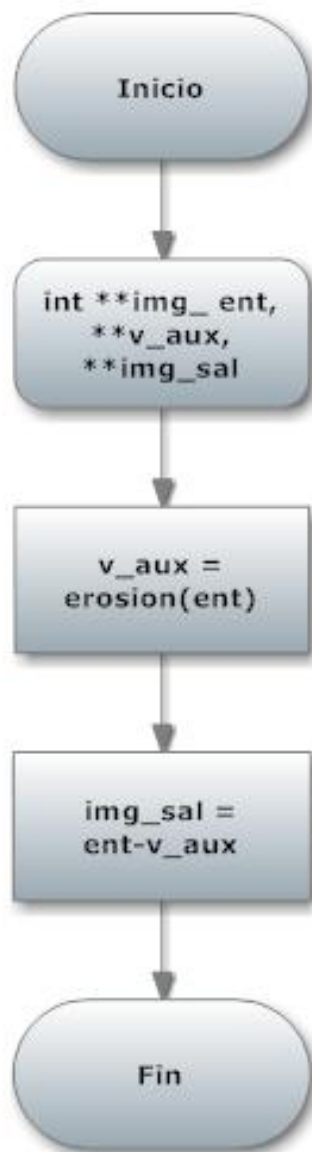


Figura 4.18 Proceso para obtención del borde

La fig. 4.18 nos muestra el proceso para la obtención del borde de la imagen; en el primer bloque de proceso se encuentran las definiciones de las variables a usar, matriz de entrada, de salida y una matriz auxiliar para realizar las operaciones necesarias; en el segundo bloque se realiza el erosionado de la imagen (img_ent), y se asigna a la matriz auxiliar v_aux; por último para el tercer bloque de proceso se realiza la diferencia entre la matriz que contiene la imagen original y la matriz v_auxiliar, para de este modo obtener el borde de la imagen.

Obtención del perímetro de la imagen

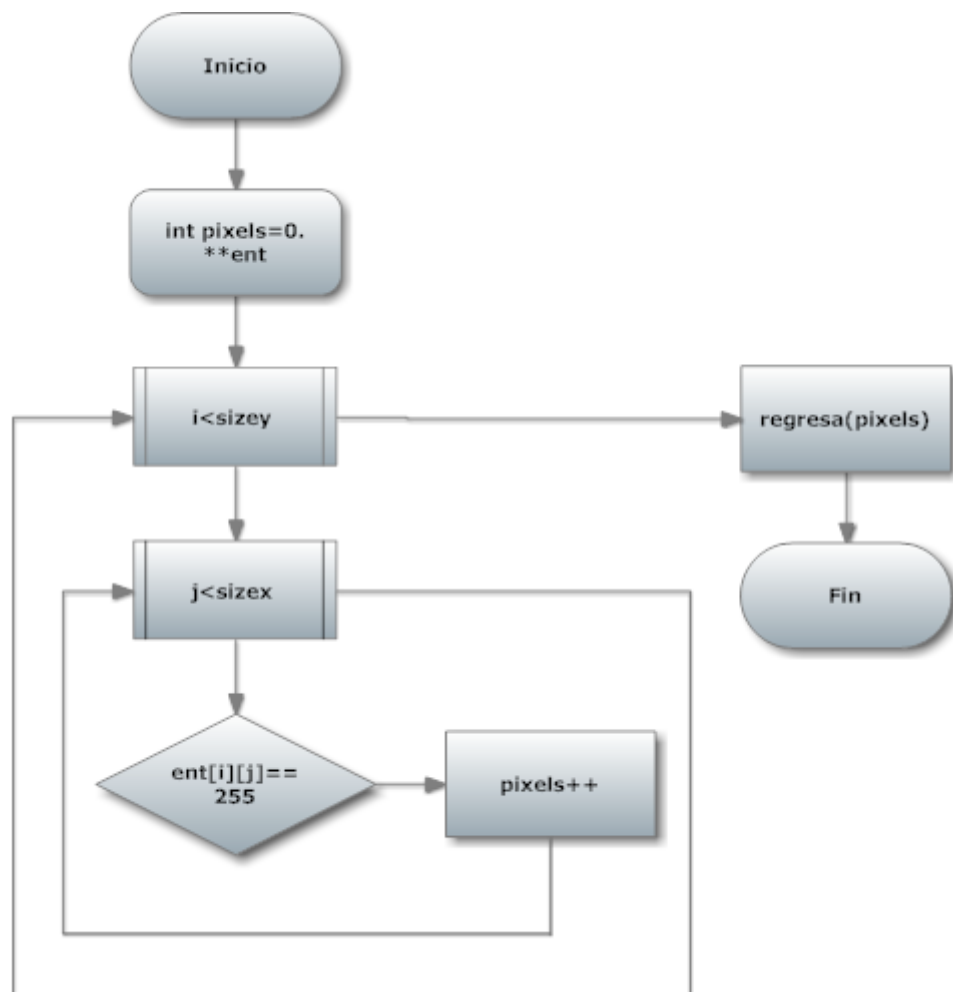


Figura 4.19 Proceso para obtener el perímetro de la imagen

Obtener el perímetro de la imagen que estamos analizando resulta sencillo debido a que solo necesitamos contar el número de píxeles del borde de la imagen, de esta forma la imagen de entrada será la imagen de salida de la fig. 4.18. El proceso consiste en verificar que el píxel sea de color blanco (lo que indica que forma parte de la hoja) para contabilizarlo ver fig. 4.19.

Obtención de la circularidad, factor de elongación, largo y ancho de la imagen

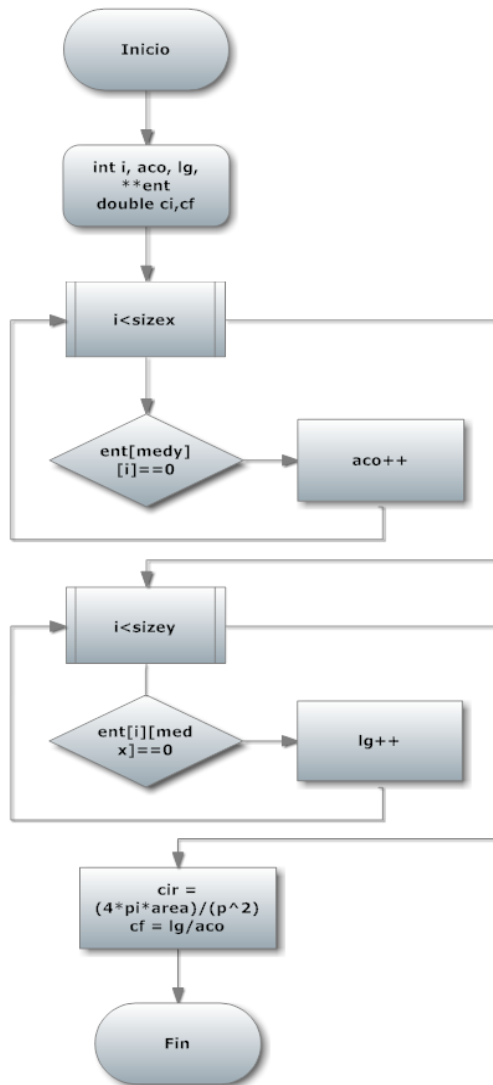
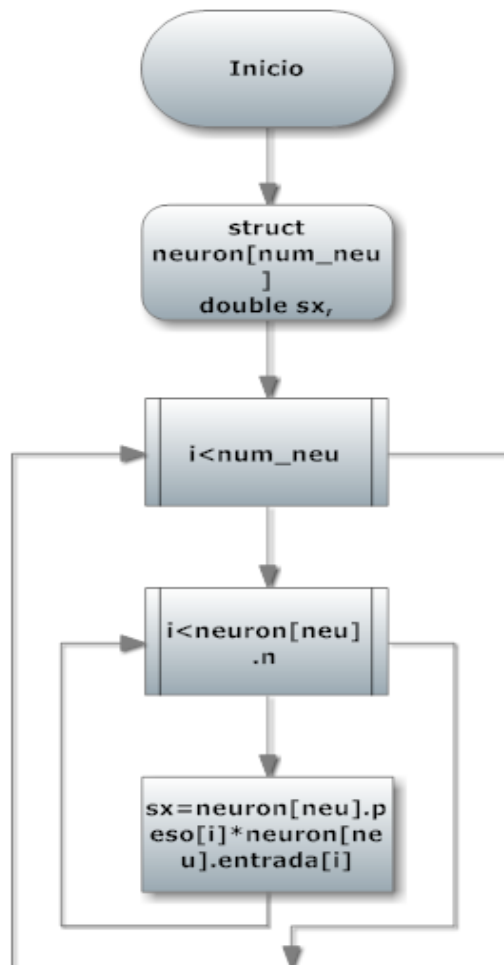


Figura 4.20 obtención de la circularidad, largo y ancho de la imagen

Ahora la fig. 4.20, muestra la forma empleada para obtener el largo, ancho y la circularidad de la imagen; para esto el primer bucle o ciclo obtiene el número de pixeles a lo ancho de la imagen a partir de la posición central del alto de la hoja; el segundo bucle obtiene el número de pixeles a lo largo de la imagen a partir de la posición central del ancho de la imagen. Para obtener la circularidad basta aplicar la fórmula del penúltimo bloque de proceso el factor de elongación que es obtenido de la razón entre la longitud y el ancho de la imagen.

4.6. RNA Aplicación/Entrenamiento



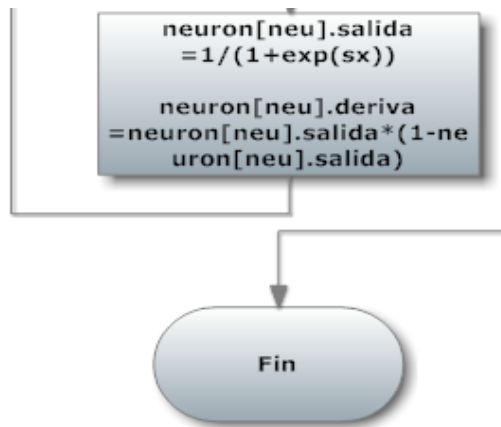


Figura 4.21 Proceso para aplicación de RNA

En la fig. 4.21, se observa el proceso de la red neuronal implementada en este caso perceptron múltiple. En el bucle interno se emplea la función de activación, el cual multiplica la entrada a cada neurona por su peso; esto se realiza para cada uno de las neuronas de la RNA. El bucle o ciclo externo contiene la función de propagación en este caso una función sigmoidea y su derivada; estos datos son almacenados en la estructura de cada neurona. El contenido de la estructura llamada neurona puede verse en la fig. 4.22.



Figura 4.22 Contenido de la estructura neurona

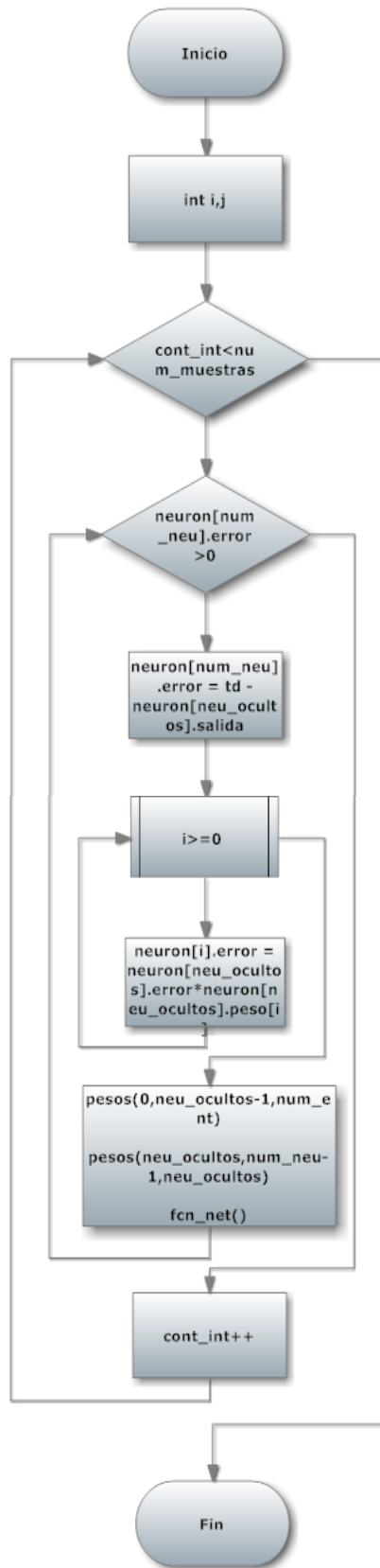


Figura 4.23 Proceso para entrenamiento por backforward

El proceso para entrenar la red es un poco más complicado, debido a que esta vez se parte de la capa de las neuronas de salida hasta llegar a la capa de las neuronas de entrada, ver fig. 4.23; en el bucle más externo, primeramente se revisa que el proceso que se vaya a realizar sea para todas las muestras; en el bucle intermedio, se revisa que el error sea mayor que cero, esta convención para generar más precisión a la hora del entrenamiento; en el siguiente bloque que es de proceso se obtiene el error que se genera en cada neurona, una vez obtenido el error de cada neurona, este se multiplica por cada uno de los pesos establecidos en la red, esto puede observarse en el bucle interno, así solo resta calcular los nuevos pesos para la red, como puede observarse en el bloque de proceso de la fig. 4.23.

Dentro del entrenamiento también como una segunda etapa se obtiene el vector de valores de la salida de la red neuronal, correspondientes a el conjunto o base de datos recabado, esto es 10 valores neurales por cada especie; estos valores son ordenados de mayor a menor y de ellos se obtiene el valor medio, de esta forma se obtiene una aproximación para determinar a qué especie pertenece la hoja que se está analizando.

4.7. Identificación

Ahora el proceso de identificación es más sencillo y para ello es necesario el valor neuronal de la RNA de la hoja que se encuentra analizando; con este valor se hace una comparación contra cada uno de los valores representativos de los grupos de especies, y así se obtiene un valor con cierta precisión. El proceso usado para la identificación puede verse en la fig. 4.24.

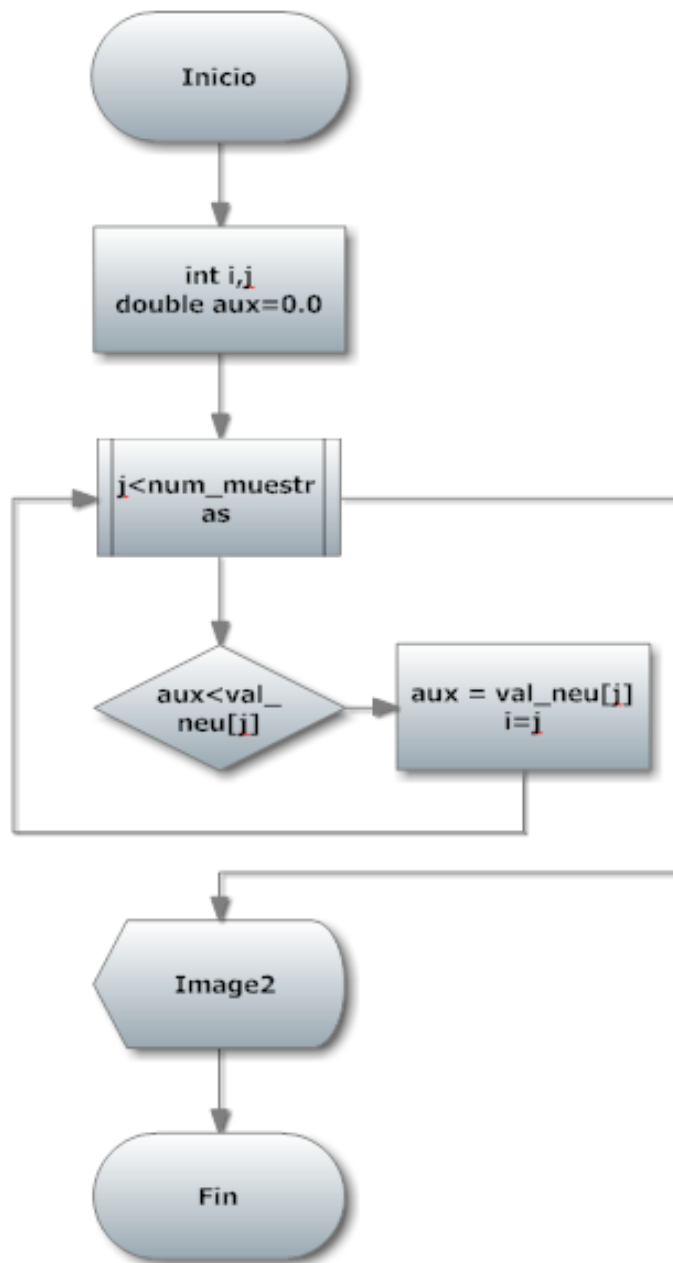


Figura 4.24 Proceso para identificación de especies

5. RESULTADOS

Con las herramientas implementadas hasta ahora, es posible obtener una forma de clasificar las especies; al abrir la imagen el programa nos proporciona datos importantes de la hoja en análisis, como lo son el ancho (que es la distancia entre los puntos extremos izquierdo y derecho en la dirección horizontal), el área foliar (que es la cantidad de superficie de hoja que ella posee), la longitud (que se mide como la distancia más grande entre los puntos de mayor a menor y que pertenecen a la hoja en la dirección vertical), el perímetro, y el factor de elongación. Para identificar la especie que se analiza, simplemente se da clic en el menú analizar y este realizara los procedimientos necesarios para identificar la especie; a continuación se muestran algunos ejemplos de la ejecución del programa y su respuesta que proporciona.

5.1. Pruebas sobre hojas de durazno

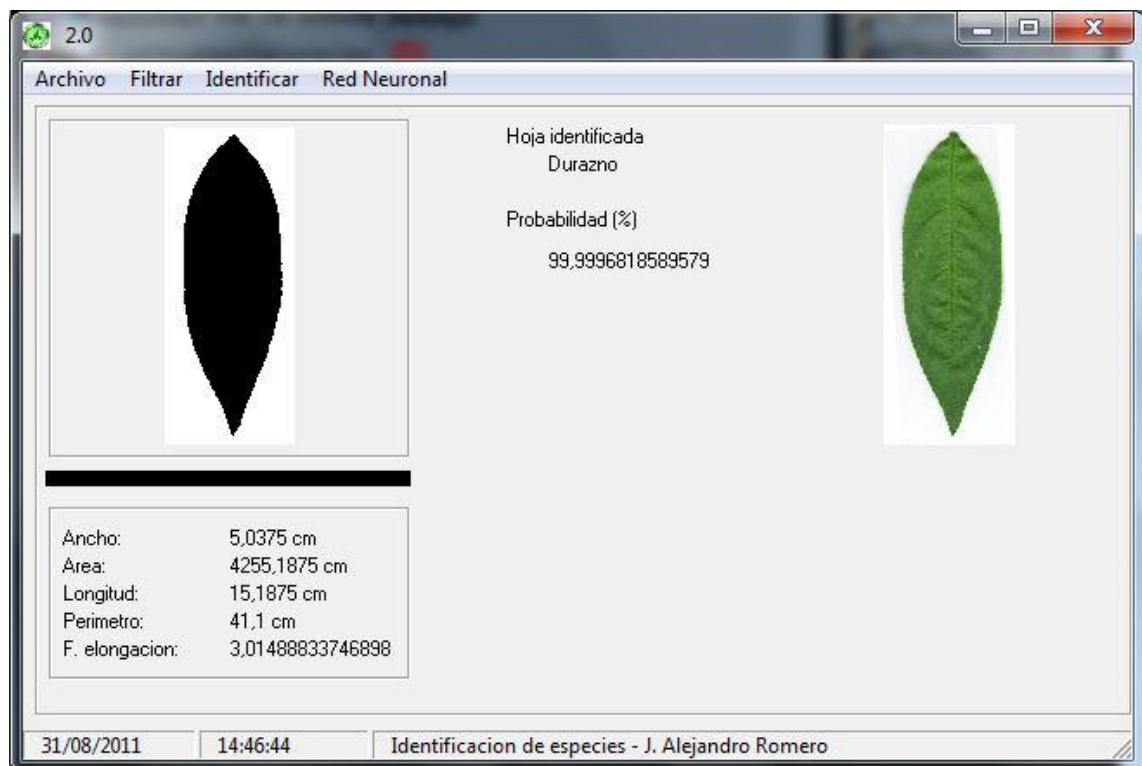


Figura 5.1 Hoja etiquetada como Durazno_H1_003

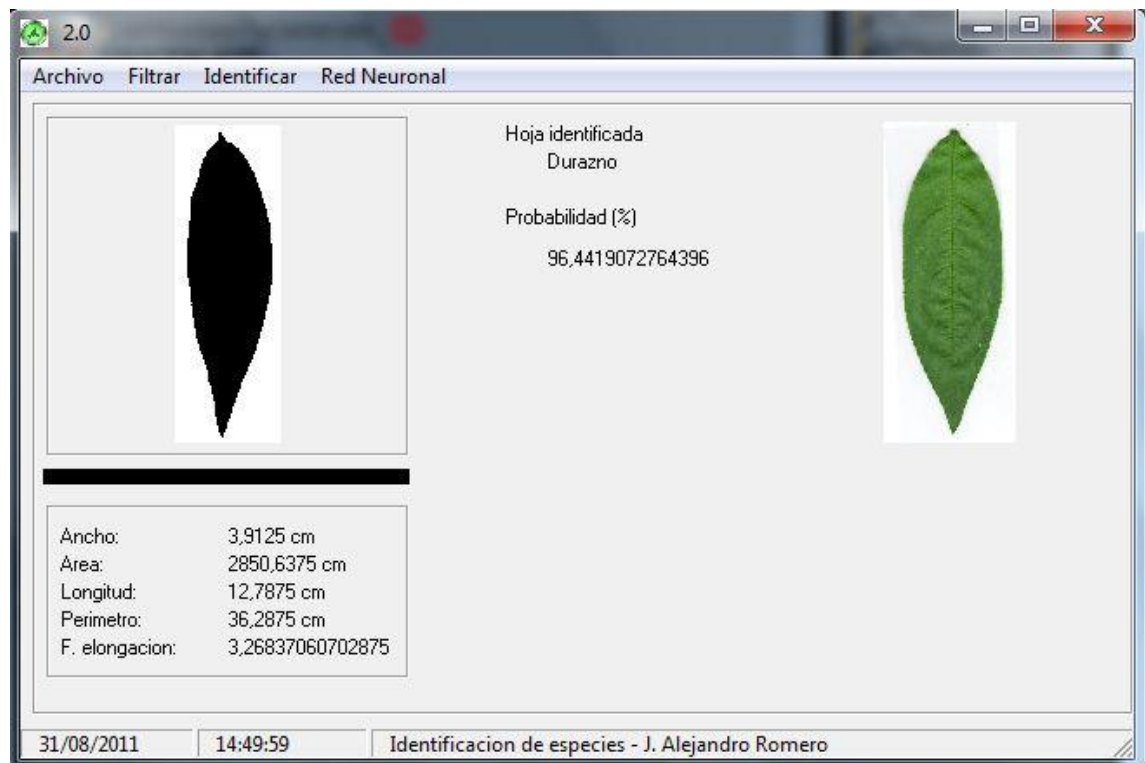


Figura 5.2 Hoja etiquetada como Durazno_H1_011

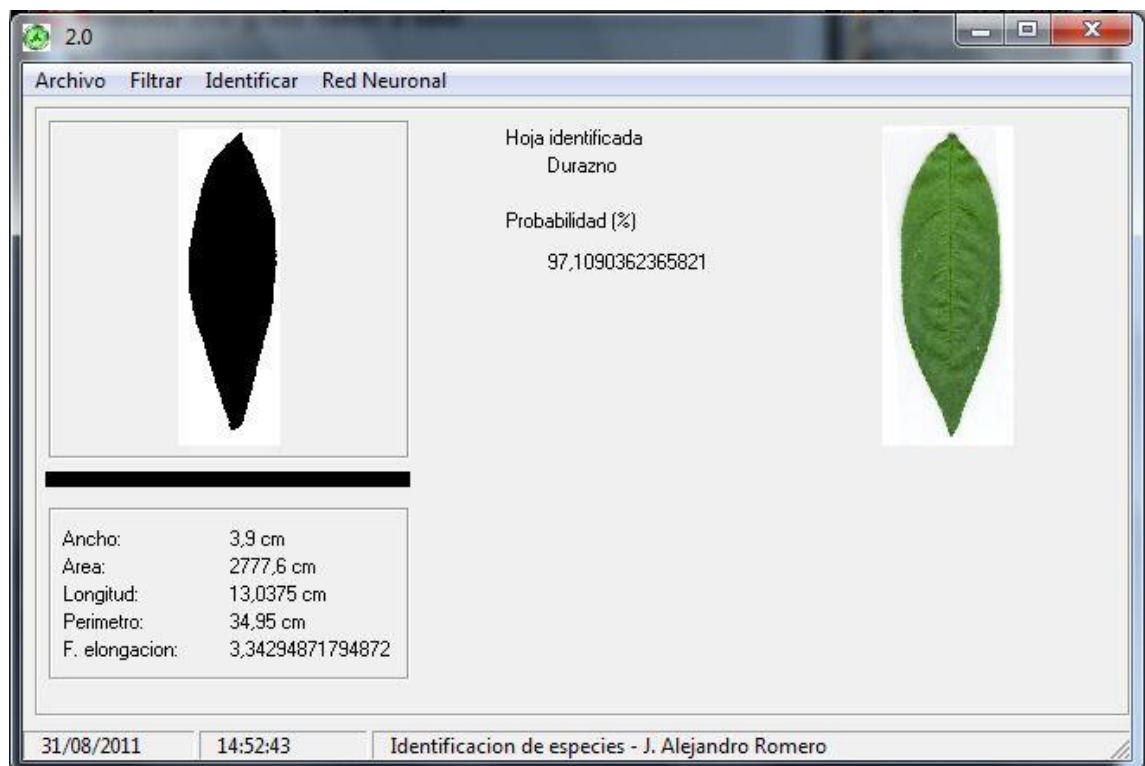


Figura 5.3 Hoja etiquetada como Durazno_H1_021

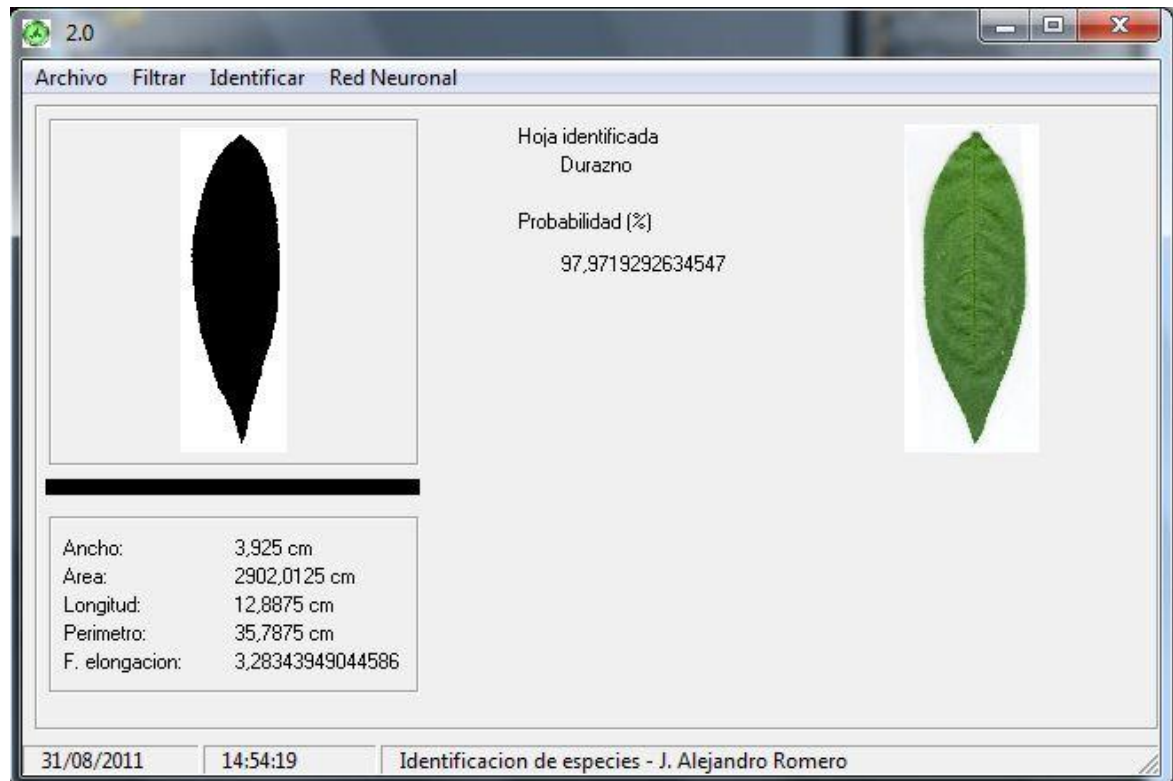


Figura 5.4 Hoja etiquetada como Durazno_H1_022

Para esta especie que es durazno los resultados mostrados fueron muy favorables, esto es debido a que las hojas no muestran imperfecciones como rupturas, agujeros, etc, lo cual favorece a los parámetros obtenidos de la hoja en análisis (fig. 5.1 hasta la fig. 5.4). La imagen a color que se muestra en las figuras es una imagen patrón elegida por carecer de imperfecciones; además sirve como indicativo visual, que le permita conocer que información se le está proporcionando.

5.2. Pruebas sobre hojas de Hedera

Este tipo de hoja que es muy distintiva por su peculiar forma también fue identificada correctamente; en la fig. 5.7 podemos observar un giro sobre uno de los picos de la hoja, pero que a pesar de dicha inclinación el programa reconoce la especie a la que pertenece debido a que la hoja no presenta deformidades y su estructura está completa.

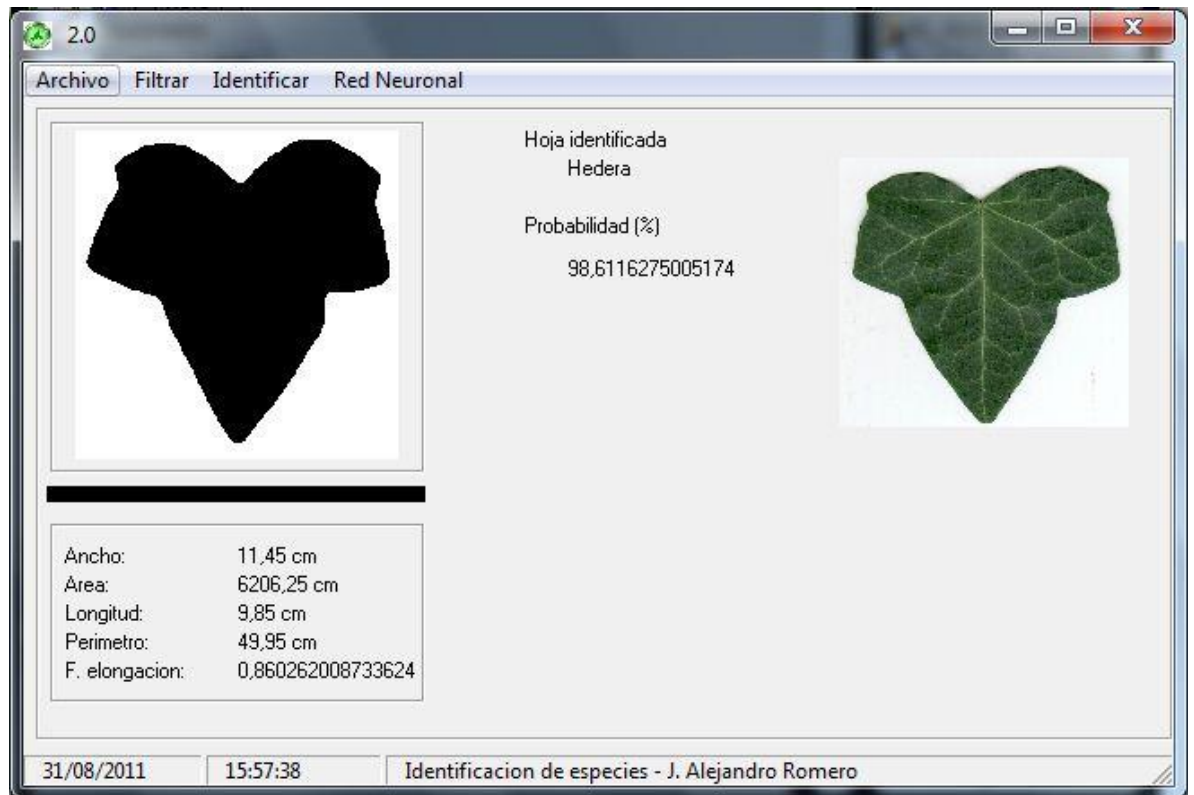


Figura 5.5 Hoja etiquetada como Hedera_H1_001

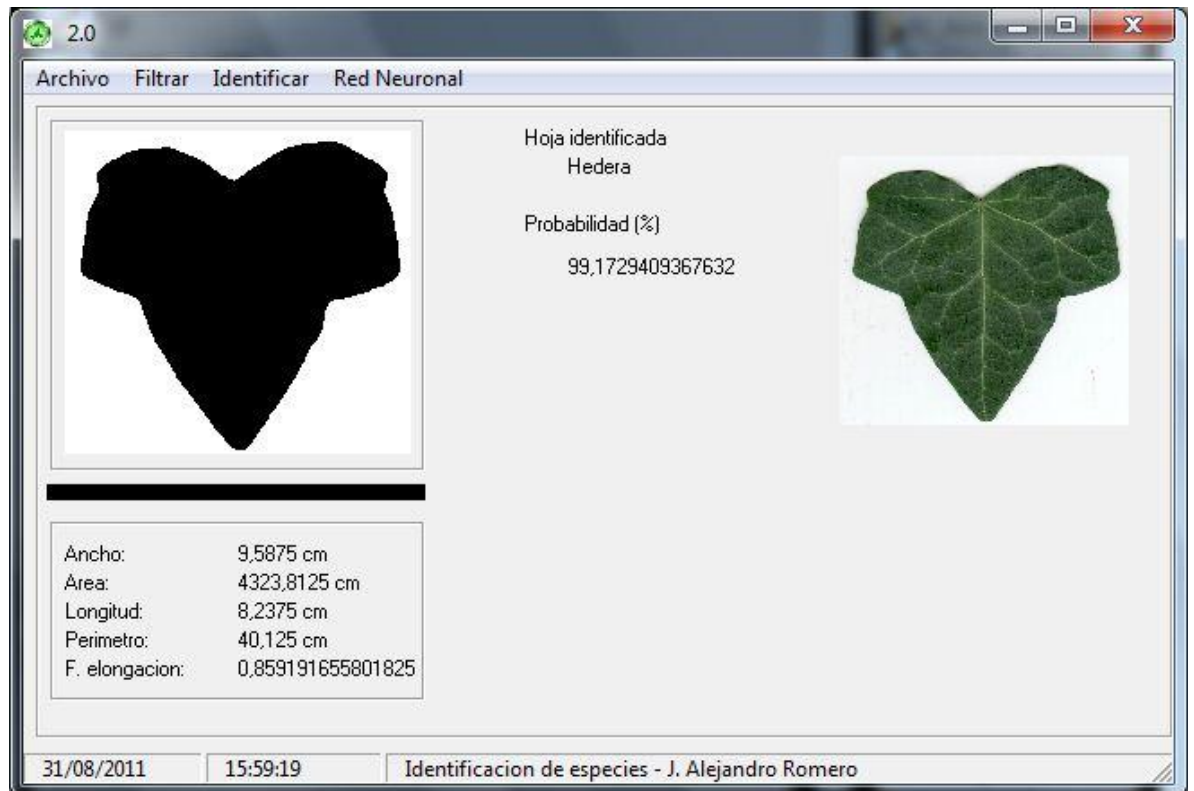


Figura 5.6 Hoja etiquetada como Hedera_H1_003

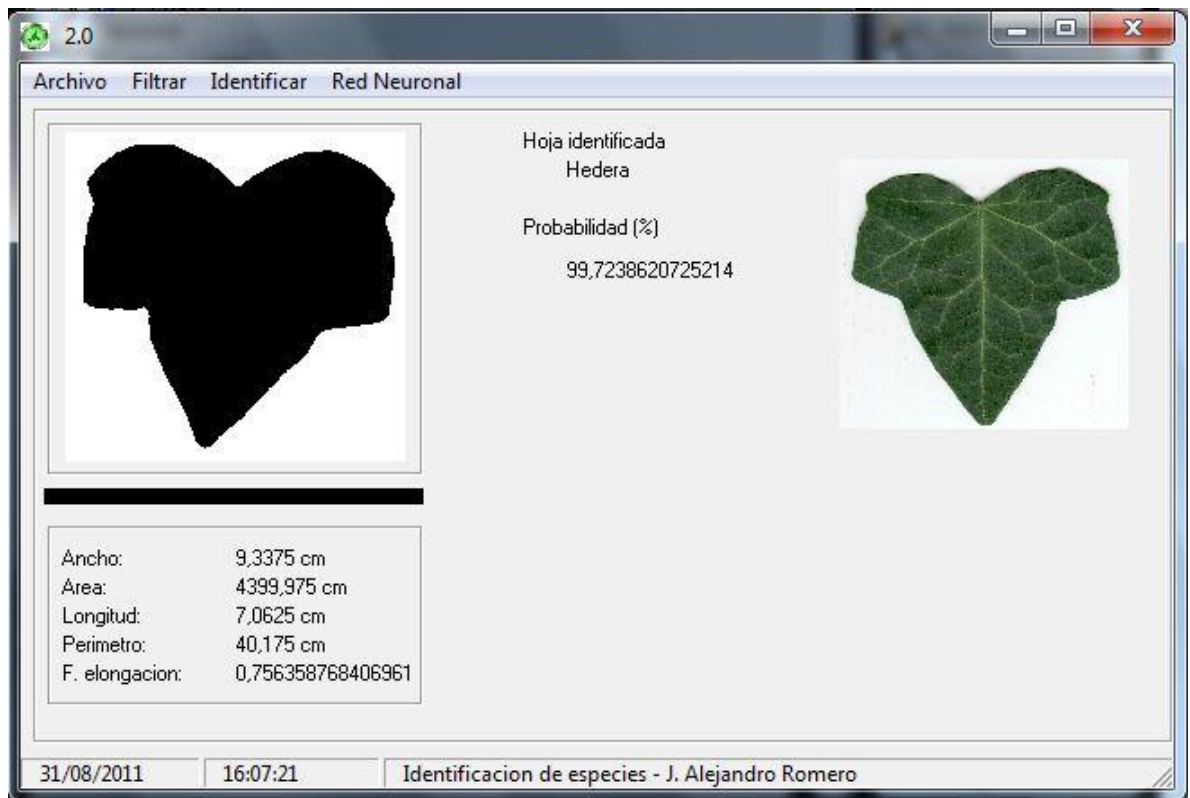


Figura 5.7 Hoja etiquetada como Hedera_H1_007

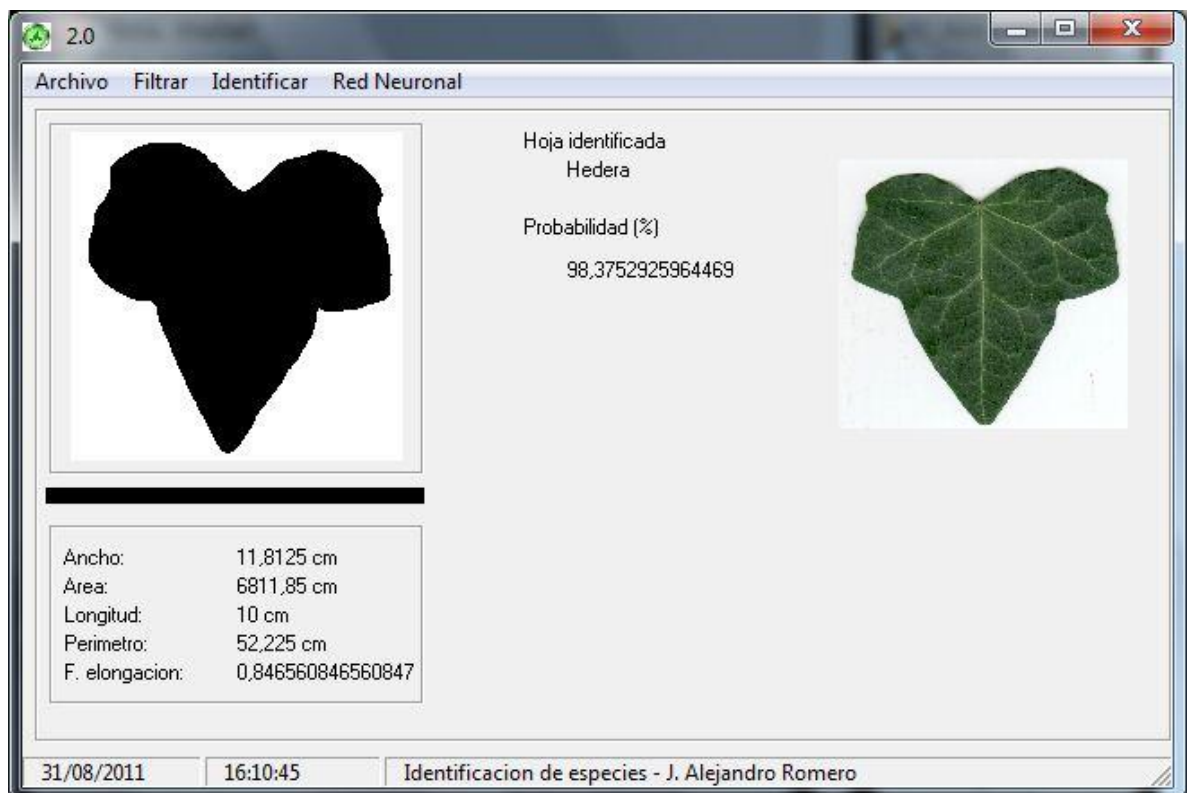


Figura 5.8 Hoja etiquetada como Hedera_H1_010

5.3. Pruebas sobre hoja Melia

En las muestras tomadas de la especie identificada como Melia se puede apreciar que cada una de ellas conforma su estructura sin algún daño o ruptura, además de que su característico perímetro permite que los datos que maneja el programa no tenga problemas a la hora de hacer la clasificación de esta especie. Se aprecia en las imágenes mostradas que los resultados fueron favorables estando entre el 98 y el 99 % de probabilidad de que sea la especie correcta.

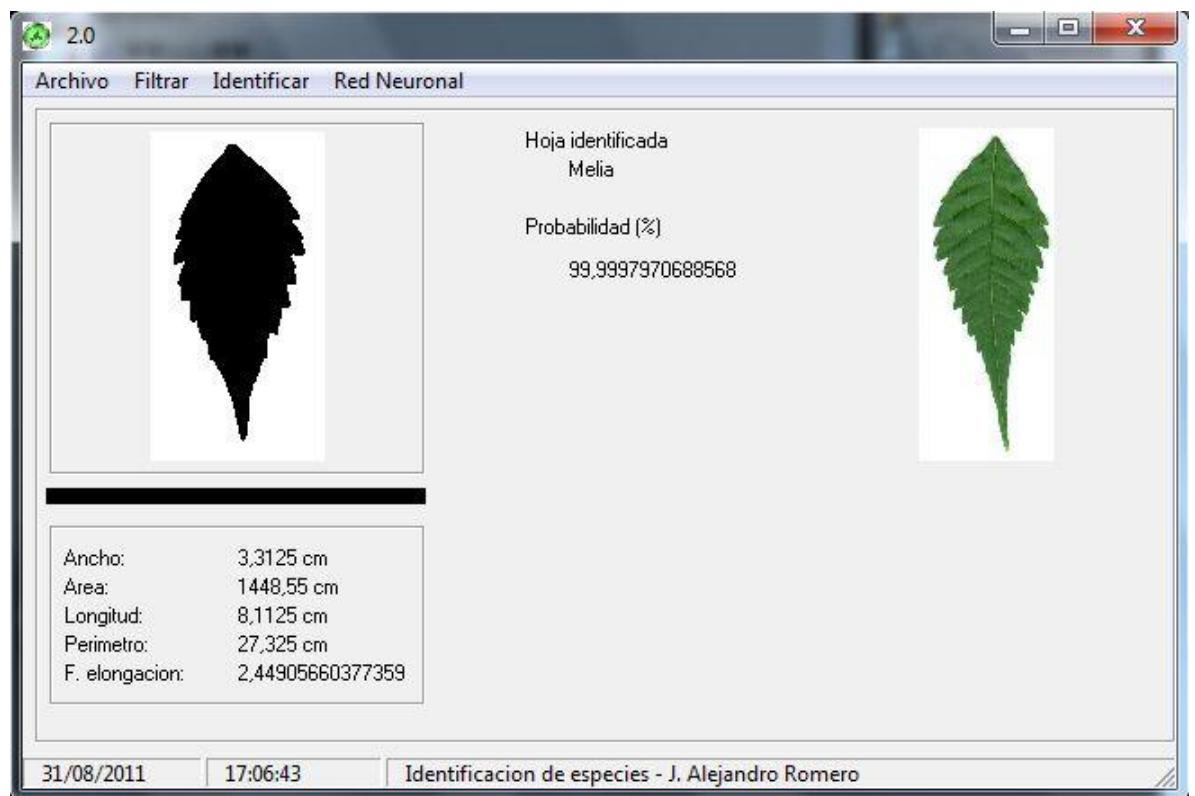


Figura 5.9 Hoja etiquetada como Melia_H1_004

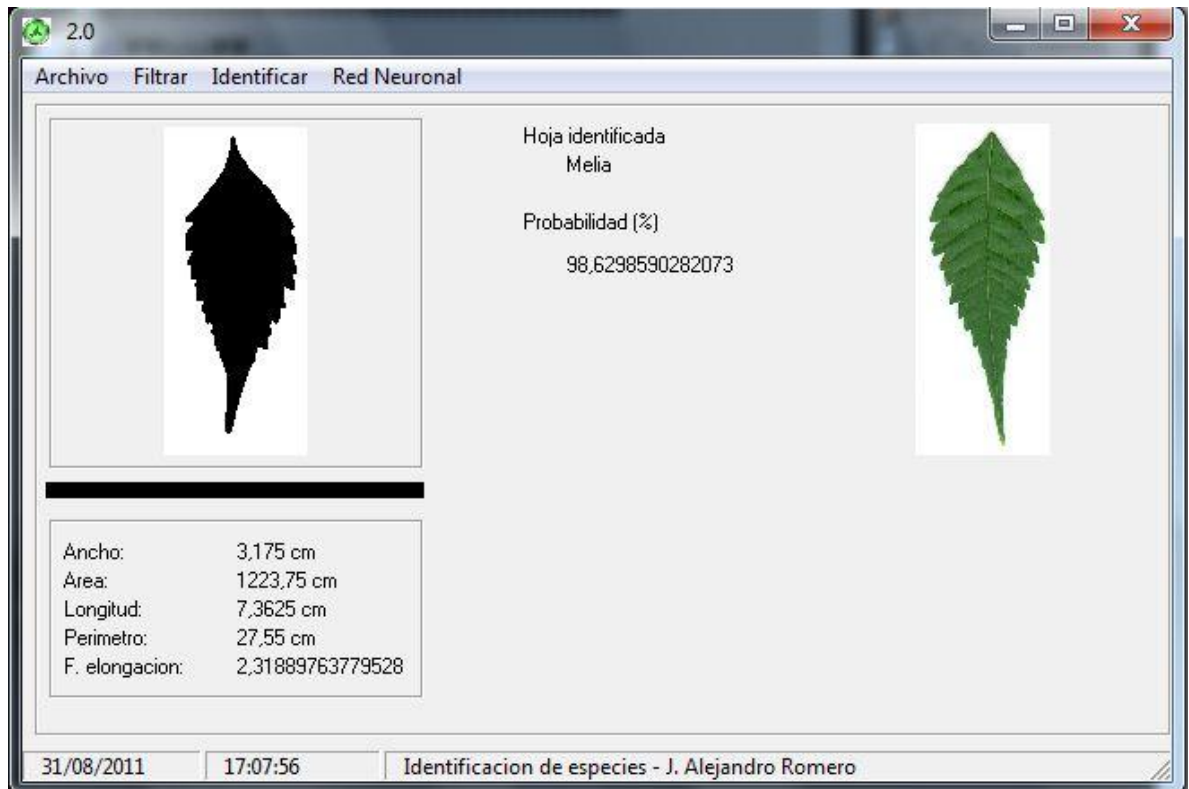


Figura 5.10 Hoja etiquetada como Melia_H1_006

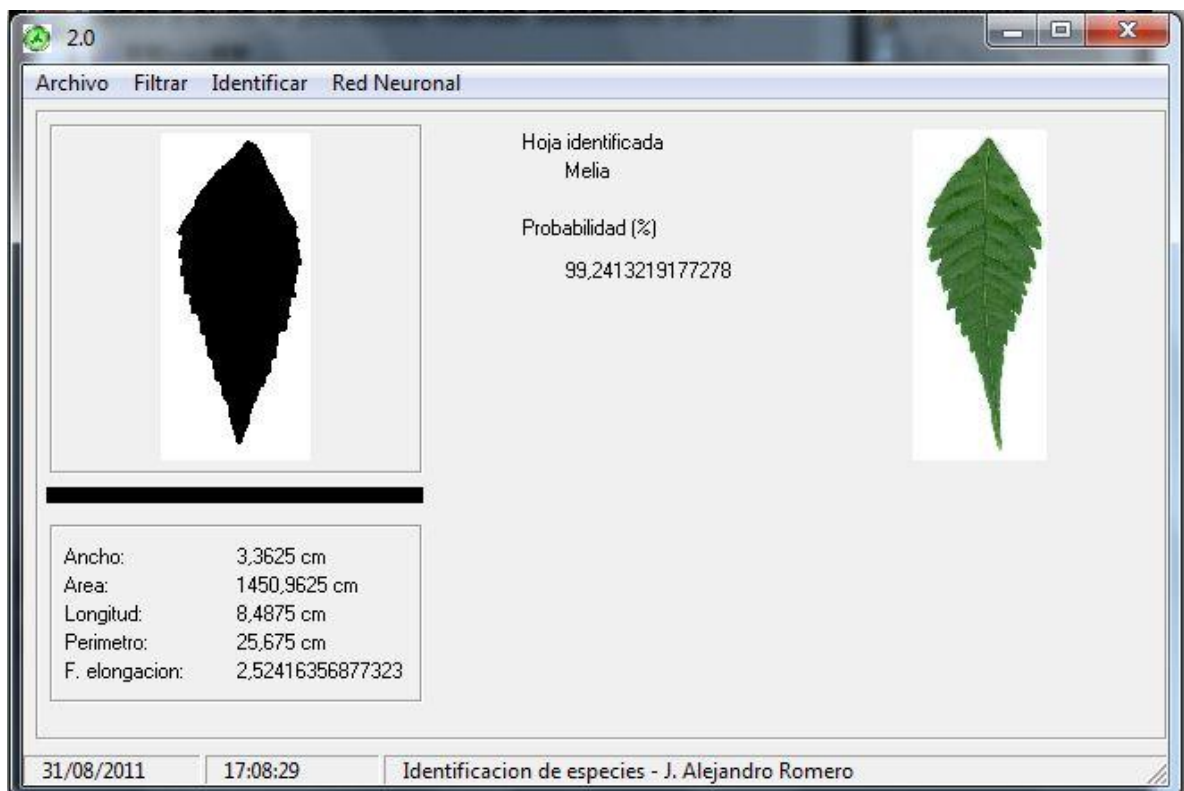


Figura 5.11 Hoja etiquetada como Melia_H1_007

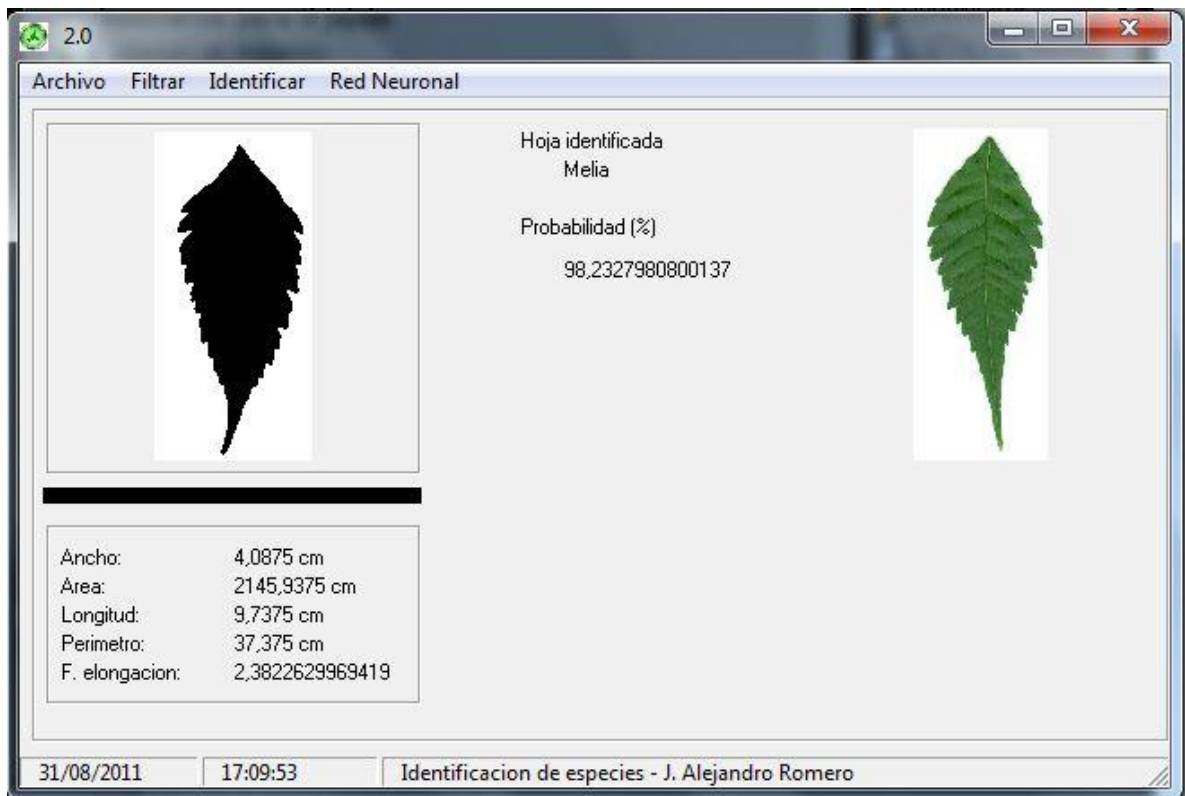


Figura 5.12 Hoja etiquetada como Melia_H1_009

5.4. Pruebas sobre hoja Melissa

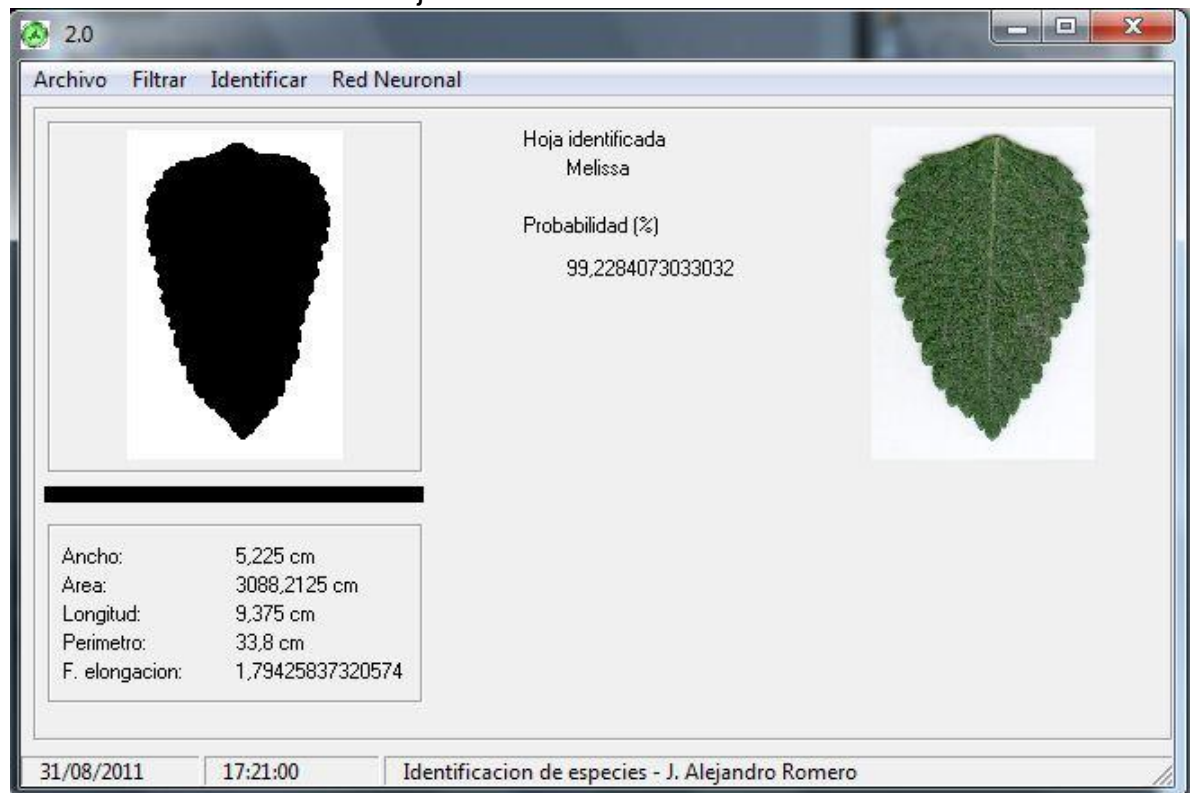


Figura 5.13 Hoja etiquetada como Melissa_H1_003

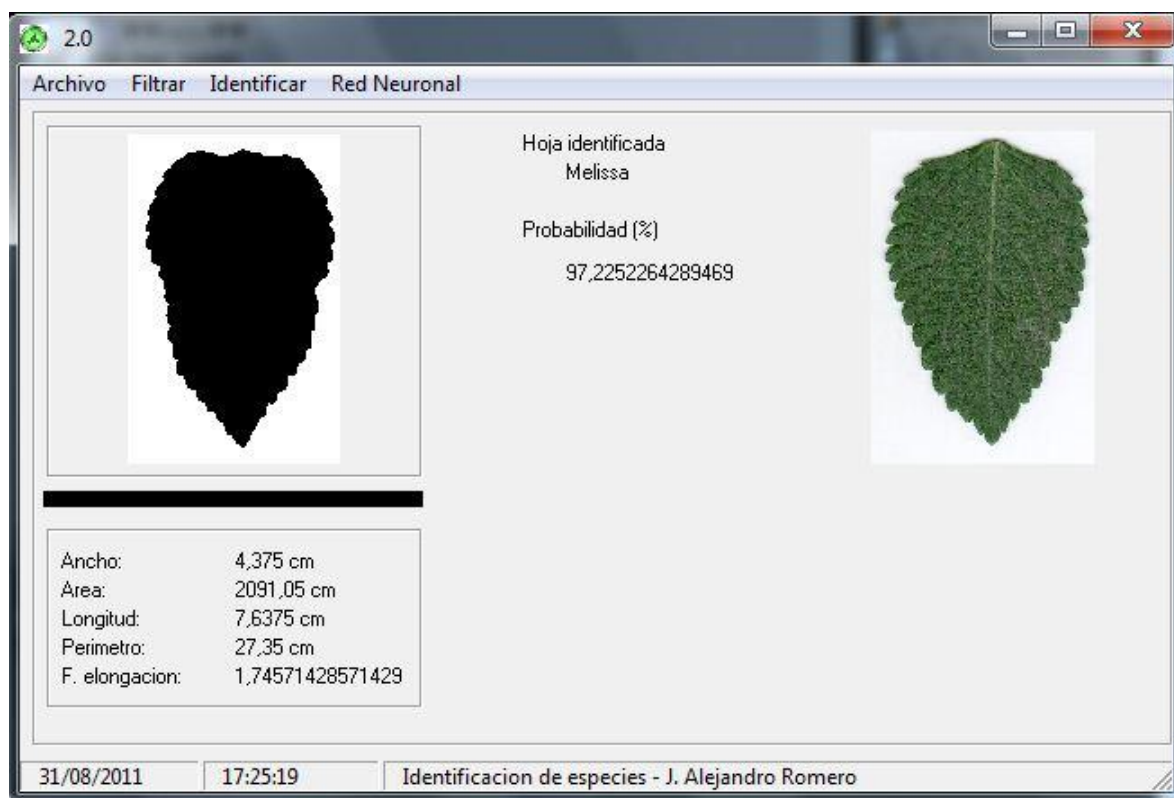


Figura 5.14 Hoja etiquetada como Melissa_H1_006

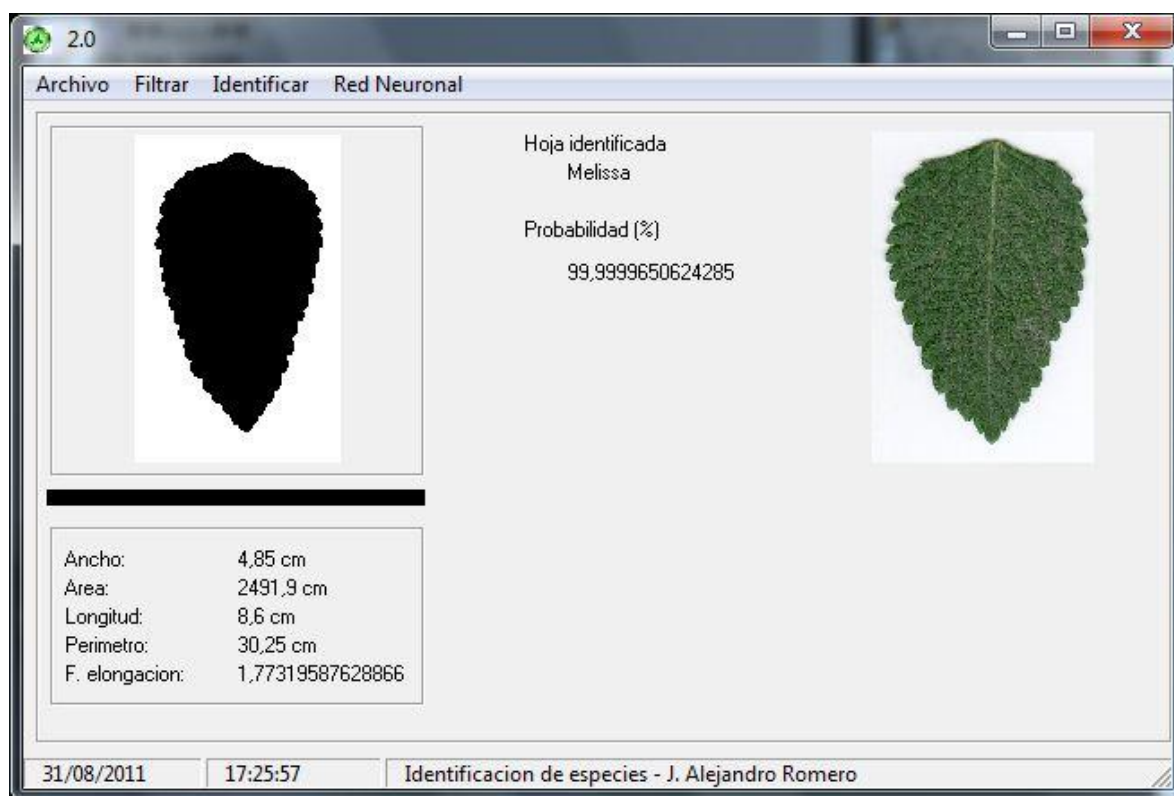


Figura 5.15 Hoja etiquetada como Melissa_H1_007

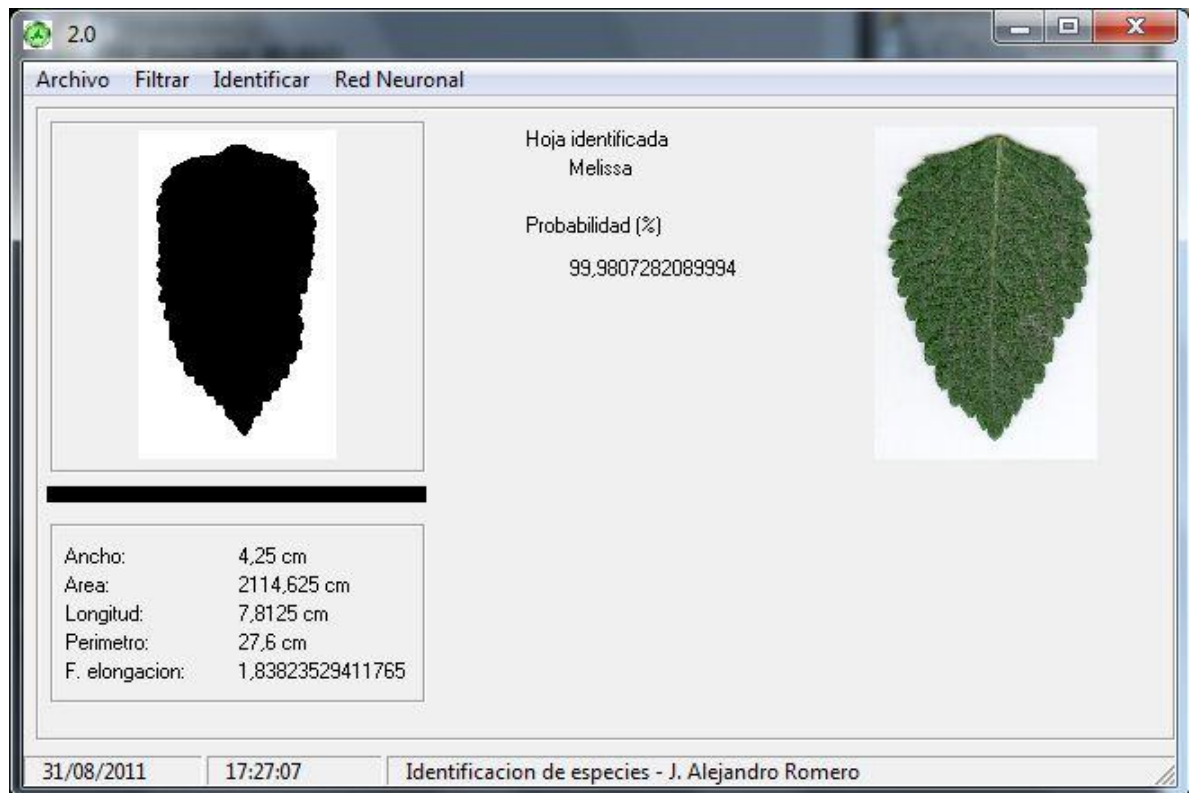


Figura 5.16 Hoja etiquetada como Melissa_H1_007

5.5. Pruebas sobre hoja Guayaba

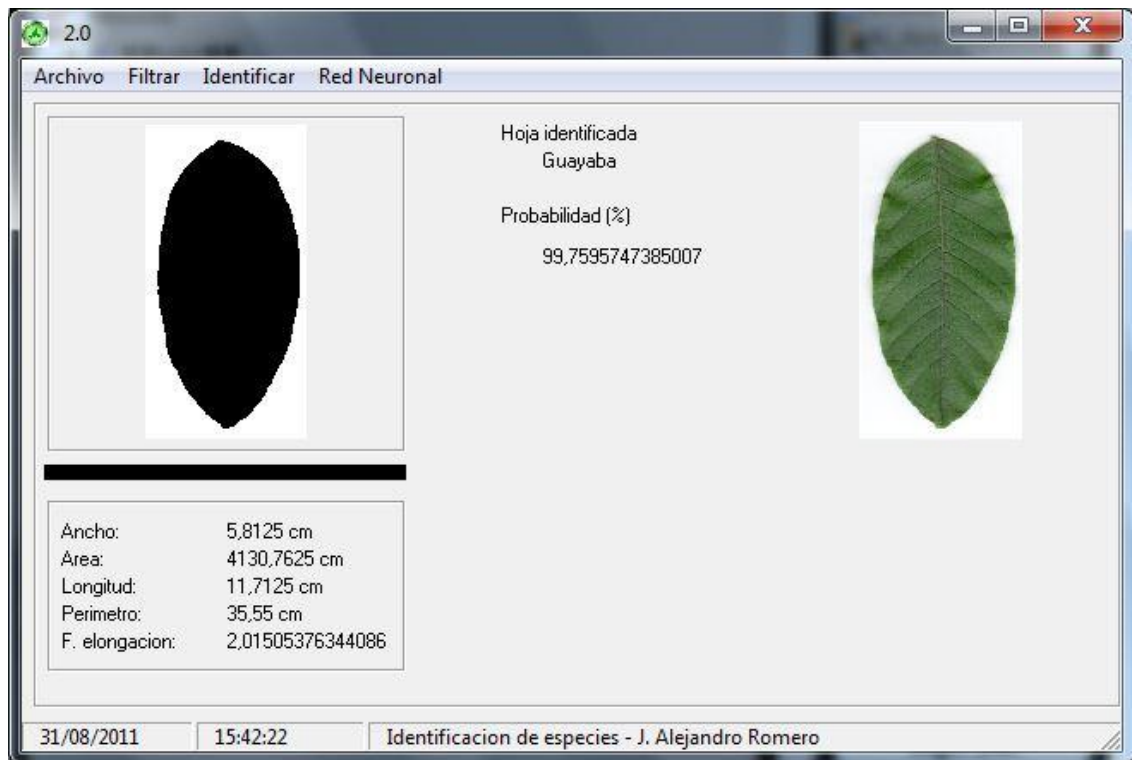


Figura 5.17 Hoja etiquetada como Guayaba_H1_002

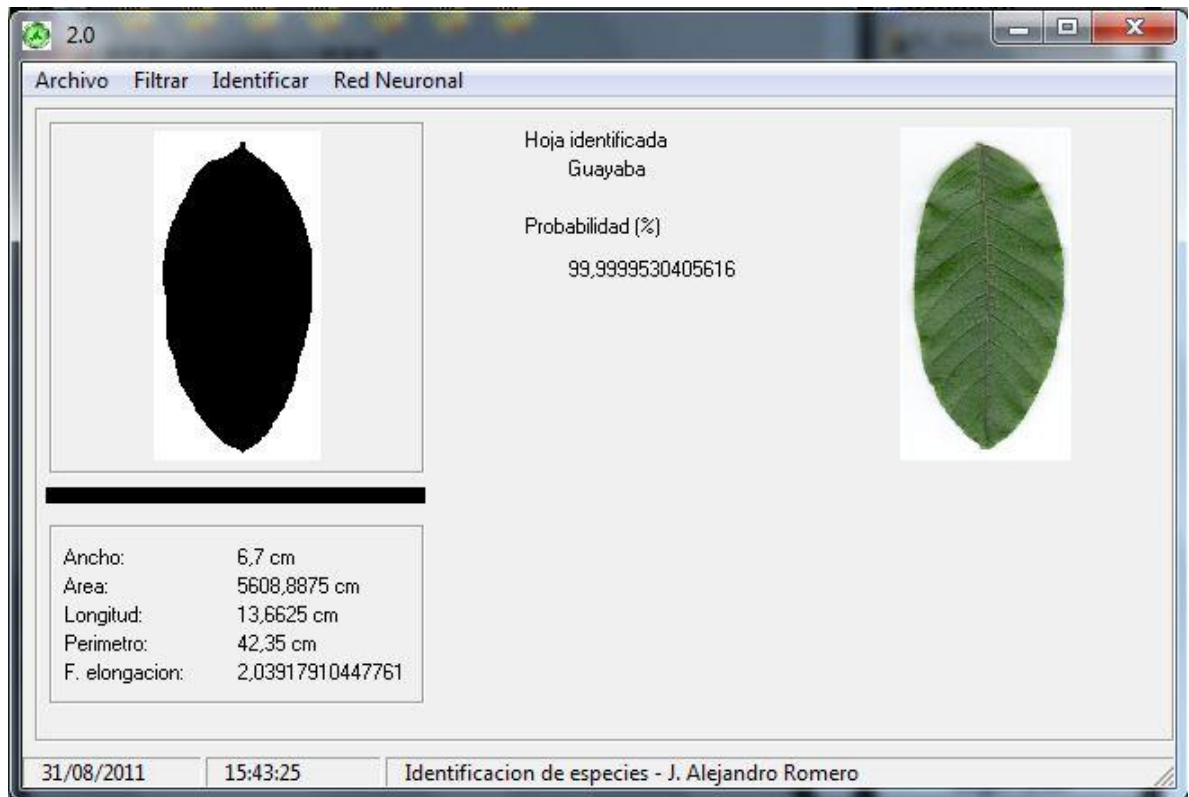


Figura 5.18 Hoja etiquetada como Guayaba_H1_004

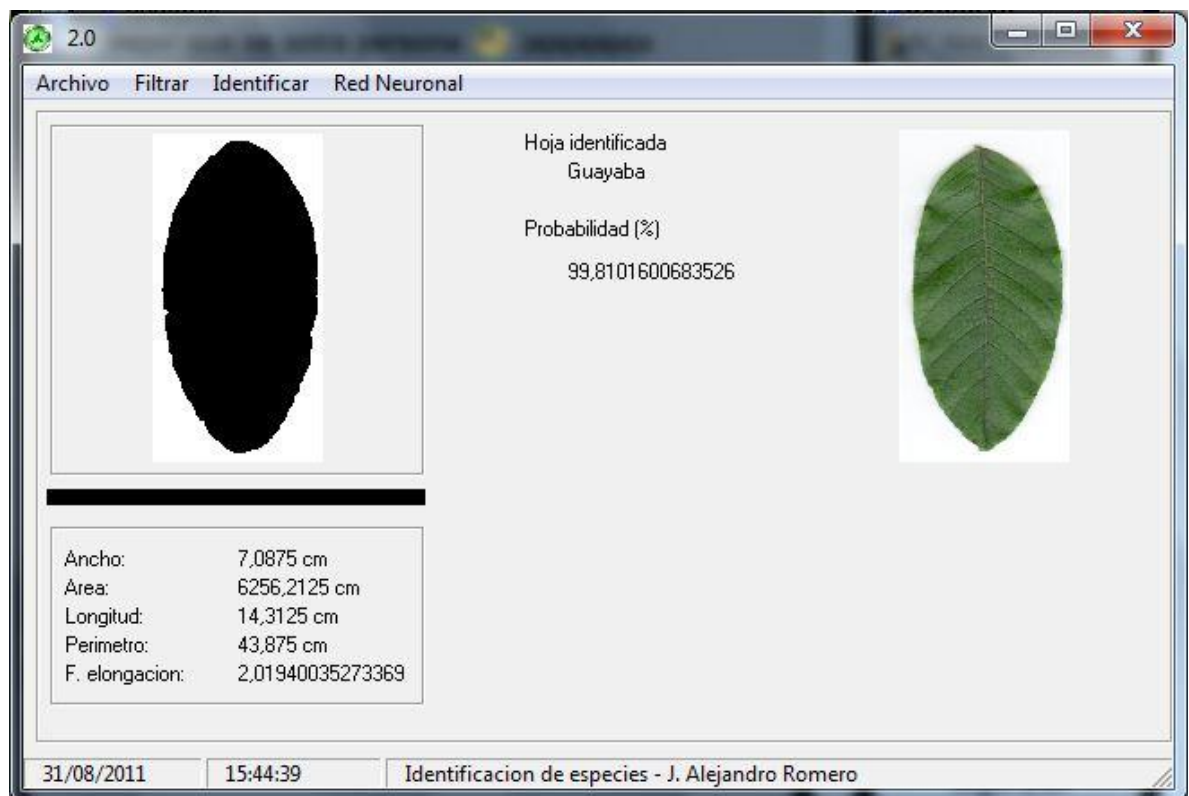


Figura 5.19 Hoja etiquetada como Guayaba_H1_005

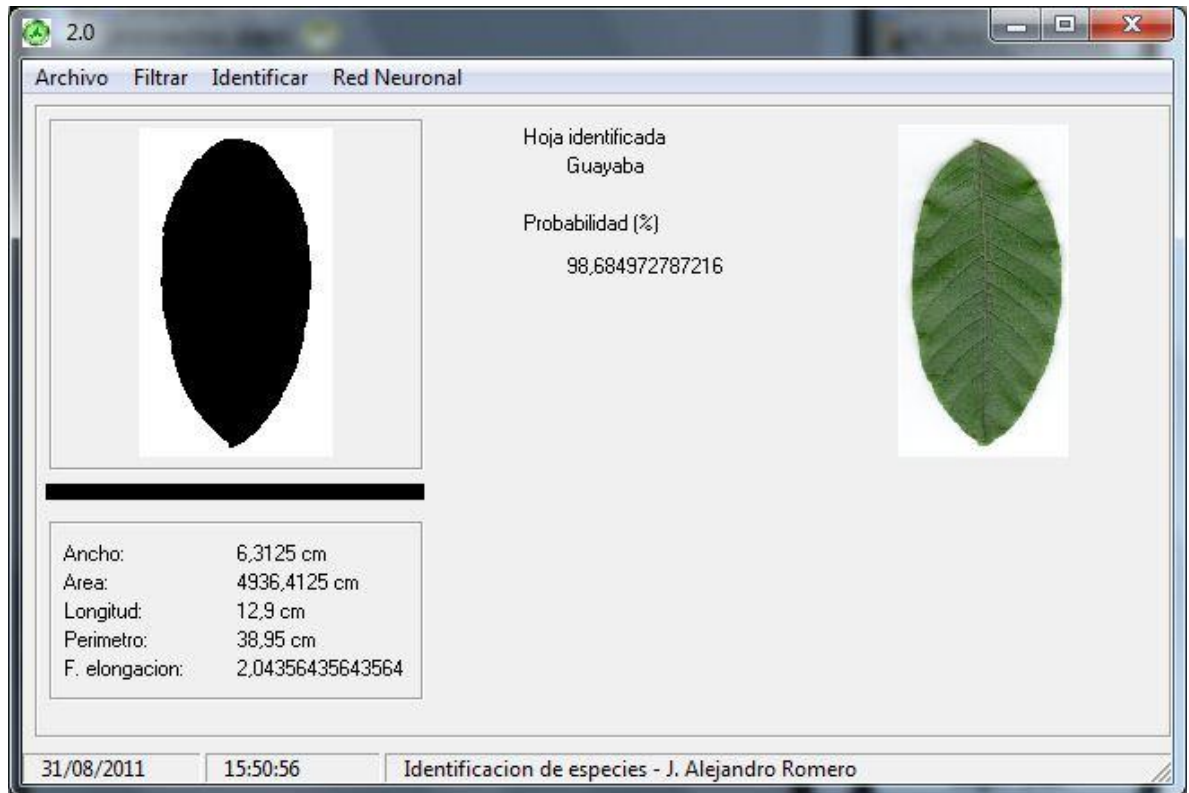


Figura 5.20 Hoja etiquetada como Guayaba_H1_036

Así como el programa ha tenido resultados favorables para identificar las especies, también ha tenido algunos resultados no adecuados sobre algunas especies (clasificando la especie de forma incorrecta, debido a la densidad de área que provoca similitud entre los parámetros de las especies), con los cuales se podría trabajar a futuro para poder corregir de alguna forma este tipo de errores y se pueda ser más precisa la identificación.

Aunque para el ojo humano las especies son muy distintivas, para el programa es complicado hacer esa distinción, ya que el cálculo está basado en la densidad de área, longitud, ancho, y perímetro de la especie; por esta razón en algunos casos en los que las hojas carecen de perfección, y presentan similitudes en su densidad de área, el programa tiende a confundir estas

similitudes con otra especie puesto que los parámetros manejados por el algoritmo tienden a converger en los de otra especie.

5.6. Descriptor de forma: Firmas de Forma

Como se ha mencionado a lo largo del desarrollo de esta tesis, los métodos empleados permiten distinguir y clasificar algunas especies, y en algunos casos puntuales el algoritmo tiende a confundirlas, aún en casos donde las hojas pueden ser distintivas en su forma. Como método adicional y propuesta en desarrollo se pretende implementar un descriptor de forma, el cual nos permita tener una posibilidad más para distinguir esas especies que fueron confundidas; como propuesta de descriptor se pretende usar las firmas de forma. Dichas firmas representan o asocian una forma con una función unidimensional derivada de los puntos del borde. Existen muchas firmas de este tipo como: distancia del centroide $r(t)$, coordenadas complejas, ángulo tangente, ángulo acumulativo, curvatura, y área, entre otros, siendo la primera la elegida (ver fig. 5.23).

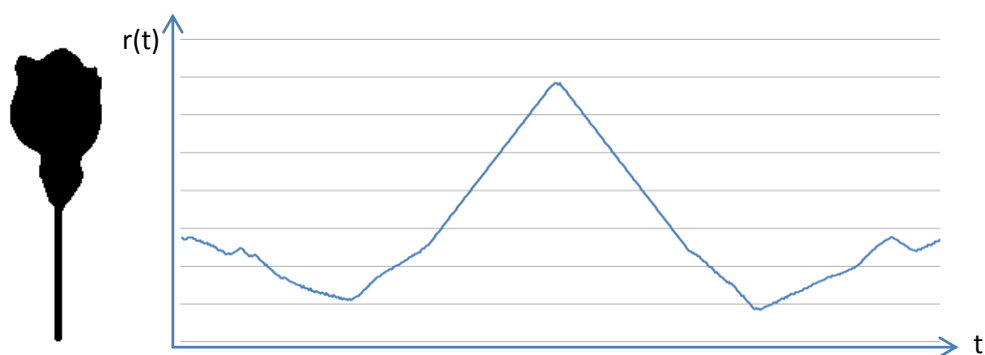


Figura 5.23 Gráfica de la distancia del centroide hacia los puntos del contorno de una rosa.

Debido a que los objetos del mundo son tridimensionales, al observar un objeto desde diferentes posiciones u orientaciones en el espacio, se pueden formar diferentes proyecciones en dos dimensiones, con lo cual sería ideal

tener un descriptor capaz de detectar estos cambios. Como ya se ha mencionado con este método se pretende aumentar la posibilidad de discriminación por parte del programa, ya que las firmas de forma nos permiten obtener más datos importantes sobre el contorno de la especie. A continuación se presentan algunas firmas de forma de las especies para reconocer los patrones distintivos que exhiben las especies.

Firmas de forma de la especie durazno

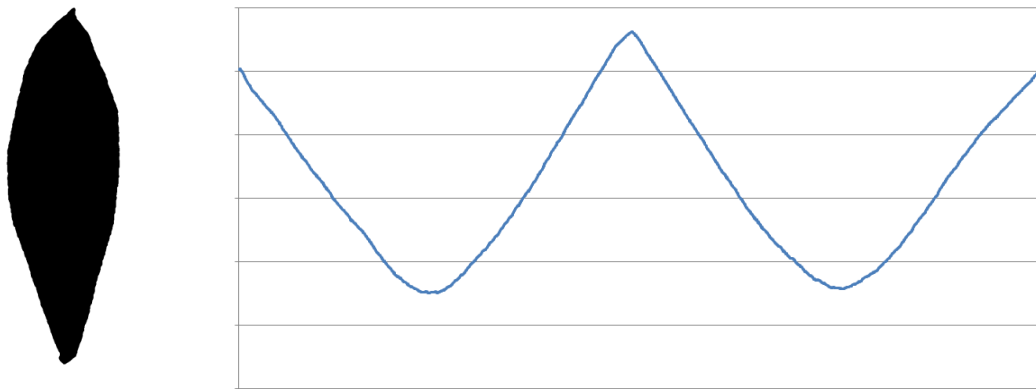


Figura 5.24 Firma de forma de la hoja etiquetada como Durazno_H1_021

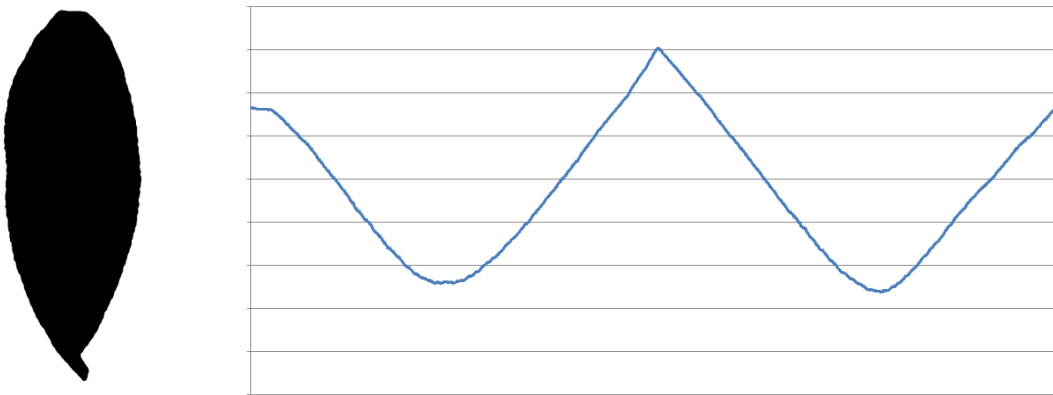


Figura 5.25 Firma de forma de la hoja etiquetada como Durazno_H1_045

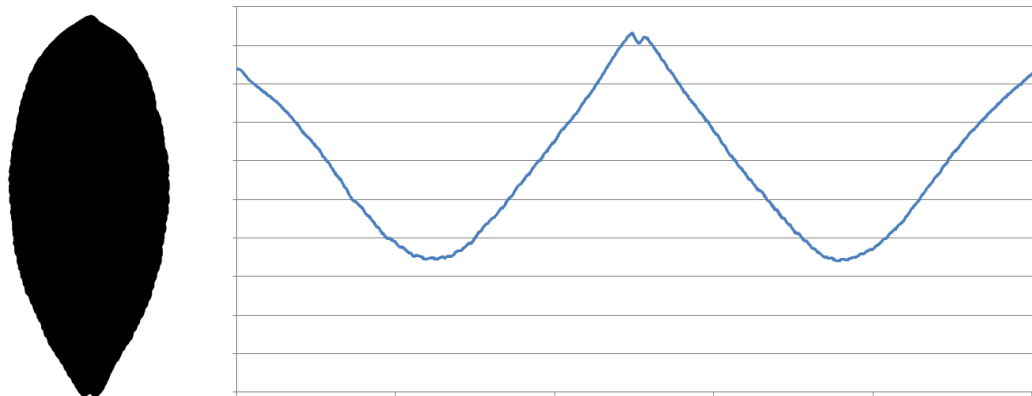


Figura 5.26 Firma de forma de la hoja etiquetada como Durazno_H1_050

Firmas de forma de la especie etiquetada como Flama

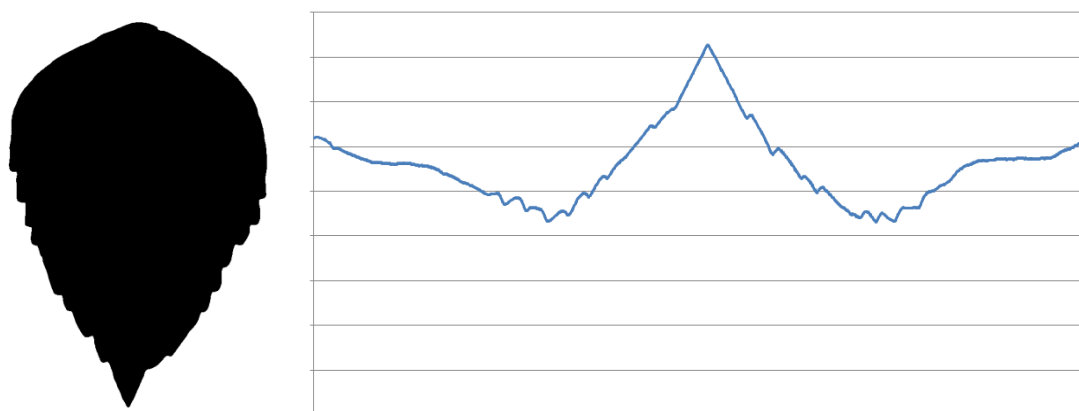


Figura 5.27 Firma de forma de la hoja etiquetada como Flama_H1_001

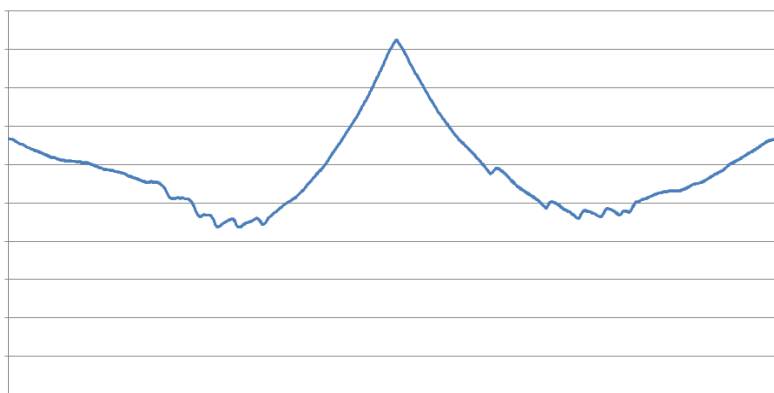


Figura 5.28 Firma de forma de la hoja etiquetada como Flama_H1_004

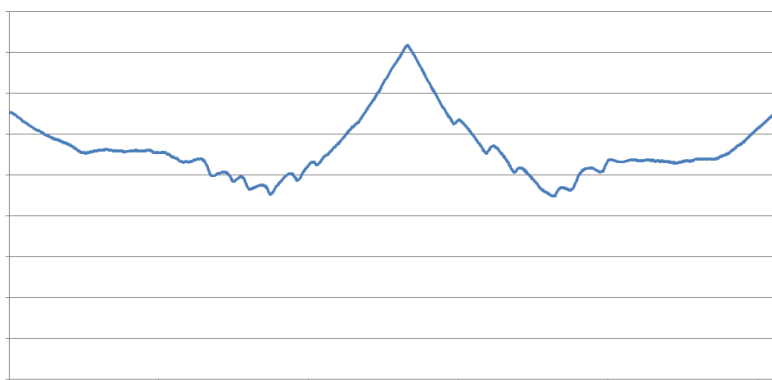


Figura 5.29 Firma de forma de la hoja etiquetada como Flama_H1_008

Firmas de forma de la especie Hedera

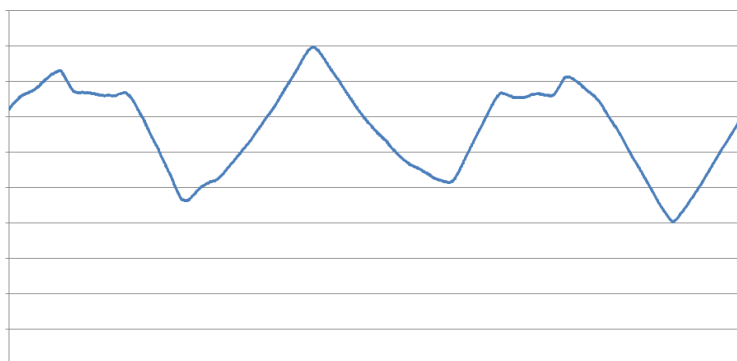
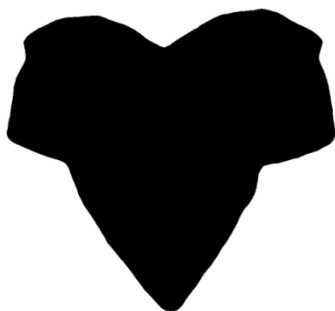


Figura 5.30 Firma de forma de la hoja etiquetada como Hedera_H1_003

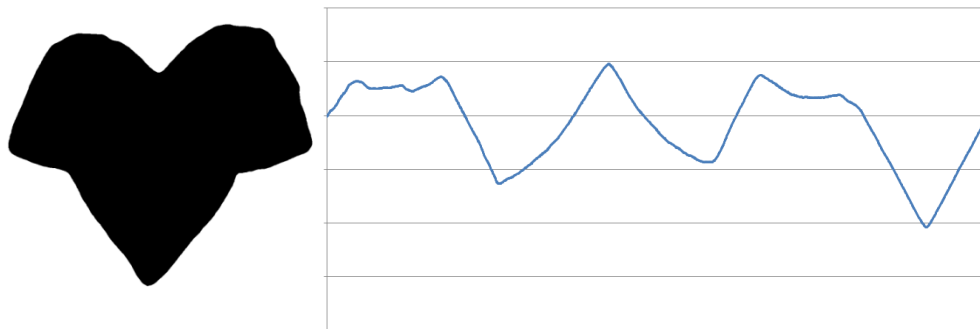


Figura 5.31 Firma de forma de la hoja etiquetada como Hedera_H1_006

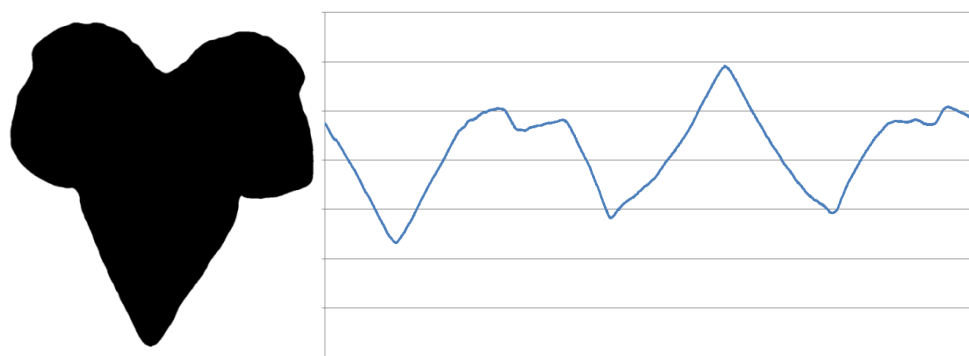


Figura 5.32 Firma de forma de la hoja etiquetada como Hedera_H1_010

Firmas de forma de la especie Malva

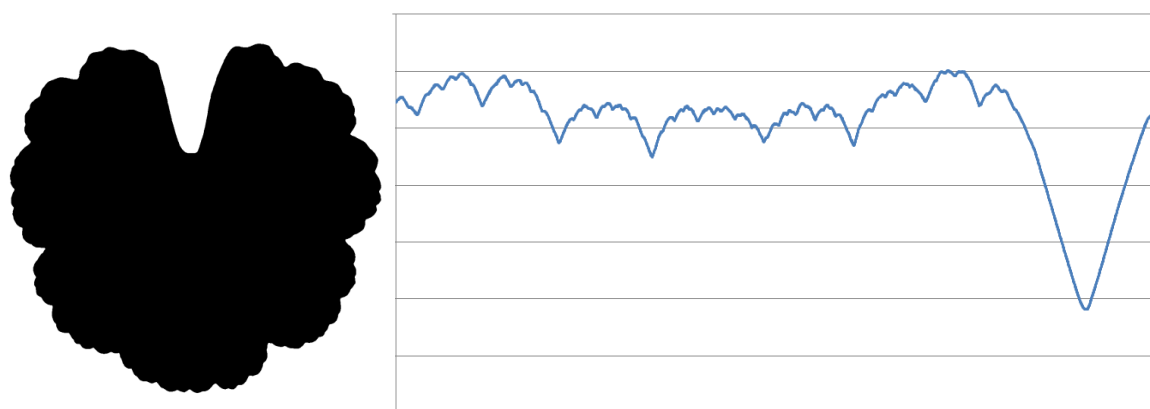


Figura 5.33 Firma de forma de la hoja etiquetada como Malva_H1_006

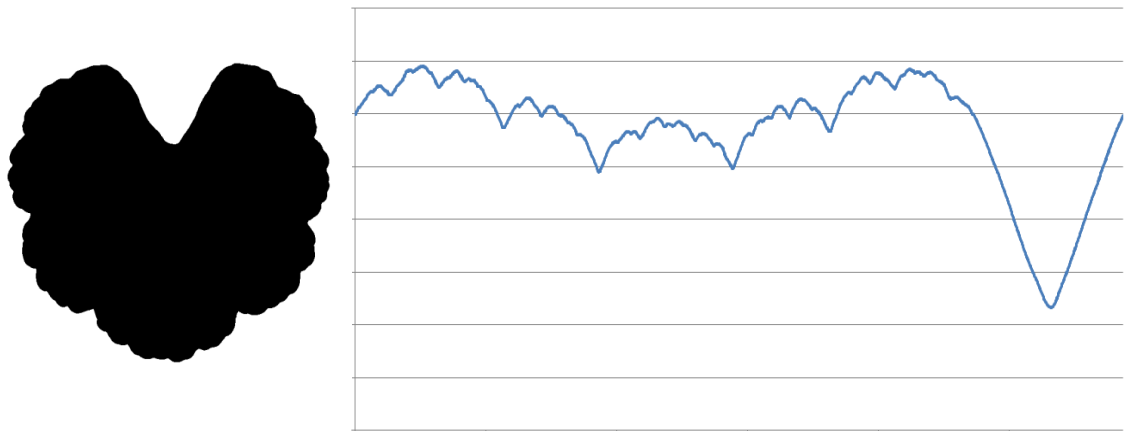


Figura 5.34 Firma de forma de la hoja etiquetada como Malva_H1_027

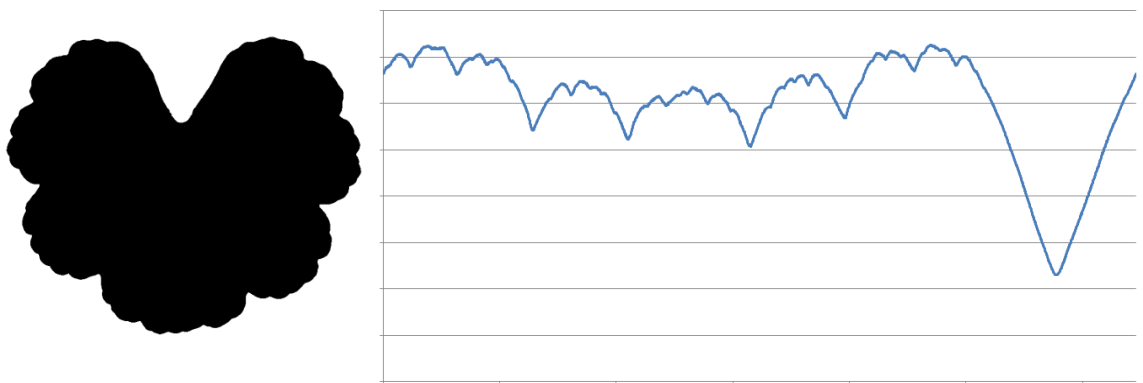


Figura 5.35 Firma de forma de la hoja etiquetada como Malva_H1_050

Como puede apreciarse en las figuras anteriores, las firmas de forma ofrecen mucha información; la futura implementación de dicho descriptor de forma debe hacer una adecuada discriminación, por lo cual sería conveniente realizar un estudio estadístico para establecer un rango de variación, y de esta forma construir una base de datos que sirva como referencia.

6. CONCLUSIONES

Con los métodos y técnicas del procesamiento de imágenes implementadas se buscó desarrollar un programa que pudiera evaluar y clasificar las especies, basado en un algoritmo, que a partir de una imagen en niveles de gris obtenida bajo condiciones de iluminación de un escáner genérico, analizara diversos parámetros como: la excentricidad, la circularidad, la rectangularidad, la relación entre perímetro y longitud, la relación entre el largo y el ancho, la relación entre el perímetro y la longitud en adición con el ancho.

Usando los parámetros distintivos que caracterizaban a estas hojas, se realizó una observación de los intervalos de variación de los resultados, que permitiera obtener datos fiables, y que en combinación en distintos procesos mostrara información significativa de la especie, permitiendo construir una base de datos. Debido a las variaciones de las características entre las especies en estudio fue necesario para cada especie generar sus propios pesos que permitieran encontrar un valor único y representativo, los pesos se usaron en una sola red neuronal para todas las especies, de las cuales fueron consideradas para las pruebas y aplicaciones 13 especies distintas, que se encontraban alrededor de la facultad de ingeniería, tomando varias muestras de hojas por especie.

Durante el desarrollo de la tesis se buscó la posibilidad de que el programa a futuro pudiera ser actualizado y entrenado para clasificar nuevos especímenes, el cual permitiera el crecimiento en la base de datos así como el incremento en el aprendizaje, y pudiesen ser aplicados en áreas distintas.

LITERATURA CITADA

1. Arber, A. (1987). *Herbals. Their Origin and Evolution. A Chapter in the History of Botany 1470-1670*. Cambridge University Press, New York., 358 p. ISBN 0-521-33879-4.
2. Askevold, I.S., and O'Brien, C.W., (1994). DELTA, an invaluable computer program for generation of taxonomic monographs. *Ann. Entomol. Soc. Am.* 87: 1–16.
3. Astegiano ED, Favaro JC, Bouzo CA, R(2000). Estimación del área foliar en distintos cultivares de tomate (*Lycopersicon esculentum* Mill.) utilizando medidas foliares lineales, Universidad Nacional del Litoral. Kreder 2805.
4. Brickell, Christopher. *A-Z Encyclopedia of Garden Plants*. London: Dorling Kindersley Publishers Ltd. 2003.
5. Bylesjö M, Segura V, Soolanayakanahally RY, Rae AM, Trygg J, Gustafsson P, Jansson S, Street NR (2008). LAMINA: a tool for rapid quantification of leaf size and shape parameters. *BMC Plant Biol.* 8(1).

6. Daughtry C, (1990). Direct measurements of Canopy Structure. *Rem. Sens. Rev.* 5(1): 45-60.
7. Femat-Díaz A., Vargas-Vázquez D., Huerta-Manzanilla E., Rico-García E., Herrera-Ruiz G., (2011). Scanner image methodology (SIM) to measure dimensions of leaves for agronomical applications, *African Journal of Biotechnology*, ISSN 1684-5315.
8. Freeman JA, Skapura DM, *Neural Networks Algorithms, Applications, and Programming Techniques*, 1991, Addison-Wesley Publishing Company.
9. Haykin S. *Neural Networks A comprehensive foundation*, Second Edition, 2001, Pearson Education, pp 43-60.
10. Hu, MK(1962). Visual Pattern Recognition by Moment Invariants, *IRE Trans. Info Theory*, vol IT-8, pp. 179-187.
11. Jaramillo MA, Fernández JA, Salazar EM, Filtro de mediana sobre redes neuronales.
12. Kuoh Cs, Song H R(2005). Interactive Key to Taiwan Grasses Using Characters of Leaf Anatomy – The ActKey Approach, 50(4): 261-271.

13. McCulloch, W. S. "A logical calculus of the ideas immanent in nervous activity". Bull Math. Biophysics, 1943, 5, 110- 115.
14. Neelakanta, P., DeGroff, D., (1994) Neuronal Network Modeling, CRC, Boca Raton.
15. O'Neal ME, Landis DA, Isaacs R(2002). An Inexpensive, accurate method for measuring leaf area and defoliation through digital image analysis, J. Econ. Entomol. 95(6): 1190-1194.
16. Olabe XB (2008). Redes Neuronales Artificiales y sus Aplicaciones, eduCommons, pp 3-5.
17. Otsu NA (1979). Threshold selection method from gray-level histograms, IEEE Trans. Syst. Man. Cybern. 9(1): 62–66.
18. Russel SJ. INTELIGENCIA ARTIFICIAL UN ENFOQUE MODERNO, Segunda edición, 2004, Pearson Prentice Hall.
19. Soille P (2003). Morphological Image Analysis, Principles and Applications, 2nd ed., Germany: Verlag Berlin Heidelberg: Springer.

20. Steinbach, Gunter y Hofmann, Reinhold y Held. Arboles. Barcelona: Blume 2003
21. Young, T.Y., Fu, K.-S., Handbook of pattern recognition and image processing, Academic Press Publishers (1986) ISBN: 0127745602
22. Website: El mundo de las plantas. (1999-2012). <http://www.botanical-online.com/botanica2.htm>. Accedido por última vez en Enero de 2012.
23. Website: Infojardin. (1999-2012). <http://www.infojardin.com/>. Accedido por última vez en Diciembre de 2011.