



UNIVERSIDAD AUTÓNOMA DEL ESTADO DE MÉXICO
CENTRO UNIVERISTARIO UAEM TEXCOCO



SISTEMA DE IDENTIFICACIÓN TRIVIAL Y COMPLEJA
DE PLANTAS: UN ANÁLISIS COMPARATIVO

T E S I S

QUE PARA OBTENER EL GRADO DE:
MAESTRO EN CIENCIAS DE LA COMPUTACIÓN

PRESENTA:

JESÚS TALTEPA MORENO

TUTOR ACADÉMICO:

DR. EN C. JAIR CERVANTES CANALES

TUTORES ADJUNTOS:

DR. EN C. C. FARID GARCÍA LAMONT

M. EN C. C. A. JOSÉ SERGIO RUÍZ CASTILLA

TEXCOCO, ESTADO DE MÉXICO, SEPTIEMBRE DE 2015.



DICTÁMEN DE AUTORIZACIÓN Y OBTENCIÓN DE GRADO DE MAESTRÍA

Texcoco, Méx. , a 17 de Agosto del 2015

Título del proyecto:

Sistema de identificación trivial y compleja de plantas: Un análisis comparativo

Tesista:

Jesús Taltempa Moreno

Dictamen:

No. de revisión: 4

- Rechazado
- Sujeto a modificaciones
- Aceptado, condicionado
- Aceptado

**POSGRADO
TITULACION
RECIBIDO**
Por Adriana Arévalo
Texcoco, Méx., a 11 de 09 del 2015

Observaciones generales:

Aceptado para la impresión
Aceptado para la defensa de grado

Tutor Adjunto	Tutor Académico	Tutor Adjunto
M. en C. C. A José Sergio Ruiz Castilla	Dr. en C. Jair Cervantes Canales	Dr. en C. Com . Farid García Lamont



Agradecimientos

A mis padres Zeferino y Guadalupe quiero agradecerles con todo el amor que mi corazón puede darles porque me han apoyado en todo momento, por su motivación constante, sus valores y por sus consejos que me han permitido ser una persona de bien. Gracias padres míos, no me alcanzara la vida para demostrarles lo orgulloso que estoy de ustedes.

A mi hermano Miguel Levi quien ha sido un gran compañero durante toda mi vida, con quien he vivido tanto y me ha enseñado a reír, jugar, compartir y a levantarme para seguir adelante.

A mi hermano Alejandro quien no vivió para verlo, pero con quien siempre estaré agradecido porque me enseñó a ver la vida de un modo distinto, y fue un claro ejemplo de fortaleza para toda mi familia.

A mi hermana Miriam quien siempre me ha apoyado incondicionalmente.

Agradezco al Dr. en C. Jair Cervantes Canales quien fue mi Tutor académico y me apoyo con su dirección durante la elaboración de esta investigación.

Al Dr. en C. Farid García Lamont, al M. en C. C. José Sergio Ruiz Castilla por sus observaciones y orientación académica hacia este trabajo.

Agradezco a la UAEM y CONACyT por la beca proporcionada para realizar mis estudios de posgrado y a COMECyT por la beca de Titulación.

Índice general

1. Introducción	1
1.1. Problemática	2
1.2. Justificación	2
1.3. Objetivos y metas	3
1.3.1. Objetivo general	3
1.3.2. Objetivos particulares	3
1.4. Hipótesis	4
1.5. Metodología	4
1.6. Estado del arte	5
2. Preliminares	11
2.1. Métodos de segmentación	12
2.1.1. Método de Otsu	12
2.1.2. El método PCA	13
2.1.3. Frontera Adaptativa	14
2.2. Extracción de características geométricas	14
2.3. Extracción de características texturales	49
2.3.1. Matriz de co-ocurrencia	49
2.3.2. Descriptores Haralick	51
2.4. Extracción de características cromáticas	56
2.4.1. Fundamentos de color	57
2.4.2. Modelos de color	57

2.4.3.	Descriptores de color	62
3.	Clasificadores	68
3.1.	Clasificadores estadísticos	69
3.1.1.	Clasificación Bayesiana	69
3.1.2.	Naive Bayes	70
3.2.	Redes Neuronales	70
3.2.1.	Elementos de una red neuronal.	71
3.2.2.	El Perceptrón	71
3.2.3.	El Perceptrón Multicapa	74
3.3.	Árboles de decisión	76
3.3.1.	ID3	77
3.3.2.	J48	79
3.3.3.	<i>Random Forest</i>	80
3.4.	Clasificación basada en SVM	80
3.4.1.	Caso linealmente separable	82
3.4.2.	Caso linealmente no separable	83
3.4.3.	SMO (Sequential Minimal Optimization)	85
4.	Algoritmos Genéticos	86
4.1.	Elementos de un algoritmo genético	86
4.2.	Algoritmo genético básico	87
4.2.1.	Población inicial	88
4.2.2.	Selección de individuos	88
4.2.3.	Cruza	89
4.2.4.	Mutación	90
4.2.5.	Condición de paro	90
4.2.6.	Ejemplo de un genético básico para la búsqueda de un óptimo	90
5.	Metodología	96
5.1.	Creación de los conjuntos de datos	96

5.2. Segmentación de imágenes	106
5.3. Extracción de características	107
5.4. Clasificación de resultados	111
5.5. Algoritmo genético para reducción de características	113
5.5.1. Algoritmo Genético propuesto	114
6. Resultados experimentales	118
6.1. Técnicas de validación	118
6.1.1. Cross-validation	118
6.1.2. F-Measure	119
6.1.3. Área ROC	121
6.2. Resultados	122
6.2.1. Conjunto de datos Complejo	124
6.2.2. Conjunto de datos Trivial	127
6.2.3. Mejora de resultados con Algoritmo Genético	129
7. Conclusiones	140
7.1. Trabajo futuro	142
7.2. Publicaciones	143
A. Resultados de clasificación del conjunto complejo.	151
B. Resultados de clasificación con Algoritmo Genético	166
C. Programas de apoyo	172
C.1. Programa para el AG básico	172
C.1.1. Manejo del programa	173
D. Artículos Publicados	176

Índice de figuras

1-1. Etapas de la metodología propuesta para este proyecto.	4
2-1. Rectángulo de delimitación mínima de un contorno, basandose en el eje mayor y eje menor.	20
2-2. Relación de Circularidad para los siguientes objetos. (a) Objeto totalmente redondo. Circularidad = 1.001, (b) Objeto rectángulo. Circularidad = 0.7415 . . .	22
2-3. Contorno convexo de una región segmentada	24
2-4. Ejemplos de regiones segmentadas con huecos y componentes conexas. a) Región con dos huecos. b) Región con dos componentes conexas	25
2-5. Representación de función curvatura de un contorno. a) Contorno normalizado a 30 puntos. b) Función curvatura de contorno de a).	27
2-6. Representación grafica de función de área de una región segmentada. a) Región que se forma de dos puntos consecutivos y el centroide. b) Función de área de a).	28
2-7. Ejemplo de cuadrícula adaptativa de una imagen. a) Imagen con cuadrícula adaptativa. b) Árbol de descomposición de la cuadrícula adaptativa de a).	29
2-8. Códigos de cadena tomando en cuenta 4 y 8 vecinos. a) Código de cadena 4-direccional; b) Código de cadena 8-direccional	30
2-9. Numeración correspondiente para el código de cadena de vértice.	31
2-10. Características simbólicas basadas en el eje de menor inercia.	32
2-11. Ejemplo de matriz de forma cuadrada. (a) Región de forma original y su matriz de forma cuadrada. (b) Región de forma resultante de (a) y su matriz de forma cuadrada.	34

2-12. Ejemplo de matriz de forma polar. a) Modelo de forma polar. b) Matriz de forma polar correspondiente a a).	35
2-13. Localización de las bandas de frecuencia en una Transformada Discreta de <i>Wavelets</i> con cuatro bandas. La convención es fila/columna	45
2-14. Relación espacial entre un pixel de referencia y sus pixeles vecinos en base a 4 direcciones. Las celdas 1 y 5 están en la dirección 0° (horizontal); las celdas 4 y 8 en están en la dirección 45° ; las celdas 7 y 3 están en la dirección 90° y las celdas 6 y 2 en la dirección 135°	50
2-15. Ejemplo de matrices de co-ocurrencia de una imagen. a) Matriz de dimensiones 4×4 con 4 niveles de gris (0, 1, 2, 3). b) Posibles combinaciones de la matriz a). c) Matrices de co-ocurrencia de a) en las 4 direcciones posibles (0° , 90° , 135° y 45°).	52
2-16. Tetraedro de color para el modelo <i>RGB</i>	58
2-17. Hexágono de color para el modelo <i>HSV</i>	60
2-18. Modelo de color <i>HLS</i>	61
2-19. Modelo de color <i>HSI</i>	62
2-20. Imagen dividida en cuadros para los niveles $k = 0$ a $k = 2$	65
3-1. Esquema general de una RNA	72
3-2. Ejemplo de separación lineal para dos clases	72
3-3. Modelo de un perceptrón para dos clases.	73
3-4. Modelo de una red neuronal multicapa	75
3-5. Problemas de clasificación que pueden ser resueltos con una RNA multicapa con dos capas.	76
3-6. Árbol de decisión para determinar si es un día apropiado para jugar Tennis . . .	77
3-7. Hiperplano que separa a dos clases	81
3-8. Clases linealmente separables.	82
3-9. Clases linealmente no separables	83
3-10. Uso de un Kernel para transformación de un espacio de datos.	84
4-1. Cruza de dos cadenas binarias y sus descendientes correspondientes.	89

4-2. Mutación de una cadena binaria	90
5-1. Etapas para la metodología propuesta.	97
5-2. Ejemplo de imágenes de hojas complejas y triviales a) Imágenes denominadas complejas (muy similares entre sí). b) Imágenes denominadas triviales (muy distintas entre sí).	98
5-3. Ejemplo de imágenes asociadas al conjunto de hojas trivial.	99
5-4. Razón de similitud de subconjuntos complejos 1-6	100
5-5. Razón de similitud de subconjuntos complejos 7-11	101
5-6. Subconjunto complejo 1	102
5-7. Subconjunto complejo 2	102
5-8. Subconjunto complejo 3	102
5-9. Subconjunto complejo 4	103
5-10. Subconjunto complejo 5	103
5-11. Subconjunto complejo 6	103
5-12. Subconjunto complejo 7	104
5-13. Subconjunto complejo 8	104
5-14. Subconjunto complejo 9	104
5-15. Subconjunto complejo 10	105
5-16. Subconjunto complejo 11	105
5-17. Imágenes de hoja de planta fácil de segmentar.	106
5-18. Imágenes de hoja de planta difícil de segmentar.	107
5-19. Ejemplo de resultados de clasificación con estructura de un fichero arff correspondiente a los resultados del subconjunto complejo 6 con segmentación OTSU.	112
5-20. Curso de la dimensionalidad.	114
5-21. Adaptación de un conjunto de características con una cadena binaria de un AG.	115
5-22. Conjunto de cadenas binarias y las precisiones obtenidas de un clasificador	116
5-23. Ejemplo de dos iteraciones en el AG propuesto para reducción de características.	116
6-1. Validación cruzada de k iteraciones	119

6-2. Representación de similitud entre dos clases. El punto de corte t determina el comportamiento del clasificador.	121
6-3. Curva ROC	122
6-4. Tipos de curvas ROC	123
6-5. Representación gráfica del aumento de precisión promedio al disminuir características con el AG. Los resultados se basan en segmentación Otsu y características cromáticas, texturales y geométricas.	132
6-6. Representación gráfica del aumento de precisión promedio al disminuir características con el AG. Los resultados se basan en segmentación Otsu y características cromáticas y texturales.	132
6-7. Representación gráfica del aumento de precisión promedio al disminuir características con el AG. Los resultados se basan en segmentación Otsu y características cromáticas y geométricas.	133
6-8. Representación gráfica del aumento de precisión promedio al disminuir características con el AG. Los resultados se basan en segmentación Otsu y características cromáticas.	133
6-9. Representación gráfica del aumento de precisión promedio al disminuir características con el AG. Los resultados se basan en segmentación Otsu y características texturales y geométricas.	134
6-10. Representación gráfica del aumento de precisión promedio al disminuir características con el AG. Los resultados se basan en segmentación Otsu y características texturales.	134
6-11. Representación gráfica del aumento de precisión promedio al disminuir características con el AG. Los resultados se basan en segmentación Otsu y características geométricas.	135
6-12. Representación gráfica del aumento de precisión promedio al disminuir características con el AG. Los resultados se basan en segmentación PCA y características cromáticas, texturales y geométricas.	135

6-13. Representación gráfica del aumento de precisión promedio al disminuir características con el AG. Los resultados se basan en segmentación PCA y características cromáticas y texturales.	136
6-14. Representación gráfica del aumento de precisión promedio al disminuir características con el AG. Los resultados se basan en segmentación PCA y características cromáticas y geométricas.	136
6-15. Representación gráfica del aumento de precisión promedio al disminuir características con el AG. Los resultados se basan en segmentación PCA y características cromáticas.	137
6-16. Representación gráfica del aumento de precisión promedio al disminuir características con el AG. Los resultados se basan en segmentación PCA y características texturales y geométricas.	137
6-17. Representación gráfica del aumento de precisión promedio al disminuir características con el AG. Los resultados se basan en segmentación PCA y características texturales.	138
6-18. Representación gráfica del aumento de precisión promedio al disminuir características con el AG. Los resultados se basan en segmentación PCA y características geométricas.	138
C-1. Interfaz del AG propuesto programado en Java.	173
C-2. Ejemplo de datos resultantes después de finalizar el AG.	174

Índice de tablas

2-1. Propiedades de los Descriptores de Fourier	43
2-2. Vectores base de Haar y Daubechies para filtrado pasa bajo y pasa alto	44
4-1. Expresiones que se utilizan en la genética con su estructura equivalente en un algoritmo genético	87
5-1. Tipo de características geométricas y número de características asociadas	108
5-2. Tipo de características texturales y número de características asociadas	108
5-3. Tipo de características cromáticas y número de características asociadas	109
5-4. Combinaciones de las técnicas de segmentación con los tipos de características extraídas en cada conjunto de hojas	109
6-1. Estructura de la Matriz de Confusión	120
6-2. Resultados del desempeño de clasificación, F-Measure y Área-ROC con cada uno de los clasificadores para el conjunto Trivial	128
6-3. Reducción de características promedio en cada combinación de características después de aplicar el Algoritmo Genético	130
6-4. Precisión promedio alcanzada en cada combinación de características y tipo de segmentación	130
6-5. Características más eliminadas y utilizadas por el Algoritmo Genético para obtener mejores precisiones	139

A-1. Resultados del desempeño de clasificación, F-Measure y Área-ROC con cada uno de los clasificadores para los 11 subconjuntos del conjunto complejo. Los resultados se basan en segmentación basada en Otsu y características Cromáticas, Texturales y Geométricas. 152

A-2. Resultados del desempeño de clasificación, F-Measure y Área-ROC con cada uno de los clasificadores para los 11 subconjuntos del conjunto complejo. Los resultados se basan en segmentación basada en Otsu y características Cromáticas y Texturales 153

A-3. Resultados del desempeño de clasificación, F-Measure y Área-ROC con cada uno de los clasificadores para los 11 subconjuntos del conjunto complejo. Los resultados se basan en segmentación basada en Otsu y características Cromáticas y Geométricas. 154

A-4. Resultados del desempeño de clasificación, F-Measure y Área-ROC con cada uno de los clasificadores para los 11 subconjuntos del conjunto complejo. Los resultados se basan en segmentación basada en Otsu y características Cromáticas, 155

A-5. Resultados del desempeño de clasificación, F-Measure y Área-ROC con cada uno de los clasificadores para los 11 subconjuntos del conjunto complejo. Los resultados se basan en segmentación basada en Otsu y características Texturales y Geométricas. 156

A-6. Resultados del desempeño de clasificación, F-Measure y Área-ROC con cada uno de los clasificadores para los 11 subconjuntos del conjunto complejo. Los resultados se basan en segmentación basada en Otsu y características Texturales 157

A-7. Resultados del desempeño de clasificación, F-Measure y Área-ROC con cada uno de los clasificadores para los 11 subconjuntos del conjunto complejo. Los resultados se basan en segmentación basada en Otsu y características Geométricas. 158

A-8. Resultados del desempeño de clasificación, F-Measure y Área-ROC con cada uno de los clasificadores para los 11 subconjuntos del conjunto complejo. Los resultados se basan en segmentación basada en PCA y características Cromáticas, Texturales y Geométricas. 159

A-9. Resultados del desempeño de clasificación, F-Measure y Área-ROC con cada uno de los clasificadores para los 11 subconjuntos del conjunto complejo. Los resultados se basan en segmentación basada en PCA y características Cromáticas y Texturales 160

A-10. Resultados del desempeño de clasificación, F-Measure y Área-ROC con cada uno de los clasificadores para los 11 subconjuntos del conjunto complejo. Los resultados se basan en segmentación basada en PCA y características Cromáticas y Geométricas. 161

A-11. Resultados del desempeño de clasificación, F-Measure y Área-ROC con cada uno de los clasificadores para los 11 subconjuntos del conjunto complejo. Los resultados se basan en segmentación basada en PCA y características Cromáticas 162

A-12. Resultados del desempeño de clasificación, F-Measure y Área-ROC con cada uno de los clasificadores para los 11 subconjuntos del conjunto complejo. Los resultados se basan en segmentación basada en PCA y características Texturales y Geométricas. 163

A-13. Resultados del desempeño de clasificación, F-Measure y Área-ROC con cada uno de los clasificadores para los 11 subconjuntos del conjunto complejo. Los resultados se basan en segmentación basada en PCA y características Texturales 164

A-14. Resultados del desempeño de clasificación, F-Measure y Área-ROC con cada uno de los clasificadores para los 11 subconjuntos del conjunto complejo. Los resultados se basan en segmentación basada en PCA y características Geométricas. 165

B-1. Comparativa de resultados antes y después de aplicar el Algoritmo Genético en datos con segmentación **Otsu** y características **Cromáticas, Texturales y Geométricas**. 166

B-2. Comparativa de resultados antes y después de aplicar el Algoritmo Genético en datos con segmentación **Otsu** y características **Cromáticas y Texturales** . . . 167

B-3. Comparativa de resultados antes y después de aplicar el Algoritmo Genético en datos con segmentación **Otsu** y características **Cromáticas y Geométricas** . . 167

B-4. Comparativa de resultados antes y después de aplicar el Algoritmo Genético en datos con segmentación Otsu y características Cromáticas, 167

B-5. Comparativa de resultados antes y después de aplicar el Algoritmo Genético en datos con segmentación Otsu y características Texturales y Geométricas.	168
B-6. Comparativa de resultados antes y después de aplicar el Algoritmo Genético en datos con segmentación Otsu y características Texturales	168
B-7. Comparativa de resultados antes y después de aplicar el Algoritmo Genético en datos con segmentación Otsu y características Geométricas.	168
B-8. Comparativa de resultados antes y después de aplicar el Algoritmo Genético en datos con segmentación PCA y características Cromáticas, Texturales y Geométricas.	169
B-9. Comparativa de resultados antes y después de aplicar el Algoritmo Genético en datos con segmentación PCA y características Cromáticas y Texturales	169
B-10. Comparativa de resultados antes y después de aplicar el Algoritmo Genético en datos con segmentación PCA y características Cromáticas y Geométricas.	169
B-11. Comparativa de resultados antes y después de aplicar el Algoritmo Genético en datos con segmentación PCA y características Cromáticas	170
B-12. Comparativa de resultados antes y después de aplicar el Algoritmo Genético en datos con segmentación PCA y características Texturales y Geométricas.	170
B-13. Comparativa de resultados antes y después de aplicar el Algoritmo Genético en datos con segmentación PCA y características Texturales	170
B-14. Comparativa de resultados antes y después de aplicar el Algoritmo Genético en datos con segmentación PCA y características Geométricas.	171

Prólogo

El proceso de clasificación de plantas, es un tema que ha sobresalido en el campo de la computación. Principalmente en México, que es uno de los países con mayor diversidad en flora, necesita de sistemas de identificación eficientes que puedan trabajar con la mayor cantidad posible de hojas de plantas.

En la actualidad existen sistemas desarrollados que han facilitado la clasificación de hojas de plantas. Algunos ejemplos son Famex y Gencomex, realizados por el Dr. José Luis Villaseñor en el departamento de Biología de la Universidad Nacional Autónoma de México [1] [9]. Tales sistemas son capaces de identificar hojas de plantas de las familias Magnoliophyta y Compositae, ambas existentes en territorio mexicano. Estos sistemas han comprobado su gran utilidad a través de su uso continuo por algunos años. Sin embargo, el rápido avance en la tecnología computacional, da pauta para seguir mejorando estos sistemas, lo que indica, que aún existe trabajo por hacer. Se requieren sistemas capaces de reconocer otras familias de plantas y clasificarlas por género. Además de utilizar las nuevas tecnologías, es importante implementar los nuevos algoritmos que se han generado recientemente por investigadores. Realizar un análisis de aquellos algoritmos en el proceso de clasificación y elegir los que generan mayor impacto en el reconocimiento de plantas es importante para generar sistemas eficientes.

Investigadores en el campo de la inteligencia artificial han mostrado interés en trabajar con proyectos para mejorar los sistemas actuales. Como parte de toda esta labor, se ha decidido aportar un análisis de las técnicas para la identificación de plantas, tomando en cuenta, métodos que han sobresalido en el campo computacional. Se pretende aportar datos, que ayuden a determinar mejor el uso de técnicas de identificación, analizando técnicas de segmentación, extracción de características y distintos clasificadores. Este análisis ayudara a identificar las ventajas y desventajas de utilizar una método sobre otro.

Sistema de Identificación Trivial y Compleja de Plantas: Un Análisis Comparativo.

por

Jesús Taltempa Moreno

Resumen

La detección de plantas a partir de la hoja es un proceso complejo aún para los especialistas cuando las plantas presentan una gran similitud entre estas.

En la literatura actual existen diversas técnicas de clasificación de hojas. Sin embargo, no existe una investigación que evalúe el impacto de la similitud en la precisión de clasificación.

El propósito general de esta investigación es realizar un análisis comparativo del desempeño de distintos clasificadores al clasificar hojas sobre dos conjuntos de datos. Un primer conjunto de datos al que hemos denominado trivial, debido a que cada una de las familias de plantas tienen formas totalmente distintas entre sí, y un segundo conjunto denominado complejo, que contiene familias de plantas cuyas formas son muy similares entre sí, no obstante que pertenezcan a distintas clases. Además se pretende obtener un análisis comparativo de la influencia de diferentes características, y tipos de segmentación. Con este fin, se modela un algoritmo genético, que permite rescatar aquellas características que más influyen en el desempeño del clasificador. El sistema propuesto se compone de tres etapas que son: segmentación, extracción de características e identificación de la planta.

Sistema de Identificación Trivial y Compleja de Plantas: Un Análisis Comparativo.

by

Jesús Taltempa Moreno

Abstract

The identification of plants through their leaves is a complex process even for specialists, when plants show great similarity among them.

In current literature there are techniques to classify leaves. However, there is no research to assess the impact of the similarity in classification accuracy.

The purpose of this research is a comparative analysis of the performance of different classifiers to classify leaves on two datasets. A first set of data, which we have called "trivial" because each family of plants included has definitely different and distinctive forms among them; a second set called "complex", containing families of plants whose forms are very similar, although they belong to different classifications. Additionally, it is intended to get a comparative analysis of the influence of different characteristics and types of segmentation. It is with this purpose that a genetic algorithm is modeled to rescue those characteristics that influence the performance of the classifier. The proposed system consists of three stages: segmentation, feature extraction and identification of the plant.

Capítulo 1

Introducción

El desarrollo de sistemas de reconocimiento de plantas basado en las hojas, es un tema de investigación muy importante. En los últimos años, varios investigadores han implementado diversas técnicas para clasificar plantas de ciertas regiones del planeta desarrollando sistemas de identificación. La identificación de hojas a partir de imágenes es un reto, debido a que la identificación de plantas por lo general, implica el análisis de partes de la muestra, como bordes, líneas, esqueleto, color etc. Las características del margen de la hoja generalmente reflejan la información importante del borde, forma y estructura de la hoja. Cada una de estas partes interviene en una correcta/incorrecta clasificación. Sin embargo, es necesario analizar el impacto que tiene cada clasificador en el reconocimiento de plantas, tomando en cuenta las características que se utilizan como muestra.

En este proyecto de tesis, se implementaron distintas técnicas de segmentación y técnicas de extracción características, con el objetivo de medir el impacto sobre el desempeño de cada clasificador. Ésta es la razón principal de la tesis: analizar el impacto de las diferentes técnicas de extracción de características y segmentación en el desempeño de los clasificadores.

Para analizar el impacto de las distintas técnicas de extracción de características, se implementó un algoritmo genético, que permitió rescatar solo aquellas características que influyeron para una buena clasificación, y elimino aquellas que no beneficiaban al clasificador, ya que solo agregaban ruido, lo que empeoraba su buen desempeño.

El análisis se implementó sobre dos conjuntos de datos, el primero, con imágenes de hojas muy diferentes entre sí, denominado conjunto trivial y otro, con imágenes de hojas muy similares

entre sí, denominado conjunto complejo. Se establecieron estos términos, debido a la facilidad de reconocer hojas por ser muy distintas entre sí, o dificultad de reconocerlas por ser muy parecidas entre sí.

1.1. Problemática

En el estado del arte, se ha notado que los trabajos son realizados sobre conjuntos de datos de imágenes con pocas clases de hojas [40] [42] [43] [50] [15], y conjuntos de datos donde las plantas a identificar son muy diferentes entre sí [27] [43] [46] [47] [49] [15]. Sin embargo, no existe un estudio donde el foco de atención sea el reconocimiento de gran variedad de hojas de planta y mucho menos donde se haga una comparativa del mejor clasificador para cada tipo de hoja.

El estudio y/o análisis del impacto de cada proceso en el desempeño de un clasificador es un reto en la identificación. En los sistemas actuales es común ver nuevos métodos de reconocimiento de patrones que hacen especial énfasis al seleccionar parámetros óptimos. Estos métodos emplean técnicas especiales para encontrar los parámetros con los que el clasificador se desempeñará de forma eficiente. Sin embargo, no existen en la actualidad técnicas que ayuden a evaluar el impacto en la selección de cada etapa de procesamiento en un sistema de reconocimiento. El desarrollo de este tipo de técnicas es fundamental para mejorar la selección de las técnicas en cada etapa y así obtener un sistema de reconocimiento de plantas eficiente.

Otros estudios, se han enfocado al mejoramiento de técnicas para extraer características, tales como la forma, la textura y el color. Sin embargo no existe un estudio, donde se implemente una combinación de varias técnicas y que además apliquen algún algoritmo genético que ayude a rescatar solo las más importantes.

1.2. Justificación

El desarrollo de este proyecto ayudará a obtener un sistema de clasificación e identificación de plantas robusto que permita identificar plantas muy similares entre sí a partir de un conjunto óptimo de características, disminuyendo el tiempo de identificación y aumentando el desempeño de los clasificadores actuales. Este proyecto ayudará a entender mejor el impacto de

cada clasificador en cada tipo de planta. Esto contribuirá a seleccionar de forma más eficiente cada clasificador y así obtener buenos resultados al momento de identificar plantas. El tener un clasificador específico, capaz de dar mejores resultados dependiendo los tipos de hojas a clasificar, abre la posibilidad de crear dispositivos clasificadores de plantas, más robustos y más eficientes. Por consecuencia, al tener clasificadores que arrojen buenos resultados, daría más confiabilidad a los expertos en botánica para el uso de herramientas computacionales en su área.

1.3. Objetivos y metas

1.3.1. Objetivo general

El objetivo general de este proyecto es obtener un análisis comparativo entre clasificadores con distintos conjuntos de datos y analizar el impacto de cada etapa de un sistema de reconocimiento en el desempeño de clasificación e identificación de plantas a partir de imágenes muy similares y disimilares entre sí para utilizar el clasificador en el sistema que más se adapte al tipo de hoja.

1.3.2. Objetivos particulares

1. Obtener conjunto de datos de distintos tipos de hojas de planta.
2. Crear dos conjuntos de datos con imágenes de la hoja, un conjunto con hojas disimilares y otro conjunto con hojas muy similares entre sí.
3. Implementar técnicas de segmentación de imágenes a los conjuntos de datos.
4. Implementar técnicas de extracción de características a los conjuntos de datos.
5. Implementar distintos clasificadores sobre las características obtenidas.
6. Implementar un algoritmo genético básico sobre las características obtenidas.
7. Obtener un análisis del impacto de cada una de las técnicas de extracción de características utilizadas.

8. Obtener un análisis del desempeño de cada clasificador en ambos conjuntos de datos.

1.4. Hipótesis

¿Existe una diferencia significativa en el desempeño de clasificadores de plantas a partir de hojas al utilizar conjuntos de datos triviales y complejos?, Y si es el caso, ¿Es posible mejorar su desempeño aplicando técnicas apropiadas?

1.5. Metodología

A continuación se presenta una breve reseña de la metodología realizada en este proyecto y se presenta de manera detallada en el Capítulo 3.

El diagrama de la Figura 1-1, representa cada una de las etapas de la metodología.

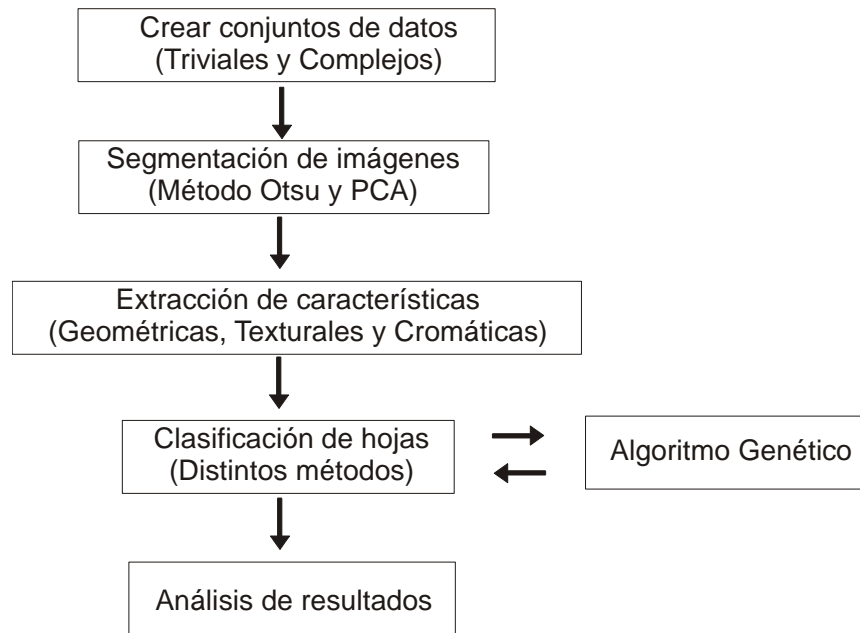


Figura 1-1: Etapas de la metodología propuesta para este proyecto.

El primer paso de este proyecto, fue el análisis de varios artículos relacionados con temas acerca de la identificación de plantas, para conocer las distintas técnicas y métodos que manejan

otros autores. El conjunto de datos con el que se trabajó, fue de 220 familias distintas. Por cada familia de hoja de planta, se contó con al menos 10 imágenes de muestra. También se realizó un filtrado de imágenes, eliminando algunas con tamaño muy pequeño y poca resolución que podrían causar errores de segmentación. En este trabajo no se aplicó la etapa de pre-procesamiento debido a que las imágenes de hojas se encontraban bajo ambientes totalmente controlados (fondo blanco y sin hojas solapadas).

La primera etapa del proyecto consistió en crear los conjuntos de datos triviales y complejos. En esta etapa se compararon de forma manual, la similitud y disimilitud entre imágenes de cada familia, creando un conjunto de datos donde se colocaron aquellas imágenes de hojas que eran muy distintas entre sí, y otro conjunto de hojas muy parecidas entre sí. El conjunto de familias similares se denominó conjunto complejo y el conjunto de familias disimilares se denominó conjunto trivial.

En la segunda etapa, a cada imagen del conjunto de datos, se les aplicó una segmentación Otsu y una segmentación PCA. Con las imágenes segmentadas se extrajeron características texturales, cromáticas y geométricas. En esta etapa, se obtuvieron varios resultados mezclando los tipos de segmentación con las distintas características.

La etapa de clasificación fue la más importante, ya que aquí se enfoca el propósito de este proyecto. Se implementó cada clasificador en las características extraídas, obteniendo resultados en cada conjunto de datos. Entonces, se aplicó un algoritmo genético que eliminó aquellas características que solo agregaban ruido al clasificador. Por lo tanto, se analizó el impacto de las características que influyeron en los correcta/incorrecta clasificación, comparando los resultados iniciales con los finales.

1.6. Estado del arte

Las plantas existen en todas partes del mundo y juegan un papel importante para la vida y el desarrollo humano, ya que no solo son de interés en investigaciones de botánica, sino también en otras ramas, tales como la agricultura [40] [42] [43] [48], ecología vegetal [40] [46] [49], medicamentos basado en plantas [27] [43] [46] [50], conservación natural y también en muchas

situaciones de interés público. En el mundo, existen aproximadamente una variedad de 310 000 a 420 000 especies de plantas [27], sin tomar en cuenta que faltan muchas por ser descubiertas, clasificadas y usadas. Por esta razón, la identificación de plantas para la mayoría de las especies no es una tarea trivial.

La clasificación de especies de plantas basándose en el análisis de la hoja se ha llevado a cabo por botánicos, especialistas en plantas y varios investigadores durante mucho años [15] [46]. Es bien sabido que la mejor forma de extraer características validas es basándose en la imagen de la hoja de la planta. Regularmente, las técnicas de reconocimiento de patrones en imágenes involucran técnicas de medición de características morfológicas y de textura, además de utilizar un sistema experto para reconocer el objeto a partir de las características. Sabiendo que la forma externa de la hoja provee rica información para clasificar, varios sistemas de identificación de planta se han enfocado en cuatro importantes tipos de características, que son: forma [27] [45] [47] [48] [50], textura [15] [42] [43], color [40] [52], y venación de la hoja [49] [54] [56].

La forma de la hoja es una de las características más importantes de la planta y regularmente se puede analizar en imágenes a escala de grises con fondo blanco o negro. Existen dos tipos de análisis para la forma la hoja, que son los basados en contorno y los basados en la región. Ejemplos de características basadas en la región incluyen el área, momentos de Zernike, momentos de Hu etc. Características basadas en el contorno usualmente son el perímetro, circularidad, rectangularidad, elipsidad y otros métodos basados en la curvatura de la hoja. Básicamente todas las características basadas en la forma de la hoja son llamadas características geométricas [40].

Usualmente, la mayoría de autores que utilizan características geométricas para el reconocimiento de plantas, incluyen características de textura para una mejor clasificación. Por ejemplo en [15] los autores emplean descriptores geométricos básicos, tales como el área, perímetro, circularidad etc. Además de un histograma que extrae descriptores de textura. En [43] los autores diseñan un dispositivo portable para el reconocimiento de hojas utilizando un detector de bordes de Canny. [47] también se enfoca al reconocimiento a base del contorno utilizando descriptores de Fourier para la traslación y rotación. Otro autor en [50] extrae características para describir las variaciones del borde de la hoja a través de un movimiento rotatorio en el

que utiliza el Teorema de Bayes, donde el objetivo es medir la similitud de dos hojas con diferentes números de características de borde. Cada una de estas investigaciones propone sistemas innovadores ya que diseñaron sistemas portables, o sistemas web para una buena accesibilidad al sistema y demuestran ser eficaces con precisiones de clasificación altas que van del 90 % al 98 %, incluso realizan pruebas con hojas secas, mojadas o deformes. Sin embargo, lo realizan con hojas muy distintas en su forma.

Otros autores afirman que se pueden reconocer hojas por la similitud de color, es decir, utilizando características cromáticas. Tal es el caso de [40] donde los autores realizaron una comparación de imágenes en base a los histogramas de color de la hoja. Otro autor en [52] realiza la comparativa basándose en el espacio de color $L^*a^*b^*$, ya que argumenta que el uso de este espacio de color es más consistente y presenta más o menos el mismo eje para toda la hoja a diferencia del espacio de color RGB. Existen varias investigaciones que se basan en descriptores cromáticos [30], [59], [37], [58], [21]. Sin embargo, muy pocas utilizan estas características como una opción para el reconocimiento de hojas de planta.

Aunque los enfoques de clasificación como la forma, textura y color son válidos, no siempre son útiles cuando se trata de identificar especies de plantas que tienen características similares en los descriptores antes mencionados. Este es el caso de las plantas que pertenecen a varios cultivos de la misma especie. Por esta razón, recientemente autores en [49], [54] y [56] declaran que las propiedades de la vena de la hoja son de alta importancia para el reconocimiento de plantas, ya que en la vena se pueden apreciar las características fisiológicas de la planta, incluso cuando su apariencia es similar. La teoría fractal, muy relacionada con la venación de la hoja, es una nueva disciplina importante en la rama de la ciencia. Así en [49] los autores proponen un método de descripción de características de hojas de plantas basado en un esquema de dimensión fractal y dimensión de venación fractal. Obtienen el contorno de la hoja y venación de imágenes fractales como características de clasificación. Además mejoran el reconocimiento con los siete momentos de Hu (descriptores geométricos), y obtienen un alto grado de reconocimiento demostrando que su método propuesto es efectivo. Más adelante en [54] también tomando en cuenta el análisis de las características morfológicas de la vena de la hoja, estudian a 3 especies de legumbres: el haba de soya, el frijol blanco y el frijol rojo donde los frijoles pertenecen a la misma especie, presentando hojas similares excepto por su color de

venación, ya que es oscura en el frijol rojo. Dado que su objetivo en la investigación buscaba la simplicidad y el bajo costo, utilizaron imágenes de hojas usando un escáner estándar y no imágenes de hojas limpias a base de procedimientos de tinción que son caros. Por último en [56] los mismos autores de [54] proponen una mejora en el sistema utilizando ahora características de venas multiescala que precisa en contraste a características de una sola escala en problemas de análisis profundo. Los resultados mostrados en este artículo superan los resultados de humanos expertos, aunque surge el mismo problema de realizarlo solo sobre pocas especies de planta.

Los algoritmos de clasificación juegan un papel importante en los sistemas de identificación de plantas. En la literatura actual utilizan diversos clasificadores para el reconocimiento de hojas. En [43] (mencionado antes) donde se diseñó un dispositivo portable para el reconocimiento de hojas, los autores utilizan el algoritmo de redes neuronales *Back-Propagation*. [49] que se basa en la venación de la hoja, utiliza el clasificador del vecino más cercano. Otros autores incluso realizan una comparativa entre clasificadores como [54] que hace una comparativa entre el clasificador SVM (*Support Vector Machines*) con núcleos lineales y gaussianos, Random Forest y PDA (*Penalized Discriminant Analysis*) de los cuales obtuvieron mejores resultados de clasificación con una precisión del 87% con el clasificador PDA. En [27], los autores también hicieron una comparativa entre clasificadores. Aunque el mejor aporte fue que compararon un método creado por ellos mismos llamado MMC (*Move Median Centers*, por sus siglas en inglés) para el reconocimiento del contorno de la hoja. Compararon la eficiencia de este método con los clasificadores 1-NN (*1-Nearest Neighbor*) y K-NN (*K-Nearest Neighbor*), donde demostraron que MMC es más robusto que otros basados también en características del contorno. Además el método logra no solo clasificar, sino que además redujo el tiempo sin sacrificar la precisión. Aunque estas comparativas tuvieron resultados eficientes lo hicieron sobre características distintas, así que no se hizo un análisis sobre qué características influyen mejor en cada clasificador. Otros autores en [42] realizan una comparación con el clasificador SVM para detectar síntomas de nutrientes enfermos de la planta de aceite, basados en sistemas visibles en sus hojas y también para su clasificación en grupo. Las SVM las evalúan con 3 Kernel distintos: Kernel lineal, Kernel polinomial con margen suave y Kernel polinomial con margen fuerte. En su estudio mostraron que el Kernel polinomial con margen suave produce los mejores resultados en un promedio de 95% de clasificación correcta comparando con los otros tipos de Kernel.

En [55] muestran que para detectar síntomas de etapa temprana es importante el ángulo con el cual se lleva a cabo la detección. Los autores se enfocan al hongo de oídio que es el más común en la vid. Utilizan un total de 35 hojas para su trabajo, 10 como prueba y 25 como ejemplos de validación. Utilizaron imágenes con 5 ángulos de 0° a 75° . Sus resultados indicaron que la sensibilidad generalmente incrementaba cuando la vista del ángulo incrementaba, con un valor pico obtenido para imágenes adquiridas en 60° . Estas investigaciones son muy importantes, ya que al desarrollar sistemas capaces de localizar infecciones iniciales, es posible llevar a cabo un tratamiento con fumigaciones especiales. Aunque en estas investigaciones se realizó una comparativa de clasificadores no se realizó para clasificar hojas si no para la detección de enfermedades de la planta.

Otras investigaciones relacionadas con la identificación de hojas de plantas aportan datos importantes al plantear que analizar solo las características de la hoja no es suficiente, ya que las imágenes en el campo son mucho más complicadas, dado a que varían con la ubicación, el tiempo, las condiciones de iluminación y contienen gran cantidad de ruido. Varios métodos existentes solo se concentran en la extracción de características de cada una de las imágenes de las hojas, mientras que no tienen en cuenta las imágenes como un todo y tienen solo éxito en imágenes sin fondo complejo, sin colores similares o superposición de otras hojas. Con los avances en la tecnología computacional, dispositivos móviles, internet etc. es fácil pensar en un sistema que pueda tomar fotografías de una planta en el campo y así poder reconocer un determinado tipo de planta. En este caso se presentan el problema de segmentación, ya que la hoja objetivo se superpone con otras hojas u objetos tales como las ramas, además su color es parecido al color de otras hojas, el fondo puede ser complicado y un objeto de la hoja puede ser considerado como otro objetivo distinto debido a las diferencias significativas de escala de grises en la misma región de la hoja. Además, las imágenes pueden perder detalles y la detección del borde puede no ser segmentada efectivamente. Por esta razón, otros autores toman en cuenta otras situaciones al momento de reconocer hojas.

Los artículos [45] y [52] proponen algoritmos tomando en cuenta imágenes de hojas en un fondo complejo. En [47] los autores proponen un nuevo algoritmo de umbralización que puede

segmentar una sola hoja de una imagen con varias hojas extraída de una transmisión de video de un sistema en línea. Utilizan los operadores OTSU y CANNY para segmentar el área de la hoja objetivo. El algoritmo tiene la ventaja de segmentar hojas de imágenes complicadas que contienen una escala de grises diferentes en la misma región. Además el sistema de video es en tiempo real y puede obtener imágenes de bordes precisos, limpios y lisos. En [52] elaboran una aplicación móvil con el objetivo de que sea accesible a cualquier persona. Proponen una aplicación llamada Folia que identifica plantas partiendo de una imagen de una hoja con un fondo natural complejo. Estos estudios son un gran aporte en el análisis de hojas ya que la gran mayoría trabaja con imágenes de hojas con fondo trivial. Aunque estos estudios demuestran que los métodos de segmentación son eficientes, solo lo hacen con un tipo de hoja.

Los métodos de reducción dimensional también juegan un papel importante en el análisis de los datos de la hoja de la planta, ya que las segmentaciones de imágenes de la hoja de alta resolución con detalles finos, puede resultar en demasiados bordes y separar el objetivo en varias regiones, además las imágenes de alta resolución pueden alentar el proceso de segmentación. Esto se realiza a menudo como paso de pre procesamiento antes de continuar con la clasificación de plantas, ya que ayuda a eliminar factores menos importantes y ruidosos. Recientemente, otros autores proponen métodos para la clasificación de plantas tomando en cuenta la reducción de dimensionalidad. En [46] hacen uso de la etiqueta de propagación para proponer una nueva medida de peso y presentar un método de clasificación llamado SLDP (Supervisión Local de Análisis de Proyección). Primero aplican el algoritmo *Warshall* para la etiqueta de propagación y obtienen la matriz de la etiqueta, entonces la incorporan en el peso. Después los puntos de datos de alta dimensión de espacio son empujados por un vecino discriminante para formar una proyección óptima de baja dimensión. Sus resultados muestran ser eficaces. Este algoritmo se puede tomar como una opción en la etapa de pre-procesamiento.

Capítulo 2

Preliminares

Los seres humanos tenemos una capacidad compleja de reconocer objetos y clasificarlos. Somos capaces de ver alguna escena, extraer toda la información necesaria y describir de qué trata la escena. Este proceso de adquisición de información a través de una imagen se puede realizar artificialmente por medio de un computador, aunque los resultados sean menos eficientes. Las investigaciones sobre imágenes tratan de mejorar el proceso de identificación a través de distintos métodos. Al momento de trabajar con estos procesos nace el concepto de procesamiento digital de imágenes (PDI) que puede definirse como la capacidad de una computadora para analizar automáticamente una imagen utilizando un conjunto de procedimientos [44]. El PDI ha adquirido gran interés en los últimos años y se ha hecho indispensable en múltiples aplicaciones científicas. Entre algunas aplicaciones, está la inteligencia artificial, que es utilizada para el reconocimiento de patrones. Desde el punto de vista de la clasificación, uno de los principales problemas en reconocimiento de patrones, es encontrar las clases de cada uno de los objetos que se encuentran en la imagen. Sin embargo, esto requiere de varios procesos como: pre-procesamiento, segmentación, extracción de características (texturales, cromáticas o geométricas) y finalmente identificación. Algunos de estos procesos influyen considerablemente en la identificación de la imagen (segmentación y pre-procesamiento)

El nivel de profundización de cada una de estas etapas depende de varios factores como: La calidad de la imagen original, hardware empleado en la adquisición y métodos utilizados en el procesamiento. Sin embargo, el término “mejorar” en PDI es de carácter subjetivo, en el sentido de que dos o más observadores pueden tener criterios distintos para decidir si una imagen ha

sido supuestamente mejorada o no, ya que todo depende de la aplicación del tratamiento de la imagen.

A continuación se describen varias técnicas de segmentación y extracción de características, que han sido tratadas en la literatura e influyen de manera significativa en este proyecto de investigación.

2.1. Métodos de segmentación

La segmentación es una de las etapas más importantes en el tratamiento de imágenes. La segmentación se puede considerar como la forma de separar los objetos de interés en una imagen, del resto que se consideran no relevante [44]. En la última década se han propuesto distintos métodos de segmentación. Cada técnica tiene desempeños notables sobre imágenes específicas, es decir, no existe una sola técnica que pueda ser utilizada para segmentar cualquier imagen. A continuación se describen algunos métodos de interés en esta tesis, a saber el método de Otsu, método PCA y método de Frontera Adaptativa.

2.1.1. Método de Otsu

Este método está basado en la técnica de umbralización, con la diferencia que este método busca el umbral óptimo T dada una imagen I para aplicar la umbralización. Dado que una imagen contiene N pixeles que representan L niveles de gris, las probabilidades acumuladas hasta T y desde T hasta L resultan ser:

$$w_1(t) = \sum_{z=1}^T P(z) \quad (2-1)$$

$$w_2(t) = \sum_{z=T+1}^L P(z) \quad (2-2)$$

donde $P(z)$ es la distribución del histograma.

Las medias y varianzas asociadas están dadas por:

$$\mu_1(t) = \sum_{z=1}^T \frac{zP(z)}{w_1(t)} \quad (2-3)$$

$$\mu_2(t) = \sum_{z=T+1}^L \frac{zP(z)}{w_2(t)} \quad (2-4)$$

$$\sigma_1^2(t) = \sum_{z=1}^T (z - \mu_1(t))^2 \frac{P(z)}{w_1(t)} \quad (2-5)$$

$$\sigma_2^2(t) = \sum_{z=T+1}^L (z - \mu_2(t))^2 \frac{P(z)}{w_2(t)} \quad (2-6)$$

Finalmente la varianza ponderada se define como:

$$\sigma_w^2(t) = w_1(t)\sigma_1^2(t) + w_2(t)\sigma_2^2(t) \quad (2-7)$$

Se elige el umbral T correspondiente al nivel de intensidad que proporcione la mínima varianza ponderada definida en 2-7. De modo que para el umbral T con la condición mayor/menor o igual se elige la imagen deseada.

2.1.2. El método PCA

El Análisis de Componentes Principales (*PCA* por sus siglas en ingles), es una técnica para identificar patrones en un conjunto de datos y representarlos de modo que sean resaltadas sus diferencias y semejanzas. La idea principal de este método consiste en reducir el conjunto de datos original a k componentes principales, es decir, proyecta el conjunto datos, dentro de un subespacio de una dimensión más reducida, eliminando la mayor cantidad de información redundante (correlacionada). Al considerar solo aquellas características de mayor valor, se obtiene una reducción de dimensiones sin perder demasiada información [39]. Es importante resaltar que este método supone que las variables de entrada son correlacionadas.

Para implementar el método, consideremos un conjunto de imágenes de entrenamiento I . Como primer paso se obtiene la media correspondiente a cada una de las imágenes de entrenamiento y se realiza un ajuste de datos, restando la media correspondiente a cada imagen, obteniendo un nuevo conjunto de datos:

$$i_1, i_2, \dots, i_n \in I - \bar{I}$$

Luego se construye una matriz X tal que cada columna es una imagen de muestra y XX^T es la matriz de covarianza de las muestras de entrenamiento. En seguida a partir de la matriz de covarianza, se calcula la matriz de eigenvectores y se organizan de mayor a menor según los eigenvectores de la matriz de covarianza, despreciando los eigenvectores menos significativos. Esto tiene una reducción de dimensiones, lo que implica la pérdida de los datos menos significativos [53].

Finalmente se obtiene la transpuesta del vector característico y se multiplica por la transpuesta de la matriz que contiene los datos iniciales ajustados.

2.1.3. Frontera Adaptativa

El método de frontera adaptativa, también conocido como umbralización adaptativa, es comúnmente usado en imágenes donde la iluminación no es uniforme o en aquellas donde los objetos son muy pequeños respecto al fondo. Este método consigue que el valor del umbral varíe como una función de las características locales de la imagen ya que la imagen se divide en subimágenes en donde a cada una se calcula un umbral [35]. El procedimiento iterativo para encontrar el umbral óptimo, es el siguiente:

1. Se selecciona un umbral inicial T . (Se puede utilizar el método de Otsu)
2. Se divide la imagen en dos zonas. G_1 corresponde a los pixeles con una intensidad menor o igual que T y G_2 corresponde a los pixeles mayores que T .
3. Se calcula un nuevo umbral $T = \frac{1}{2}(\mu_1 + \mu_2)$.
4. Se repiten los pasos 2 a 4 hasta que la diferencia sea menor que un parámetro dado.

2.2. Extracción de características geométricas

Para que una computadora sea capaz de reconocer alguna imagen, es necesaria la extracción de características de la imagen. Sin embargo la elección de características es muy dependiente de la aplicación en sí, ya que se busca resaltar solo aquellos aspectos representativos que faciliten la separación de la imagen en clases distintas. En algunos casos, las formas geométricas de ciertas

figuras, pueden dar suficiente información para las técnicas de clasificación de la imagen. No obstante, debemos tomar en cuenta que las características geométricas solo pueden identificar formas con grandes diferencias, por lo tanto, es necesario combinarlas con otros descriptores para mejorar el desempeño. A continuación, se describen las características geométricas más usuales en PDI.

Área

El área de un región segmentada O puede ser calculada sencillamente mediante el número de pixeles que lo conforman. Si la imagen segmentada está representada por su región se calcula mediante:

$$Area(O) = N = |O| \quad (2-8)$$

donde N es el número de puntos que conforman a la imagen.

Si la imagen segmentada está representada por su contorno, entonces se calcula con:

$$A = \frac{1}{2} \left| \sum_{i=1}^N (x_i y_{i+1} - x_{i+1} y_i) \right| \quad (2-9)$$

Perímetro

El perímetro de un objeto O se determina a través de su contorno exterior. Para el cálculo del perímetro es necesario tomar en cuenta el tipo de vecindad, ya que la distancia del perímetro de un contorno bajo la vecindad 4-vecinos es mayor a la de 8-vecinos. En el caso de una vecindad 8-vecinos, los movimientos horizontales y verticales del contorno tienen una distancia de 1, mientras que las diagonales una distancia de $\sqrt{2}$. Para comprender mejor el tipo de vecindad, puede consultar el código de cadena básico, explicado mas adelante.

El perímetro puede ser calculado a través de:

$$Perímetro(O) = \sum_{i=1}^M longitud(c_i) \quad (2-10)$$

donde

$$longitud(ci) = \sqrt{(x_i - x_{i-1})^2 + (y_i - y_{i-1})^2} \quad (2-11)$$

El perímetro que se calcula mediante las ecuaciones anteriores presentan un valor que calcula las distancias verdaderas. En la práctica este valor normalmente es ajustado. El nuevo valor del perímetro es definido como $U(O)$ donde:

$$U(O) = 0,95 * Perimetro (O) \quad (2-12)$$

Centro de gravedad

El centro de gravedad, también llamado Centróide. Es el punto de la imagen, donde, por su geometría, se encuentra concentrada la masa de la imagen.

Si la imagen segmentada está representada por su región, el centro de gravedad se calcula como:

$$gx = \frac{1}{N} \sum_i x_i \quad (2-13)$$

$$gy = \frac{1}{N} \sum_i y_i \quad (2-14)$$

donde N es el número de puntos de la forma $y f(x_i, y_i) = 1$ ó $f(x_i, y_i) = 0$, dependiendo el tipo de umbralización aplicada a la imagen.

Si la imagen segmentada está representada por su contorno, entonces la posición del centro de gravedad está dada por:

$$gx = \frac{1}{6A} \sum_i (x_i + x_{i+1})(x_i y_{i+1} - x_{i+1} y_i) \quad (2-15)$$

$$gy = \frac{1}{6A} \sum_i (y_i + y_{i+1})(x_i y_{i+1} - x_{i+1} y_i) \quad (2-16)$$

donde A es el área del contorno.

Eje de menor inercia

El eje de menor inercia sirve como línea de referencia para preservar la orientación de la región segmentada. Este descriptor se define como la línea donde la integral del cuadrado de distancias de los puntos del borde de la región segmentada es el mínimo.

Dado que $x \sin \theta - y \cos \theta = 0$ es la ecuación paramétrica del eje de menor inercia, la pendiente del ángulo θ está estimada como sigue:

Dado que α es el ángulo entre el eje de menor inercia y el eje x . La inercia está dada por:

$$I = \frac{1}{2}(a + c) - \frac{1}{2}(a - c) \cos(2\alpha) - \frac{1}{2}b \sin(2\alpha) \quad (2-17)$$

donde

$$a = \sum_i x_i^2, b = 2 \sum_i x_i y_i, c = \sum_i y_i^2. \quad (2-18)$$

Por lo tanto,

$$\frac{dI}{d\alpha} = (a - c) \sin(2\alpha) - b \cos(2\alpha) \quad (2-19)$$

$$\frac{d^2I}{d\alpha^2} = 2(a - c) \cos(2\alpha) + 2b \sin(2\alpha) \quad (2-20)$$

dado $\frac{dI}{d\alpha} = 0$, se obtiene:

$$\alpha = \frac{1}{2} \arctan\left(\frac{b}{a - c}\right), \quad -\frac{\pi}{2} < \alpha < \frac{\pi}{2} \quad (2-21)$$

La pendiente del ángulo θ está dada por:

$$\theta = \begin{cases} \alpha + \frac{\pi}{2} & \text{if } \frac{d^2I}{d\alpha^2} < 0 \\ \alpha & \text{otherwise} \end{cases} \quad (2-22)$$

Energía de flexión promedio

Este descriptor se obtiene integrando los cuadrados de la función de curvatura $K(i)$. La función de curvatura se puede obtener a partir del cambio de la pendiente y está dada por:

$$K(i) = \tan^{-1} \begin{pmatrix} y_{i+k} - y_i \\ x_{i+k} - x_i \end{pmatrix} - \tan^{-1} \begin{pmatrix} y_i - y_{i-k} \\ x_i - x_{i-k} \end{pmatrix} \quad (2-23)$$

El contorno del objeto se recorre en sentido de las manecillas del reloj. Entonces, la energía de flexión promedio está dado por:

$$EFP = \frac{1}{N} \sum_{i=1}^N K(i)^2 \quad (2-24)$$

donde N es el número de puntos del contorno de la imagen.

Este descriptor es robusto y puede ser usado para objetos muy similares entre sí. El círculo es la figura que tiene el valor mínimo de energía de flexión promedio.

Método de ejes principales

Los ejes principales de una forma dada, están definidos como dos segmentos de líneas que se cruzan ortogonalmente en el centroíde de la región segmentada y representan las direcciones con cero correlación cruzada [8]. El contorno de una imagen es visto como una instancia de una distribución estática. Los ejes principales se calculan a partir de la matriz de covarianza C de un contorno, está se calcula como sigue:

$$C = \frac{1}{N} \sum_i \begin{pmatrix} x_i - g_x \\ y_i - g_y \end{pmatrix} \begin{pmatrix} x_i - g_x \\ y_i - g_y \end{pmatrix}^T = \begin{pmatrix} c_{xx} & c_{xy} \\ c_{yx} & c_{yy} \end{pmatrix} \quad (2-25)$$

donde N es el número de puntos del contorno de la imagen. $G(g_x, g_y)$ es el centroide de la forma y los valores c_{xx} , c_{xy} , c_{yx} y c_{yy} estan dados por:

$$c_{xx} = \frac{1}{N} \sum_i (x_i - g_x)^2 \quad (2-26)$$

$$c_{xy} = \frac{1}{N} \sum_i (x_i - g_x)(y_i - g_y) \quad (2-27)$$

$$c_{yx} = \frac{1}{N} \sum_i (y_i - g_y)(x_i - g_x) \quad (2-28)$$

$$c_{yy} = \frac{1}{N} \sum_i (y_i - g_y)^2 \quad (2-29)$$

Las longitudes de los dos ejes principales son iguales a los eigenvalores λ_1 y λ_2 de la matriz de covarianza C de un contorno. Entonces los eigenvalores λ_1 y λ_2 se pueden calcular como sigue:

$$\lambda_1 = \frac{1}{2} \left(c_{xx} + c_{yy} + \sqrt{(c_{xx} + c_{yy})^2 - 4(c_{xx}c_{yy} - c_{xy}^2)} \right) \quad (2-30)$$

$$\lambda_2 = \frac{1}{2} \left(c_{xx} + c_{yy} - \sqrt{(c_{xx} + c_{yy})^2 - 4(c_{xx}c_{yy} - c_{xy}^2)} \right) \quad (2-31)$$

Rectángulo de delimitación mínima

Este descriptor también llamado caja de delimitador mínimo, es el rectángulo más pequeño que contiene todos los puntos de la región segmentada. Para encontrar este rectángulo, es necesario obtener la longitud del eje mayor y eje menor del objeto [28]. El eje mayor se puede determinar mediante el diámetro de un contorno cerrado C que viene dado por la expresión que sigue:

$$Diametro(C) = \text{máx}[D(p_1, p_2)] \quad (2-32)$$

donde D , es la medida de distancia entre los puntos p_1 y p_2 , los cuales son los puntos más extremos del contorno. Si solo se conocen los puntos que representa al eje mayor, su longitud se puede determinar mediante:

$$Longitud \text{ de eje mayor} = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \quad (2-33)$$

El eje menor es perpendicular al eje mayor. Una vez encontrados los puntos que determinan

al eje mayor, la longitud del eje menor se calcula de igual manera que 2-33. La Figura 2-1 muestra el rectángulo formado en un contorno de la forma basándose en el eje mayor y eje menor.

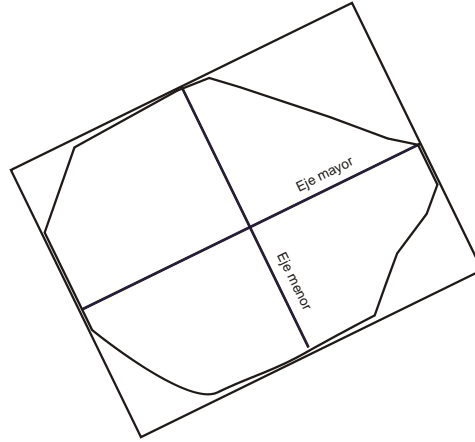


Figura 2-1: Rectángulo de delimitación mínima de un contorno, basandose en el eje mayor y eje menor.

Excentricidad

La excentricidad es la relación entre la longitud del eje mayor y la longitud del eje menor. Esta se puede calcular principalmente con el método de ejes principales o método del rectángulo de delimitación mínima (*Bounding Box*). Utilizando el método de ejes principales, la excentricidad puede ser calculada mediante los eigenvalores:

$$E = \frac{\lambda_2}{\lambda_1} \quad (2-34)$$

Si utilizamos el método del rectángulo de delimitación mínima, la excentricidad está dada por la siguiente expresión:

$$E = \frac{W - H}{H} \quad (2-35)$$

donde W es la anchura y H la altura del objeto las cuales están dadas por:

$$W = \text{máx } X(t) - \text{mín } X(t) \quad (2-36)$$

$$H = \text{máx } Y(t) - \text{mín } Y(t) \quad (2-37)$$

donde $X(t)$ e $Y(t)$ son respectivamente, el eje mayor y menor del objeto.

Relación de circularidad

La relación de circularidad representa el grado de similaridad de una imagen segmentada con un círculo [17]. Existen 3 definiciones. La primera nos dice que la relación de circularidad es la relación entre el área de una forma y el área de un círculo, es decir que tienen el mismo perímetro. Con esta definición se puede calcular la circularidad como sigue:

$$C_1 = \frac{A_s}{A_c} \quad (2-38)$$

donde A_s es el área de la imagen y A_c es el área del círculo que tiene el mismo perímetro que la imagen.

Ahora, supongamos que el perímetro es O , entonces podemos calcular el área del círculo como:

$$A_c = \frac{O^2}{4\pi} \quad (2-39)$$

Dada la ecuación anterior también podemos calcular a la circularidad como:

$$C_1 = \frac{4\pi * A_s}{O^2} \quad (2-40)$$

Ahora, sabiendo que 4π es una constante, tenemos la segunda definición, la cual nos dice que relación de circularidad es la relación entre el área y el perímetro cuadrado de la imagen:

$$C_2 = \frac{A_s}{O^2} \quad (2-41)$$

La relación de circularidad también se llama círculo varianza y se define de la siguiente forma:

$$C = \frac{\sigma_R}{\mu_R} \quad (2-42)$$

Donde μ_R y σ_R son la media y la desviación estándar de la distancia radial desde el centroide (g_x, g_y) de la imagen a los puntos de frontera (x_i, y_i) , $i \in [0, N - 1]$: Para el cálculo de estas se utilizan las siguientes fórmulas respectivamente:

$$\mu_R = \frac{1}{N} \sum_{i=1}^{N-1} d_i \quad (2-43)$$

$$\sigma_R = \sqrt{\frac{1}{N} \sum_{i=1}^{N-1} (d_i - \mu_R)^2} \quad (2-44)$$

donde

$$d_i = \sqrt{(x_i - g_x)^2 + (y_i - g_y)^2} \quad (2-45)$$

El valor de circularidad es máximo 1 y la forma más compacta es un círculo, así que cualquier otra forma varía su valor de redondez entre 0 y 1. La Figura 2-2 muestra el valor de circularidad de un círculo y un rectángulo.

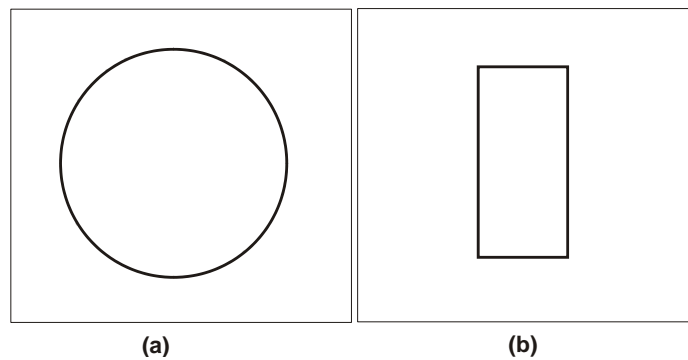


Figura 2-2: Relación de Circularidad para los siguientes objetos. (a) Objeto totalmente redondo. Circularidad = 1.001, (b) Objeto rectángulo. Circularidad = 0.7415

Varianza de elipse

La varianza de elipse es un error de mapeo de la imagen segmentada para adaptarse a una elipse que tiene una matriz de covarianza como: $C_{elipse} = C$ 2-25. Si asumimos que:

$$V_i = \begin{pmatrix} x_i - g_x \\ y_i - g_y \end{pmatrix} \quad (2-46)$$

$$d'_i = \sqrt{V_i^T * C^{-1} * V_i} \quad (2-47)$$

$$\mu R = \frac{1}{N} \sum_{i=1}^N d'_i \quad \text{y} \quad \sigma R = \sqrt{\frac{1}{N} \sum_{i=1}^N (d_i - \mu R)^2} \quad (2-48)$$

entonces la varianza de elipse se obtiene como:

$$E_{ve} = \frac{\sigma R}{\mu R} \quad (2-49)$$

Rectangularidad

La rectangularidad se representa como la forma rectangular de la imagen y se calcula mediante la siguiente ecuación

$$R = \frac{A_S}{A_R} \quad (2-50)$$

donde A_S es el área de la imagen segmentada y A_R es el área del rectángulo de delimitación mínima.

Casco Convexo

La región convexa H de una imagen segmentada es la región más pequeña que incluye toda la imagen. Es decir, para una región S , el casco convexo, está definido como el polígono convexo más pequeño que contiene a todos los puntos de S . Con el fin de minimizar el ruido, lo más común es suavizar un límite antes de la partición. La Figura 2-3 muestra un ejemplo de un contorno convexo.

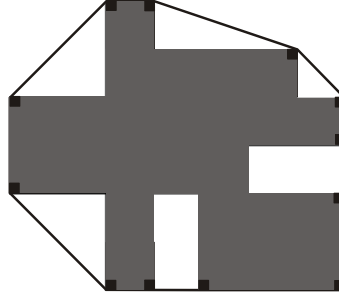


Figura 2-3: Contorno convexo de una región segmentada

Convexidad

La convexidad es definida como la relación del perímetro del casco convexo $O_{convexhull}$ con el contorno original O y está dada por:

$$Convexidad = \frac{O_{convexhull}}{O} \quad (2-51)$$

Solidez

La solidez es la medida en la que una forma es convexa o cóncava y está definida por:

$$Solidez = \frac{A_S}{H} \quad (2-52)$$

donde, A_S es el área de la región de la imagen y H es el área del casco convexo de la imagen.

Número de Euler

El número de Euler describe la relación entre el número de partes conexas y el número de huecos en una forma. En la Figura 2-4 parte a) se muestra una región con dos huecos y en la parte b) se muestra una región con dos componentes conexas. Si asignamos a S es el número de partes conexas y a N el número de huecos. Entonces el número de Euler viene dado por la siguiente expresión:

$$Eul = S - N \quad (2-53)$$

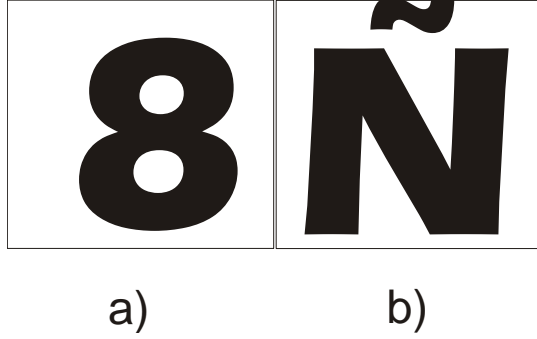


Figura 2-4: Ejemplos de regiones segmentadas con huecos y componentes conexas. a) Región con dos huecos. b) Región con dos componentes conexas

Perfiles

Los perfiles son la proyección del eje x y el eje y en el sistema de coordenadas cartesiano. Para su cálculo se obtienen las siguientes funciones:

$$pro_x(i) = \sum_{j=j \text{ mín}}^{j \text{ máx}} f(i, j) \text{ y } pro_y(j) = \sum_{i=i \text{ mín}}^{i \text{ máx}} f(i, j) \quad (2-54)$$

donde $f(i, j)$ representa la región de la forma.

Relación del área de agujeros

Esta relación es definida como:

$$RAA = \frac{A_h}{A_s} \quad (2-55)$$

donde A_s es el área de la región segmentada y A_h es el área total de orificios en la imagen. Esta relación del área es muy efectiva para diferenciar entre símbolos que tienen orificios grandes y símbolos con orificios pequeños.

Coordenadas complejas

Esta función unidimensional es también llamada *signatura*. Existen varias funciones para representar una *signatura*, entre estas se encuentra la función de coordenadas complejas, la cual

es una representación del contorno de la región segmentada en una función unidimensional que es más sencilla que la función bidimensional para definir el contorno [13].

La función de coordenadas complejas, es un simple número complejo generado con las coordenadas de los puntos del contorno y se obtienen mediante:

$$z(n) = [x(n) - g_x] + i[y(n) - g_y] \quad (2-56)$$

donde (g_x, g_y) es el centroide de la región segmentada dada por ?? ó ??.

Función de distancia del centroide

La función de distancia de centroide es otra *signatura* y se expresa por las distancias de los puntos del contorno al centroide de la región segmentada.

$$r(n) = [(x(n) - g_x)^2 + (y(n) - g_y)^2]^{\frac{1}{2}} \quad (2-57)$$

Debido a la sustracción del centroide, que representa la posición de la imagen de las coordenadas de contorno, Tanto las coordenadas complejas como la función de distancia del centroide son invariantes a la traslación [13].

Ángulo Tangente

Intuitivamente, los ángulos tangentes del contorno de una forma indican el cambio de direcciones angulares de la imagen segmentada. Por lo tanto, la función ángulo tangente en un punto $P_n(x(n), y(n))$, está definida por la dirección tangencial del contorno [13]:

$$\theta(n) = \theta_n = \arctan \frac{y(n) - y(n-w)}{x(n) - x(n-w)} \quad (2-58)$$

donde cada contorno es un curva digital y w es una ventana pequeña para calcular $\theta(n)$ de una manera precisa.

Curvatura de contorno

El contorno es una característica importante para juzgar la similaridad entre imágenes. Con el fin de usar $K(n)$ para representar la región segmentada, la función de curvatura $K(n)$ está dada como:

$$K(n) = \frac{\dot{x}(n)\ddot{y}(n) - \dot{y}(n)\ddot{x}(n)}{(\dot{x}(n)^2 + \dot{y}(n)^2)^{\frac{3}{2}}} \quad (2-59)$$

donde $\dot{x}(n)$ y $\dot{y}(n)$ corresponden a la primera derivada de x y y respectivamente. De igual manera, $\ddot{x}(n)$ y $\ddot{y}(n)$ corresponden a la segunda derivada de x y y respectivamente.

Si n es el parámetro de longitud de arco normalizado, entonces 2-59 puede reescribirse como:

$$K(n) = \dot{x}(s)\ddot{y}(n) - \dot{y}(n)\ddot{x}(n) \quad (2-60)$$

En la Figura 2-5 se muestra una imagen normalizada a 30 puntos y su función curvatura de contorno.

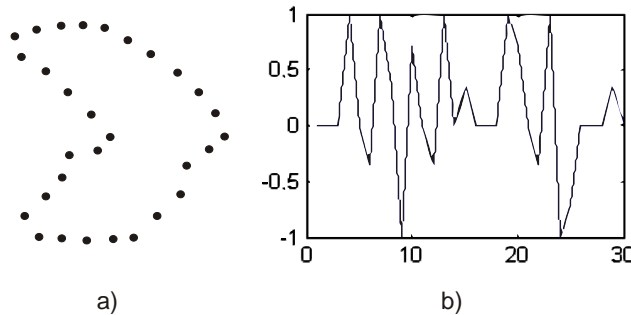


Figura 2-5: Representación de función curvatura de un contorno. a) Contorno normalizado a 30 puntos. b) Función curvatura de contorno de a).

Función de área

Al recorrer los puntos del contorno de la imagen segmentada, se forma un triángulo entre dos puntos consecutivos y el centroide. Esto forma una función de área. Por lo tanto, la función $S(n)$, es el área de dos puntos de contorno sucesivos P_n, P_{n+1} y el centro de gravedad G [13].

Para cada triángulo formado por el centroide, P_n y P_{n+1} , el área está dada por:

$$A(i) = \frac{1}{2} |x_1(i)y_2(i) - x_2(i)y_1(i)| \quad (2-61)$$

La Figura 2-6 muestra un ejemplo de función de área.

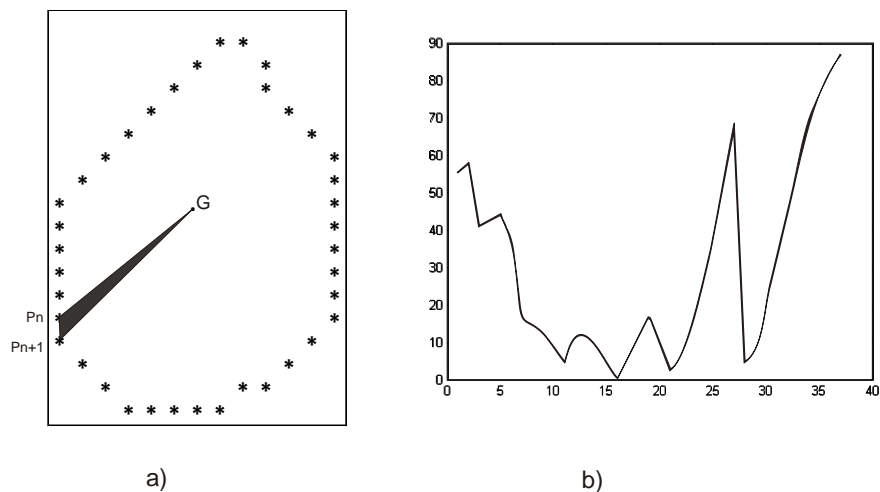


Figura 2-6: Representación grafica de función de área de una región segmentada. a) Región que se forma de dos puntos consecutivos y el centroide. b) Función de área de a).

Representación de área-triángulo

La representación de área-triángulo es calculada mediante el área de los triángulos formados por los puntos del contorno de la imagen segmentada. La curvatura del punto del contorno se mide usando a TAR como sigue.

Para cada tres puntos consecutivos $P_{n-ts}(x_{n-ts}, y_{n-ts})$, $P_n(x_n, y_n)$, $P_{n+ts}(x_{n+ts}, y_{n+ts})$ donde $n \in [1, N]$ y $ts \in [1, \frac{N}{2-1}]$, N es par. La representación de área-triángulo se forma por esos puntos dados por:

$$TAR(n, ts) = \frac{1}{2} \begin{vmatrix} x_{n-ts} & y_{n-ts} & 1 \\ x_n & y_n & 1 \\ x_{n+ts} & y_{n+ts} & 1 \end{vmatrix} \quad (2-62)$$

donde el contorno es recorrido en sentido contrario a las manecillas de reloj.

Función de longitud de cuerda

La función de longitud de cuerda, se deriva de un punto del contorno de la imagen sin usar ningún punto como referencia. Para cada punto del contorno p , la función de longitud de cuerda es la distancia más corta entre p y otro punto del contorno p' , de modo que la línea formada por pp' es perpendicular a la tangente del vector en p .

La función de longitud de cuerda es invariante a la traslación. Sin embargo, es muy sensible al ruido.

Cuadrícula adaptativa

La cuadrícula adaptativa es una cuadrícula lo suficientemente grande para cubrir la figura completa donde cada cuadro puede variar de tamaño, acorde al contenido de la imagen segmentada. Es decir, los cuadros se adaptan a la imagen, de tal forma que existe alta resolución donde la imagen lo requiere y baja resolución donde no es necesario.

La representación de la cuadrícula adaptativa se realiza mediante un árbol de descomposición de cuadros. La descomposición está basada en la subdivisión sucesiva de los cuadros en cuatro cuadrantes iguales. Si el cuadrante no contiene totalmente parte de la imagen, este se subdivide recursivamente en otros cuadros más pequeños hasta alcanzar el objetivo [10]. La Figura 2-7, muestra un ejemplo de cuadrícula adaptativa y su árbol de descomposición.

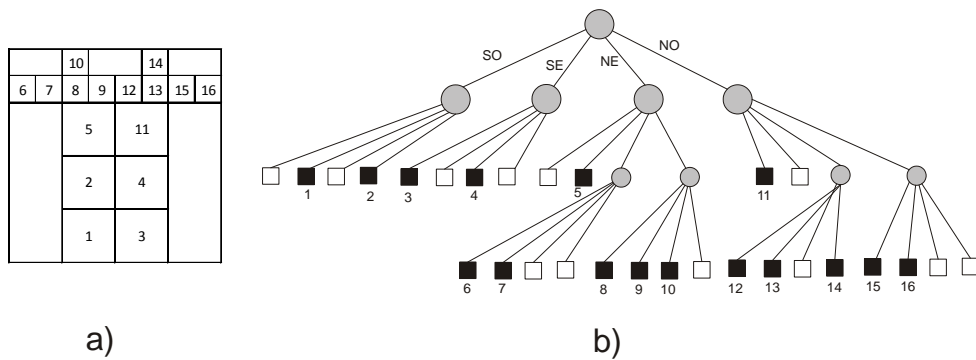


Figura 2-7: Ejemplo de cuadrícula adaptativa de una imagen. a) Imagen con cuadrícula adaptativa. b) Árbol de descomposición de la cuadrícula adaptativa de a).

Código de cadena básico

Los códigos de cadena se utilizan para representar un conjunto de puntos y saber si constituyen una línea recta o no, además sirven para representar una frontera como un conjunto de segmentos con longitud y direcciones específicas. Los movimientos en una curva digital usan dos códigos llamados: 8-direccional y 4-direccional. Para la definición del código de cadena, se tiene en cuenta la localización de un pixel (i, j) y sus 8 vecinos en las direcciones cuantizadas de 45° o sus 4 vecinos en las direcciones cuantizadas de 90° . A cada una de las direcciones se les asigna un valor numérico. Así los números 4, 5, 6, 7, 8, 1, 2 y 3 se utilizan en un código de cadena 8-direccional y 2, 3, 4 y 1 son utilizados en un código de cadena 4-direccional. La Figura 2-8 representa ambos códigos de cadena.

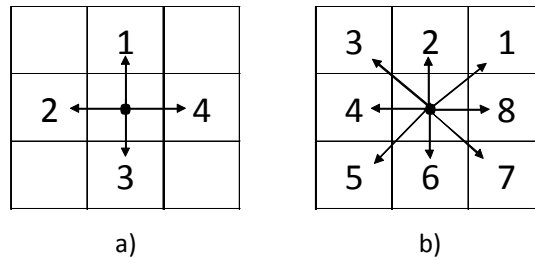


Figura 2-8: Códigos de cadena tomando en cuenta 4 y 8 vecinos. a) Código de cadena 4-direccional; b) Código de cadena 8-direccional

Para cada borde se obtendrá un código de cadena comenzando en un extremo del mismo y terminando en el extremo opuesto.

Código de cadena vértice

Para mejorar el código de cadena básico, en [29] se propuso un código de cadena de vértice (*CCV*). Un elemento del *CCV* indica el número de vértices de celdas que están en contacto con el contorno de la región segmentada. Solo se pueden usar tres elementos para representar el contorno de la imagen que son "1", "2" y "3". La longitud de la cadena es la suma de las longitudes de sus elementos. La Figura 2-9 muestra los elementos del *CCV* para representar el contorno de la imagen.

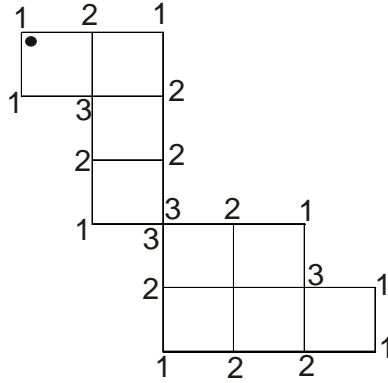


Figura 2-9: Numeración correspondiente para el código de cadena de vértice.

Histograma del código de cadena de vértice

Para cada segmento se construye el correspondiente histograma de códigos de cadena, de manera, que la barra correspondiente a cada número de código representa la frecuencia de aparición del número de código. El *HCCV* se define como:

$$h_i = \#\{i \in M\}$$

donde M es el rango de código de cadena.

Representación simbólica basada en el eje de menor inercia

Esta representación extrae las características con referencia al eje de menor inercia (*EMI*) que es único en la imagen. Una vez que *EMI* es calculado, cada punto en la curvatura de la imagen se proyecta sobre *EMI*. Los dos puntos proyectados más lejanos, se toman como puntos extremos. La distancia Euclidiana entre esos dos puntos, definen la longitud de *EMI*. La longitud de *EMI* es dividida uniformemente por un número n fijo. Los puntos intermedios son llamados puntos característicos. En cada punto escogido, se traza una línea imaginaria. En la Figura 2-10 se puede notar que esas líneas perpendiculares pueden intersectar la imagen de la curva en varios puntos. La longitud de cada línea imaginaria se calcula y el conjunto de esas longitudes de imagen ascendente definen el valor de las características.

Dado que S es la representación de la figura y n el número de puntos característicos escogidos

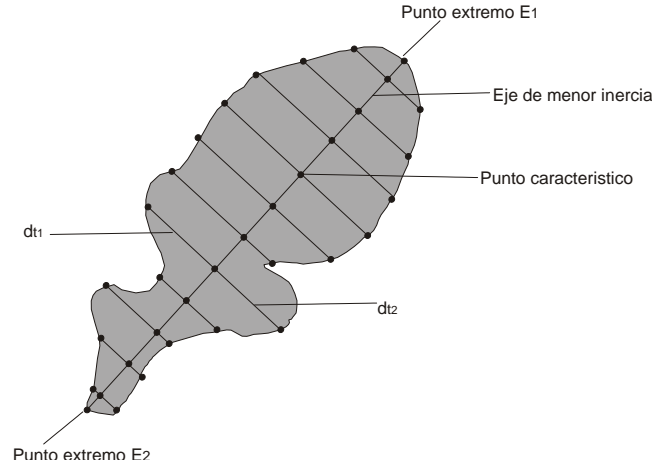


Figura 2-10: Características simbólicas basadas en el eje de menor inercia.

en *EMI*. Entonces el vector de características F representa la figura S , y es en general de la forma:

$$F = [f_1, f_2, \dots, f_t, \dots, f_n]$$

donde $f_t = \{d_{t_1}, d_{t_2}, \dots, d_{t_k}\}$ para $t_k > 1$.

Ángulo estático *Beam*

El ángulo estático *Beam* es un descriptor que se define como el conjunto de líneas que conectan cada punto del contorno con el resto. Si B es el contorno de la imagen segmentada. $B = \{P_1, P_2, \dots, P_N\}$ que está representado por una secuencia de puntos conectados, $P_i = (x_i, y_i)$, $i = 1, 2, \dots, N$, donde N es el número de puntos del contorno. Para cada punto P_i , existe un ángulo entre los ángulos del vector siguientes $V_{i+k} = \overrightarrow{P_i P_{i+k}}$ y los ángulos del vector anteriores $V_{i-k} = \overrightarrow{P_i P_{i-k}}$ en el orden k^{th} para el sistema de vecinos. El ángulo Beam $C_k(i)$ es entonces calculado como:

$$C_k(i) = (\theta_{v_{i+k}} - \theta_{v_{i-k}}) \tag{2-63}$$

donde

$$\theta v_{i+k} = \arctan \frac{y_{i+k} - y_i}{x_{i+k} - x_i} \quad (2-64)$$

$$\theta v_{i-k} = \arctan \frac{y_{i-k} - y_i}{x_{i-k} - x_i} \quad (2-65)$$

El ángulo $C_k(i)$ puede tomar una variable aleatoria con una función de probabilidad densa $P(C_k(i))$. Por lo tanto, los ángulos estáticos *Beam*, pueden proveer una representación compacta de un descriptor de forma. Para este propósito, el momento m^{th} de la variable $C_k(i)$ se define como sigue:

$$E[C_k(i)] = \sum_{k=1}^{\left(\frac{N}{2}\right)-1} C_k^m(i) * P_k(C_k(i)) \quad m = 1, 2, \dots \quad (2-66)$$

donde E indica el valor esperado.

El ángulo estático *Beam* es un descriptor que captura la información perceptual usando la información estática basada en las líneas de los puntos individuales. Este descriptor es bastante estable e invariante a la traslación, rotación y escala.

Matriz de forma cuadrada

Este descriptor es una matriz $M \times N$ que representa una región segmentada. Existen dos modelos de matriz de forma: El modelo Cuadrado y Modelo Polar.

Dada una imagen S , la matriz de forma cuadrada se construye con el siguiente algoritmo [6] :

1. Encontrar el centro de gravedad $G = (gx, gy)$ de la imagen S .
2. Encontrar cada punto $M = (x_m, y_m)$ de $M \in G$ y $d(M, T) = \text{máx } d(A, T)$, donde d es la distancia Euclidiana en R_2 .
3. Construir un cuadro con el centro en T y con el tamaño del lado $2.d(M, T)$. El punto M se encuentra en el centro de un lado.
4. Dividir el cuadro en una matriz $n \times n$.

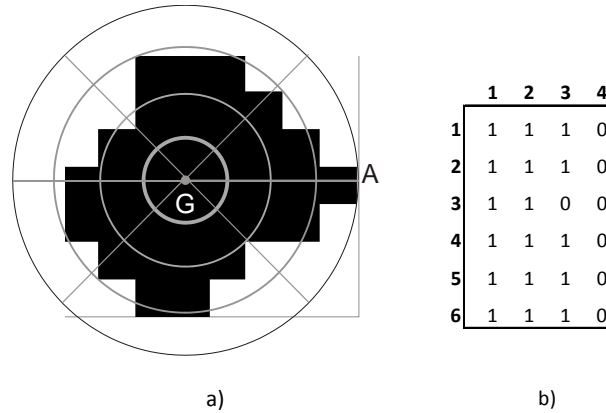


Figura 2-12: Ejemplo de matriz de forma polar. a) Modelo de forma polar. b) Matriz de forma polar correspondiente a a).

Contexto de forma

El contexto de forma ha mostrado ser una poderosa herramienta para el reconocimiento de objetos. Este descriptor encuentra las características correspondientes entre un modelo e imagen. El contexto de forma comienza tomando a N ejemplos de los elementos de la región segmentada que hacen el contorno. Esos puntos pueden ser contornos internos o externos. Considera los vectores originándose de un punto a todos los puntos de ejemplo en la imagen. Esos vectores expresan la apariencia completa relativa para el punto de referencia. Este descriptor es el histograma de las coordenadas polares relativas de todos los puntos:

$$h_i(k) = \#\{Q \neq P_i : (Q - P_i) \in \text{bin}(k)\} \quad (2-68)$$

El contexto de forma se usa frecuentemente para encontrar puntos correspondientes entre dos imágenes segmentadas. Es muy aplicado en el reconocimiento de objetos ya que es un descriptor invariante a la traslación, escalado y rotación.

Distribución de cuerdas

La idea básica de la distribución de cuerda es calcular la longitud de todas las curvas en la imagen segmentada y construir un histograma de sus longitudes y orientaciones. El histograma de longitudes es invariante a la rotación y escalado. El histograma de ángulos es invariante al

tamaño de objetos y desplazamiento rotatorio del objeto.

Shock graphs

El *Shock graphs* es un descriptor basado en el ángulo medio. El ángulo medio ha sido propuesto como una herramienta de forma de abstracción útil para la representación y modelado de imágenes animadas. El esqueleto y los ejes medios han sido usados extensamente para caracterizar objetos satisfactoriamente usando estructuras que están compuestas de líneas o patrones de arco. El ángulo medio es una operación en la imagen que reduce las formas de entrada con representaciones más delgadas.

Un *shock graphs* es un tipo de abstracción que descompone una imagen en un conjunto de partes primitivas jerárquicamente. Los segmentos *shock* son curvas segmentadas del ángulo medio con un flujo monótono y dan una partición más refinada de los segmentos del eje.

Los puntos de esqueletos son primeramente etiquetados acorde a la variación local del radio de función de cada punto. El *shock graph* puede distinguir las figuras pero no el eje medio.

Para calcular la distancia entre dos *shock graphs*, algunos autores emplean un algoritmo de tiempo polinomial. Sin embargo los mismos autores indican que la complejidad computacional es muy alta.

Momentos

Los momentos de contorno pueden ser usados para reducir la dimensión de la representación de un contorno de imagen. Asumiendo que el contorno está representado por un forma $1 - D$, los r^{th} momentos m_r y momentos centrales μ_r se pueden estimar como:

$$m_r = \frac{1}{N} \sum_{i=1}^N [z(i)]^r \quad (2-69)$$

$$\mu_r = \frac{1}{N} \sum_{i=1}^N [z(i) - m_1]^r \quad (2-70)$$

donde $z(i)$ son las coordenadas complejas del contorno y N es el número de puntos del contorno

Los momentos normalizados $m_r = \frac{\mu_r}{(\mu_2)^{\frac{r}{2}}}$ y $\mu_r = \frac{\mu_r}{(\mu_2)^{\frac{r}{2}}}$ son invariantes a la traslación, rotación y escalado. Los descriptores de forma con menos sensibilidad al ruido se pueden obtener de:

$$F_1 = \frac{(\mu_2)^{\frac{1}{2}}}{m_1} \quad (2-71)$$

$$F_2 = \frac{\mu_3}{(\mu_2)^{\frac{3}{2}}} \quad (2-72)$$

$$F_3 = \frac{\mu_4}{(\mu_2)^2} \quad (2-73)$$

Otro métodos basados en forma $1-D$ con una función $z(i)$ como una variable v y creando k contenedores de histograma $p(v_i)$ de $z(i)$ son los r^{th} momentos centrales que se obtiene mediante:

$$\mu_r = \sum_{i=1}^K (v_i - m)^r p(v_i) \quad (2-74)$$

$$m = \sum_{i=1}^N v_i p(v_i) \quad (2-75)$$

La forma general para la función m_{pq} de orden $(p+q)$ de la región de la imagen esta dado como:

$$m_{pq} = \sum_{x=1}^M \sum_{y=1}^N \Psi_{pq}(xy) f(x, y) \quad p, q = 0, 1, 2, \dots \quad (2-76)$$

donde Ψ_{pq} es conocido como momento de ponderación de Kernel o conjunto básico y $f(x, y)$ es la región de la forma.

Momentos invariantes de Hu

Los momentos invariantes son un conjunto de descriptores que permiten reconocer un objeto independientemente de su posición, tamaño y orientación [36]. Tomando a $f(x, y)$ como la intensidad del punto (x, y) en una región. El momento de orden $(p+q)$ para la región se

define como:

$$m_{pq} = \sum_{x=1}^M \sum_{y=1}^N x^p y^q f(x, y) \quad (2-77)$$

donde el sumatorio se toma sobre todas las coordenadas espaciales (x, y) de puntos de la región segmentada.

Utilizando momentos centralizados la fórmula general quedaría:

$$\mu = \sum_{x=1}^M \sum_{y=1}^N (x - \bar{x})^p (y - \bar{y})^q f(x, y) \quad (2-78)$$

$$\bar{x} = \frac{m_{10}}{m_{00}} \quad \bar{y} = \frac{m_{01}}{m_{00}} \quad (2-79)$$

Se definen lo momentos centralizados normalizados como:

$$\eta_{pq} = \frac{\mu_{pq}}{\mu_{00}^k} \quad (2-80)$$

donde $k = (p + q)/2$ y $p + q \geq 2$

Los momentos centralizados normalizados son invariantes a la posición y el escalado. Para conseguir la invariancia ante la rotación, se obtienen los momentos invariantes propuestos por Hu (1962).

$$H_1 = \mu_{20} + \mu_{02} \quad (2-81)$$

$$H_2 = (\mu_{20} - \mu_{02})^2 + 4\mu_{11}^2 \quad (2-82)$$

$$H_3 = (\mu_{30} - 3\mu_{12})^2 + (3\mu_{20} - \mu_{03})^2 \quad (2-83)$$

$$H_4 = (\mu_{30} - \mu_{12})^2 + (\mu_{21} - \mu_{03})^2 \quad (2-84)$$

$$\begin{aligned}
H_5 = & (\mu_{30} - 3\mu_{12})(\mu_{30} - \mu_{12})^2[(\mu_{30} - \mu_{12})^2 - 3(\mu_{21} + 3\mu_{03})^2] + \\
& (3\mu_{21} - \mu_{03})(\mu_{21} + \mu_{03})[3(\mu_{30} - \mu_{12})^2 - (\mu_{21} + 3\mu_{03})^2]
\end{aligned} \tag{2-85}$$

$$H_6 = (\mu_{20} - \mu_{02})[(\mu_{30} + \mu_{12})^2 - (\mu_{21} + 3\mu_{03})^2] + 4\mu_{11}(\mu_{30} + \mu_{12})(\mu_{21} - \mu_{03}) \tag{2-86}$$

$$\begin{aligned}
H_7 = & (3\mu_{21} - \mu_{03})(\mu_{30} + \mu_{12})[(\mu_{30} - \mu_{12})^2 - 3(\mu_{21} + 3\mu_{03})^2] + \\
& (3\mu_{12} - \mu_{30})(\mu_{21} + \mu_{03})[3(\mu_{30} - \mu_{12})^2 - (\mu_{21} + 3\mu_{03})^2]
\end{aligned} \tag{2-87}$$

Momentos de Zernike

Los momentos de Zernike (MZ) se pueden definir como un conjunto de polinomios complejos que conforman una base ortogonal completa en un círculo unitario. Primero, los polinomios complejos se definen como:

$$V_{nm}(\rho, \theta) = V_{nm}(r \cos \theta, \sin \theta) = R_{nm}(\rho) \exp(im\theta) \tag{2-88}$$

donde

$$\rho = \sqrt{x^2 + y^2} \tag{2-89}$$

$$\theta = \arctan\left(\frac{y}{x}\right) \tag{2-90}$$

y

$$R_{mn}(\rho) = \sum_{s=0}^{\frac{(n-|m|)}{2}} (-1)^s \frac{(n-s)!}{s! * \left(\frac{n-2s+|m|}{2}\right)! \left(\frac{n-2s-|m|}{2}\right)!} \rho^{n-2s} \tag{2-91}$$

donde los valores $m = 0, 1, 2, \dots, \infty$ definen el orden y n es un integrador (puede ser negativo o positivo) que describe una rotación sujeto a las siguientes condiciones:

$$m - |n| = \text{par} \quad \text{y} \quad |n| \leq m \quad (2-92)$$

Entonces se define un momento de Zernike como:

$$Z_{nm} = \frac{n+1}{\pi} \sum_r \sum_{\theta} f(\rho \cos \theta, \rho \sin \theta) * R_{mn}(\rho) * \exp(im\theta) \quad p \leq 1 \quad (2-93)$$

Los momentos de Zernike son invariantes a la rotación, sin embargo no son invariantes ante traslación ni escala.

Momentos de Radial Chebyshev

Un momento radial de Chebyshev de orden p y repetición q está definido como:

$$S_{pq} = \frac{1}{2\pi\rho(p, m)} \sum_{r=0}^{m-1} \sum_{\theta=0}^{2\pi} t_p(r) * \exp(-jq\theta) * f(r, \theta) \quad (2-94)$$

donde $t_p(r)$ es el escalado Chebyshev polinomial para una imagen de tamaño $N \times N$:

$$t_0(x) = 1 \quad (2-95)$$

$$t_1(x) = \frac{(2x - N + 1)}{N} \quad (2-96)$$

$$t_2(x) = \frac{(2p - 1)t_1(x)t_{p-1}(x) - (p - 1) \left\{ 1 - \frac{(p-1)^2}{N^2} \right\} t_{p-2}(x)}{p}, \quad p > 1 \quad (2-97)$$

$\rho(p, N)$ es la norma-cuadrada:

$$\rho(p, N) = \frac{N \left(1 - \frac{1}{N^2}\right) \left(1 - \frac{2^2}{N^2}\right) \dots \left(1 - \frac{p^2}{N^2}\right)}{2p + 1}, \quad p = 0, 1, \dots, N - 1 \quad (2-98)$$

y

$$m = \left(\frac{N}{2}\right) + 1. \quad (2-99)$$

El mapeo entre (r, θ) y coordenadas de la imagen (x, y) están dadas por:

$$x = \frac{rN}{2(m-1)} \cos(\theta) + \frac{N}{2} \quad (2-100)$$

$$y = \frac{rN}{2(m-1)} \sin(\theta) + \frac{N}{2} \quad (2-101)$$

Curvatura de espacio y escala

El objetivo original del esquema de escala y curvatura, es simplificar la curva de una imagen, de modo, que su estructura desaparezca conforme se incrementa el parámetro σ . Esta teoría es muy aplicada en el tratamiento de imágenes debido a que suaviza y elimina los pequeños detalles. En particular el espacio y escala de curvatura se enfocan a representar las curvas planas [57]. Su ecuación es la siguiente:

$$\Gamma(\mu) = (x(\mu), y(\mu)) \quad (2-102)$$

Una versión evolucionada para Γ_σ de Γ se puede calcular con:

$$\Gamma_\sigma(\mu) = (X(\mu, \sigma), Y(\mu, \sigma)) \quad (2-103)$$

donde:

$$X(\mu, \sigma) = x(\mu) \otimes g(\mu, \sigma) \quad (2-104)$$

$$Y(\mu, \sigma) = y(\mu) \otimes g(\mu, \sigma) \quad (2-105)$$

donde \otimes es el operador de convolucion y $g(\mu, \sigma)$ denota un filtro Gaussiano con desviación estándar σ que se calcula como sigue:

$$g(\mu, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(\frac{-\mu^2}{2\sigma^2}\right) \quad (2-106)$$

Entonces la curvatura está dada por:

$$k(\mu, \sigma) = \frac{X_\mu(\mu, \sigma)Y_{\mu\mu}(\mu, \sigma) - X_{\mu\mu}(\mu, \sigma)Y_\mu(\mu, \sigma)}{(X_\mu(\mu, \sigma)^2 - Y_\mu(\mu, \sigma)^2)^{\frac{3}{2}}} \quad (2-107)$$

donde

$$X_\mu(\mu, \sigma) = x(\mu) * g_\mu(\mu, \sigma) \quad (2-108)$$

$$X_{\mu\mu}(\mu, \sigma) = x(\mu) * g_{\mu\mu}(\mu, \sigma) \quad (2-109)$$

$$Y_\mu(\mu, \sigma) = y(\mu) * g_\mu(\mu, \sigma) \quad (2-110)$$

$$Y_{\mu\mu}(\mu, \sigma) = y(\mu) * g_{\mu\mu}(\mu, \sigma) \quad (2-111)$$

Esta técnica es adecuada para remover ruido y suavizar una curva como una forma simple de la imagen. Es robusta en cambios de escala y orientación de los objetos, además es confiable y rápida. Sin embargo a pesar de tener tantas ventajas, no siempre da resultados acorde al sistema de visión humano.

Mapa de puntos de intersección

Otra técnica basada en el esquema espacio y escala es el mapa de puntos de intersección. Esta técnica tiene el mismo objetivo de simplificar una imagen, pero en vez de caracterizar la curvatura involucrando las derivadas de segundo orden, éste usa los puntos de intersección entre la curva suave y la curva original. Conforme la desviación estándar del kernel Gaussiano incrementa, el número de puntos de intersecciones disminuye. Para analizar esos puntos restantes, se definen características por un patrón. Este método es más rápido que el método curvatura de espacio escala en imágenes con características equivalentes.

Descriptores de Fourier

El objetivo de los descriptores de Fourier es obtener la forma de una imagen segmentada a partir de sus puntos de contorno en forma continua. Dada una imagen en un plano complejo, donde la parte real se representa por $Re(t)$ y la parte imaginaria por $Im(t)$. Para cualquier contorno se puede definir:

$$u(n) = Re(t) + jIm(t) \quad n = 0, 1, \dots, N - 1 \quad (2-112)$$

Para una figura cerrada, la función se considera periódica o de periodo N y se puede representar por una Serie de Fourier. Dado que las imágenes son discretas, la Serie de Fourier pasa a ser una Transformada Discreta de Fourier, donde sus coeficientes también son discretos. La Transformada Discreta de Fourier $F(u, v)$ de la imagen $f(x, y)$ está dada por:

$$F(u, v) = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y) \exp\left(-2\pi i \left(\frac{ux}{M} + \frac{vy}{N}\right)\right) \quad (2-113)$$

donde $u = 0, 1, 2, \dots, M - 1$ y $v = 0, 1, 2, \dots, N - 1$

Los coeficientes de Fourier se calculan directamente a partir de la posición del trazado del contorno:

$$a(n) = \frac{1}{N} \sum_{k=1}^N u(n) \exp\left(\frac{j2\pi kn}{N}\right) \quad 0 \leq n \leq N - 1 \quad (2-114)$$

Los Descriptores de Fourier son invariantes a la traslación y rotación e independientes del punto de comienzo u_0 . La Tabla 2-1 enuncia las propiedades de los Descriptores de Fourier.

Tabla 2-1: Propiedades de los Descriptores de Fourier

Transformación	Descriptores de Fourier
Normal	$a(k)$
Traslación	$a(k) = a(k) + u_0\delta(k)$
Escalado	$a(k) = \alpha * a(k)$
Punto de comienzo	$a(k) = a(k)e^{\frac{-2\pi n_0 k}{N}}$
Rotación	$a(k) = a(k)e^{\frac{j\theta}{\sigma}}$
Reflexión	$a(k) = a^*(-k)e^{j2\theta_0} + 2\gamma\delta(k)$

La transformada de *Wavelets*

El análisis del descriptor *wavelets* consiste en la descomposición de una curva en componentes de diferentes escalas, de modo que el componente de la escala más gruesa, lleve a la información global más aproximada, mientras la escala más delgada contiene la información de los detalles locales. Resulta complicado calcular los coeficientes de todas las escalas y posiciones, por lo que se recurre a una versión más discreta denominada “La transformada discreta de wavelets”, donde solo se eligen algunas escalas y posiciones [31].

La implementación de este esquema se realiza mediante el uso de filtros, tales filtros son de baja frecuencia y alta frecuencia. El filtrado de baja frecuencia suaviza la imagen mientras que el filtrado de alta frecuencia extrae los bordes. La transformada de *wavelets*, descompone la imagen en cuatro imágenes, el resultado consta de una imagen filtrada a pasa alto en dirección vertical como horizontal, una imagen filtrada a pasa bajo en dirección vertical como horizontal, una imagen filtrada a pasa alto en dirección vertical y pasa bajo en dirección horizontal y otra filtrada a pasa bajo en dirección vertical y pasa alto en dirección horizontal. Los filtros deben ser filtros de reconstrucción perfecta, es decir que cualquier distorsión introducida por la transformación directa debe ser cancelada por la transformación inversa. Los filtros más utilizados para implementar la transformada de *wavelets* son los de *Daubechies* y el de *Haar* cuyos vectores se muestran en la Tabla 2-2.

Tabla 2-2: Vectores base de Haar y Daubechies para filtrado pasa bajo y pasa alto

	Haar	Daubechies
Pasa Bajo	$\frac{1}{\sqrt{2}}[1, 1]$	$\frac{1}{4\sqrt{2}}[1 - \sqrt{3}, 3 - \sqrt{3}, 3 + \sqrt{3}, 1 + \sqrt{3}]$
Pasa Alto	$\frac{1}{\sqrt{2}}[-1, 1]$	$\frac{1}{4\sqrt{2}}[-1 - \sqrt{3}, 3 + \sqrt{3}, -3 + \sqrt{3}, 1 + \sqrt{3}]$

Una vez elegido el tipo de filtro, se obtiene la transformada de wavelets con los siguientes pasos.

1. Realizar la convolución de las filas con el filtro pasa bajo y guardar resultados.
2. Realizar la convolución de las columnas con el filtro pasa bajo, a partir de los resultados del paso 1. Obtener una imagen reducida tomando solo un pixel de cada dos, que genera una versión pasa bajo/pasa bajo de la imagen.

3. Realizar la convolución del resultado del paso 1 con el filtro pasa alto en las columnas. Obtener una imagen reducida tomando solo un pixel de cada dos, obteniendo una imagen pasa bajo/pasa alto.
4. Realizar la convolución de la imagen original con el filtro pasa alto en las filas y guardar el resultado.
5. Realizar la convolución del resultado del paso 4 con el filtro pasa bajo en las columnas. Obtener una imagen reducida tomando solo un pixel de cada dos obteniendo una imagen pasa alto/pasa bajo.
6. Realizar la convolución de las columnas del resultado del paso 4 con el filtro pasa alto. Obtener una imagen reducida tomando solo un pixel de cada dos obteniendo una imagen pasa alto/pasa alto.

La Figura 2-13 muestra cómo se organizan las bandas de frecuencia en una transformada discreta de *Wavelets*

Pasa Bajo / Pasa Bajo	Pasa Bajo / Pasa Alto
Pasa Alto / Pasa Bajo	Pasa Alto / Pasa Alto

Figura 2-13: Localización de las bandas de frecuencia en una Transformada Discreta de *Wavelets* con cuatro bandas. La convención es fila/columna

El descriptor Wavelets tiene propiedades tales como la invariancia a la traslación, rotación y escala.

Transformación radial angular

Esta transformación se basa en el sistema de coordenadas polares donde la función sinusoidal está definida en un disco unitario. Dada una función de la imagen en coordenadas polares $f(\rho, \theta)$,

los coeficientes F_{nm} de orden n y orden m de la transformación radial angular (TRA) se pueden definir como sigue:

$$F_{nm} = \int_0^{2\pi} \int_0^1 V_{nm}(\rho, \theta) f(\rho, \theta) \rho d\rho d\theta \quad (2-115)$$

donde $V_{nm}(\rho, \theta)$ es la función TRA que es separada a lo largo de las direcciones angulares y radiales de modo que:

$$V_{nm}(\rho, \theta) = A_m(\theta) R_n(\rho) \quad (2-116)$$

donde $A_m(\theta)$ es una función que se usa con el fin de alcanzar una invariancia a la rotación y $R_n(\rho)$ es una función de base radial y se definen como:

$$A_m(\theta) = \frac{1}{2\pi} e^{jm\theta} \quad (2-117)$$

$$R_n(\rho) = \begin{cases} 1 & \text{if } n = 0 \\ 2 \cos(\pi n \rho) & \text{if } n \neq 0 \end{cases} \quad (2-118)$$

Incorporación armónica de la Signatura de la forma

Una función armónica se puede obtener mediante el kernel Poisson $P_R(r, \theta)$ y la convolución entre una función del contorno $u(R e^{j\phi})$, definidos por las siguientes ecuaciones:

$$P_R(r, \theta) = \frac{R^2 - r^2}{R^2 - 2Rr \cos(\theta) + r^2} \quad (2-119)$$

$$u(R e^{j\phi}) = u(R e^{jw_0 n}) = s[n] \quad (2-120)$$

donde $s[n]$ es cualquier signatura de forma, mientras que w_0 y ϕ están dados por:

$$w_0 = \frac{2\pi}{N} \quad (2-121)$$

$$\phi = w_0 n \quad (2-122)$$

El kernel Poisson $P_R(r, \theta)$ tiene como característica un filtro *pasa bajos*, donde el radio r esta inversamente relacionado con el ancho de banda del filtro. Por lo tanto, la función armónica u se puede definir como sigue:

$$u(re^{j\theta}) = \frac{1}{2\pi} \int_0^{2\pi} u(Re^{j\phi})P_R(r, \phi - \theta)d\phi \quad (2-123)$$

Transformada \mathfrak{R}

La transformada R está basada en la transformada de Radom, que se determina por un conjunto de proyecciones de líneas tomadas a lo largo de la imagen tomando ángulos distintos. La transformada de Radom está definida como sigue:

$$T_R(\rho, \theta) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y)\delta(x \cos \theta + y \sin \theta)dx dy \quad (2-124)$$

donde $\delta(\cdot)$ es la función delta Dirac:

$$\delta(x) = \begin{cases} 1 & \text{if } x = 0 \\ 0 & \text{otro caso} \end{cases} \quad (2-125)$$

$\theta \in [0, \pi]$ y $\rho \in (-\infty, \infty)$. En otras palabras, la transformada de Radom es la integral de f sobre la línea $L_{(\rho, \theta)}$ definida por $p = x \cos \theta + y \sin \theta$.

En base a lo anterior se puede definir la Transformada \mathfrak{R} como:

$$\mathfrak{R}_f(\theta) = \int_{-\infty}^{\infty} T_R^2(\rho, \theta)d\rho \quad (2-126)$$

donde $T_R(\rho, \theta)$ es la transformada de Radom del dominio de función f .

Descriptor *Shapelets*

Este descriptor tiene el propósito de presentar un modelo para extraer partes significativas de los objetos. El modelo asume que la región segmentada está compuesta por una superposición de línea de un número de bases de imagen. Las funciones básicas están sujetas a una transformación de una matriz de 2×2 :

$$A_k = \begin{bmatrix} a_k & b_k \\ c_k & d_k \end{bmatrix} \quad (2-127)$$

Las variables para describir una base están denotadas por $b_k = (A_k, \mu_k, \sigma_k)$ y son denominadas elementos básicos. Entonces una *shapelet* se define por:

$$\gamma(s; b_k) : A_k \psi(s; \mu_k, \sigma_k) \quad (2-128)$$

Un *shapelet* γ_0 está definido como una elipse. Los *shapelets* están contruidos por bloques de contorno de forma, formando curvas cerradas por adición lineal.

$$\Gamma(s) = \begin{bmatrix} x_0 \\ y_0 \end{bmatrix} + \sum_{k=1}^K \begin{bmatrix} a_k & b_k \\ c_k & d_k \end{bmatrix} \psi(s; \mu_k, \sigma_k) + n(s)$$

Momentos invariantes de Flusser

Los momentos invariantes de Flusser, son un grupo de características estadísticas derivadas de los momentos invariantes de Hu, y al igual que estos, son invariantes a la traslación, rotación y escala. Los momentos de Flusser, también son invariantes bajo transformaciones generales afines, por lo que pueden ser utilizadas para el reconocimiento de objetos deformados [7].

Los momentos invariantes afines, están basados en la teoría de invariantes algebraicos:

$$u = a_0 + a_1x + a_2y \quad (2-129)$$

$$v = b_0 + b_1x + b_2y \quad (2-130)$$

Ya que los momentos de Flusser parten de los momentos utilizados por Hu, los momentos de orden general m_{pq} y los momentos centrales μ_{pq} , se definen en 2-77 2-78 respectivamente.

El invariante más simple, consiste en momentos de segundo orden:

$$I_1 = \frac{(\mu_{20}\mu_{02} - \mu_{11}^2)}{\mu_{00}^4} \quad (2-131)$$

El segundo invariante, consiste en momentos de tercer orden y es el siguiente:

$$I_2 = \frac{(\mu_{30}^2 \mu_{03}^2 - 6\mu_{30} \mu_{21} \mu_{12} \mu_{03} + 4\mu_{30}^3 \mu_{12}^3 + 4\mu_{21}^3 \mu_{03} - 3\mu_{21}^2 \mu_{12}^2)}{\mu_{00}^{10}} \quad (2-132)$$

Finalmente, de los momentos de segundo y tercer orden:

$$I_3 = \frac{(\mu_{20}[\mu_{21} \mu_{03} - \mu_{12}^2] - \mu_{11}[\mu_{30} \mu_{03} - \mu_{21} \mu_{12}] + \mu_{02}[\mu_{30} \mu_{12} - \mu_{21}^2])}{\mu_{00}^7} \quad (2-133)$$

2.3. Extracción de características texturales

El uso de características texturales provienen de la habilidad innata que los seres humanos tenemos para distinguir infinidad de texturas. Por esta razón, el análisis de características texturales, juega un papel importante en la identificación de objetos o regiones de interés en una imagen.

En los últimos años, se han propuesto varias formas de medir la textura en una imagen. Estas medidas se dividen en Primer, Segundo y Tercer Orden. Las medidas de primer orden se calculan a partir del histograma de los niveles de gris de la imagen y estadísticas tales como, la media, desviación estándar etc. En este orden no se toman en cuenta la relación entre vecinos. Por esta razón, se sostuvo que esta medida no era suficiente para una descripción adecuada de textura y se sugirió implementar estadísticas de segundo y tercer orden. Las medidas de segundo orden consideran la relación de dos pixeles en una matriz de co-ocurrencia. Las medidas de tercer orden consideran las relaciones entre 3 y más pixeles en la imagen. Sin embargo, aunque su cálculo es posible, requieren de mucho costo computacional [18].

A continuación se describe como crear una matriz de co-ocurrencia y los descriptores de textura definidos por Haralick.

2.3.1. Matriz de co-ocurrencia

Una característica de textura se puede obtener a partir de la distribución y dependencia espacial entre los tonos de grises en un área local [4]. La matriz de co-ocurrencia describe esa relación espacial que existe entre dos niveles de gris dentro de una región determinada. La relación espacial entre el pixel de referencia y su pixel vecino, puede ser en alguna de las 4 direcciones ($0^\circ, 45^\circ, 90^\circ$ y 135°). La Figura 2-14 muestra los 8 vecinos de un pixel respecto a la

dirección utilizada.

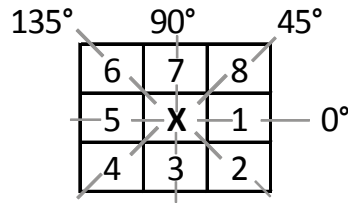


Figura 2-14: Relación espacial entre un píxel de referencia y sus píxeles vecinos en base a 4 direcciones. Las celdas 1 y 5 están en la dirección 0° (horizontal); las celdas 4 y 8 en están en la dirección 45°; las celdas 7 y 3 están en la dirección 90° y las celdas 6 y 2 en la dirección 135°.

Para ejemplificar la construcción de la matriz de co-ocurrencia tomamos como ejemplo la Figura 2-15 parte a) que muestra la matriz de una imagen con 4 niveles de gris (0, 1, 2, 3). La matriz de co-ocurrencia considera una relación espacial entre un píxel de referencia y el píxel vecino, esto se puede expresar como (0, 1), donde 0 es el píxel en la dirección x y 1 el píxel en la dirección y . A partir de los niveles de intensidad de cada imagen, se pueden formar múltiples combinaciones entre el píxel de referencia y el píxel vecino. En la Figura 2-15 parte b) se muestra una tabla con todas las combinaciones posibles que se pueden realizar con 4 niveles de gris. Existen distintas matrices de co-ocurrencia según se considere alguna de las 4 direcciones posibles, ya que dada la dirección, se incluyen píxeles vecinos diferentes. Por ejemplo, si se toma la dirección 0°, se incluyen como vecinos, el píxel izquierdo y derecho del píxel de referencia.

La primera celda de la matriz de co-ocurrencia, se llena con la cantidad de veces que ocurre (0, 0). Es decir, cuantas veces en la matriz de la imagen, un píxel vecino con nivel de gris igual a 0, está situado a la derecha o izquierda del píxel de referencia con nivel de gris 0. En el ejemplo de la Figura 2-15 parte a) se puede notar que el número de veces que ocurre (0, 0) son 4. Las primeras 2 ocurrencias se encuentran en la primera columna, donde dos píxeles de referencia con valor de gris 0, tienen un píxel vecino a la derecha con nivel de gris 0. Las otras 2 ocurren en la segunda columna donde dos píxeles de referencia con valor 0 tienen un píxel vecino a la izquierda con nivel de gris 0. Sin embargo el número de veces que ocurre (0, 1) son 2, ya que solo dos píxeles de referencia con nivel de gris 0 tienen como píxel vecino a la derecha al nivel de gris 1, y no existe ninguna ocurrencia de píxeles de referencia con valor de gris 1 con valores a la izquierda como píxeles vecinos. En la Figura 2-15 parte c) se muestra la matriz de co-ocurrencia

con dirección 0° , la matriz de co-ocurrencia con dirección 90° , la matriz de co-ocurrencia con dirección 135° y la matriz de co-ocurrencia con dirección 45° . También se puede notar que todas las matrices son del mismo tamaño. El tamaño de la matriz de co-ocurrencia está definido por el número de niveles de gris. En el ejemplo se utilizaron 4 niveles, por lo tanto las matrices tienen un tamaño de 3×3 .

No importando la dirección que uno escoja, el resultado debe arrojar una matriz simétrica, que se debe expresar como probabilidad. Esta se puede calcular mediante la siguiente ecuación:

$$C_{i,j} = \frac{V_{i,j}}{\sum_{i,j} V_{i,j}} \quad (2-134)$$

donde i y j describen el número de filas y el número de columnas respectivamente; $V_{i,j}$ es el valor de la celda (i, j) y $C_{i,j}$ es la probabilidad de la celda (i, j) .

2.3.2. Descriptores Haralick

En [3] Haralick desarrolló un modelo para analizar patrones de texturas sobre imágenes, Los descriptores de textura propuestos por Haralick, se pueden calcular a partir de la matriz de co-ocurrencia. Dicha matriz contiene información sobre los patrones de textura, ya que contabiliza las veces que se repite una combinación de valores, dentro de una imagen en una de las 8 direcciones posibles. Las 14 características se describen a continuación:

1) Segundo Momento Angular

También conocida como energía o uniformidad, debido a que mide la uniformidad textural. Cuanto más suave es la textura, mayor valor de uniformidad se tiene. Su ecuación es la siguiente:

$$C_1 = \sum_{i,j} [p(i, j)]^2 \quad (2-135)$$

2) Contraste

Calcula una medida de contraste entre las intensidades de los pixeles analizados. Es decir, cuanto mayor es la variación de los tonos de gris, mayor es el contraste. Esta medida se puede calcular mediante la siguiente ecuación:

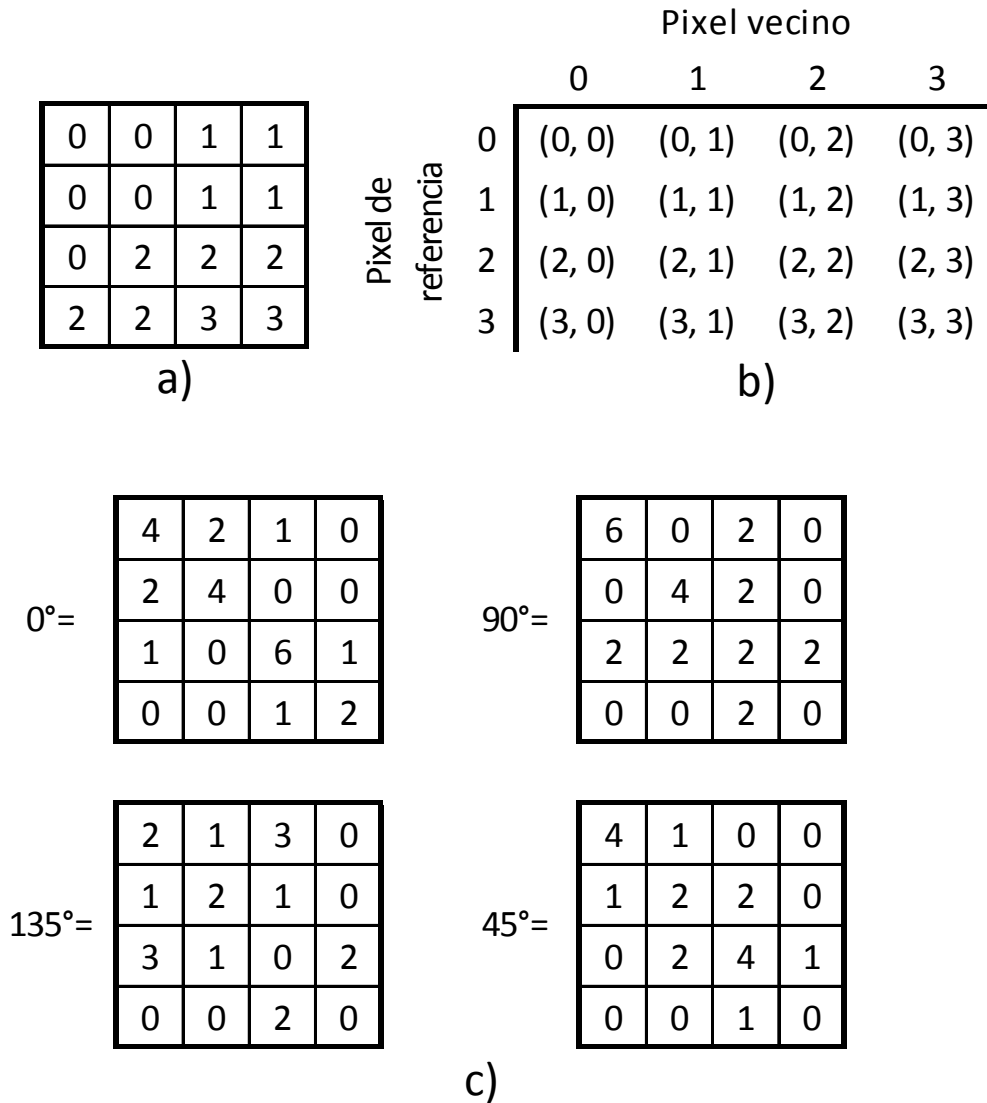


Figura 2-15: Ejemplo de matrices de co-ocurrencia de una imagen. a) Matriz de dimensiones 4x4 con 4 niveles de gris (0, 1, 2, 3). b) Posibles combinaciones de la matriz a). c) Matrices de co-ocurrencia de a) en las 4 direcciones posibles (0°, 90°, 135° y 45°).

$$C_2 = \sum_{i,j} p(i,j)(i-j)^2 \quad (2-136)$$

3) Correlación

Esta medida calcula la medida de correlación de cada pixel con su vecino; si la correlación es 0, no existe correlación lineal entre los niveles de gris. La correlación se define mediante la siguiente ecuación.

$$C_3 = \sum_{i,j} \left[\frac{(ij)p(i,j) - \mu_i\mu_j}{(\sigma_i\sigma_j)} \right] \quad (2-137)$$

donde μ_i , μ_j , σ_i y σ_j son medidas de desviación estándar de p_i y p_j y se definen como:

$$\mu_i = \sum_{i,j} ip(i,j) \quad (2-138)$$

$$\mu_j = \sum_{i,j} jp(i,j) \quad (2-139)$$

$$\sigma_i = \sqrt{\sum_{i,j} (i - \mu_i)^2 p(i,j)} \quad (2-140)$$

$$\sigma_j = \sqrt{\sum_{i,j} (j - \mu_j)^2 p(i,j)} \quad (2-141)$$

4) Varianza

Indica la distribución de tonos de gris en la imagen. Es decir, la variación es grande si los niveles de gris se distribuyen ampliamente. La varianza se calcula mediante:

$$C_4 = \sum_{i,j} (i - \mu)^2 p(i,j) \quad (2-142)$$

5) Momento de diferencia inversa

También conocido como homogeneidad, se refiere a la distribución de pixeles en la imagen. Este descriptor es alto cuando la matriz de co-ocurrencia se concentra a lo largo de la diagonal. La ecuación del momento de diferencia inversa es la siguiente:

$$C_5 = \sum_{i,j} \frac{p(i,j)}{1 + (i-j)^2} \quad (2-143)$$

6) Media

Como su nombre lo indica, calcula la media de los tonos de gris de una imagen. Se espera que este valor sea alto si los tonos de gris son altos. La media se puede calcular con la siguiente ecuación.

$$C_6 = \frac{1}{2} \sum_{i,j} (ip(i,j) + jp(i,j)) \quad (2-144)$$

7) Suma de Varianza

La suma de varianza se puede calcular con la siguiente ecuación:

$$C_7 = \sum_{i=2}^{2N_g} (i - C_8)^2 p_{x+y}(i) \quad (2-145)$$

donde:

$$p_{x+y}(i) = \sum_{\substack{i,j=1 \\ i+j=i}}^{N_g} p(i,j) \quad (2-146)$$

y N_g es el número de los distintos niveles de gris en la imagen.

8) Suma de Entropía

La suma de entropía se puede calcular con la siguiente ecuación:

$$C_8 = - \sum_{i,j} p_{x+y}(i) \log[p_{x+y}(i)] \quad (2-147)$$

9) Entropía

Este descriptor mide la aleatoriedad de los niveles de gris en la imagen. A mayor entropía, mayor complejidad, es decir, alcanza un máximo cuando todos los elementos de la matriz de co-ocurrencia son iguales y toma un valor bajo cuando la textura es suave (menos aleatoria.). Su ecuación es la siguiente:

$$C_9 = - \sum_{i=2}^{2N_g} p(i, j) \log(p(i, j)) \quad (2-148)$$

10) Diferencia de Varianza

La diferencia de varianza se puede calcular con la siguiente ecuación:

$$C_{10} = \text{varianza de } p_{x-y} \quad (2-149)$$

donde:

$$p_{x-y}(k) = \sum_{\substack{i,j=1 \\ |i-j|=k}}^{N_g} p(i, j), \quad k = 0, 1, \dots, N_g - 1. \quad (2-150)$$

11) Diferencia de Entropía

La diferencia de entropía se puede calcular con la siguiente ecuación:

$$C_{11} = - \sum_{i=0}^{N_g-1} p_{x-y}(i) \log\{p_{x-y}(i)\} \quad (2-151)$$

12) Medida de correlación 1

Este descriptor define las medias y desviaciones estándar de p_x y p_y . Su cálculo se realiza con la siguiente ecuación:

$$C_{12} = \frac{HXY - HXY1}{\text{máx}(HX, HY)} \quad (2-152)$$

donde HX y HY son entropías de p_x y p_y , y:

$$HXY1 = - \sum_{i,j} p(i, j) \log[p_x(i)p_y(j)] \quad (2-153)$$

13) Medida de correlación 2

Este descriptor se calcula mediante la siguiente ecuación:

$$C_{13} = (1 - \exp[-2(HXY2 - HXY)])^{\frac{1}{2}} \quad (2-154)$$

donde:

$$HXY2 = - \sum_{i,j} p_x(i)p_y(j) \log[p_x(i)p_y(j)] \quad (2-155)$$

14) Coeficiente de máxima correlación

El coeficiente de máxima correlación se calcula con la siguiente ecuación:

$$C_{14} = (\text{Segundo eigenvalor más grande de } Q)^{\frac{1}{2}} \quad (2-156)$$

donde:

$$Q(i, j) = \sum_k \frac{p(i, k)p(j, k)}{p_x(i)p_y(k)} \quad (2-157)$$

Por cada matriz de co-ocurrencia formada, se obtienen las 14 características de Haralick. Así que, de las 4 matrices, se obtienen un total de 56 valores. Sin embargo de las 14 características, algunas están fuertemente relacionadas, por lo que se puede hacer una reducción a 7 características distintas. Al realizar esta reducción en cada matriz de co-ocurrencia, se obtiene un total de 28 características de textura.

2.4. Extracción de características cromáticas

Las características mencionadas hasta ahora, han sido diseñadas para aplicarse en imágenes a escala de grises. Sin embargo, debido a la creciente disponibilidad de imágenes a color, es importante analizar distintos descriptores cromáticos.

Los colores de una imagen, son a menudo tratados como características que pueden aumentar en gran manera las tasas de clasificación. Sin embargo se ha demostrado que por sí solas no producen un buen rendimiento, por lo que se han fusionado características cromáticas con características acromáticas, ejemplo de esto son las características fusionadas con *SIFT*. Esta desventaja en imágenes a color, podría atribuirse a las variaciones de iluminación y sombras que pueden causar fuertes cambios en la tonalidad y la saturación de la imagen [59].

A continuación se da una introducción de los fundamentos del color, aunado a esto, se describen varias representaciones o modelos de color. Posteriormente se presentan algunos de-

scriptores de color con propiedades de invariancia. En primer lugar, se analizan los descriptores de color basados en histogramas, luego los momentos de color y momentos invariantes de color, finalmente los descriptores de color basados en *SIFT*.

2.4.1. Fundamentos de color

Las características utilizadas generalmente para distinguir un color de otro son: brillo, matiz y saturación. El brillo es la noción cromática de intensidad, el matiz es un atributo asociado con la longitud de onda dominante en la mezcla de longitudes de onda de luz, este es el color percibido por el observador. La saturación se refiere a la pureza relativa o a la cantidad de luz blanca mezclada con un matiz. El matiz y la saturación conjuntamente, se denominan cromaticidad, por lo tanto, un color se puede caracterizar por su brillo y cromaticidad [31].

2.4.2. Modelos de color

Existen dos clases de modelos de color. Unos son los modelos orientados a equipos, tales como cámaras, monitores o pantallas de televisión, denominados modelos sensoriales. Otros son los modelos que se asemejan más a la percepción humana y están orientados al procesamiento de imágenes, estos se denominan modelos perceptuales.

Dentro de los modelos sensoriales de color, existen tres modelos comúnmente utilizados que son: *RGB*, *CMY* y *YIQ*. En los modelos perceptuales están los modelos: *HSV*, *HLS* y *HSI*. Cada uno de estos modelos se describe a continuación:

Modelo *RGB*

Este modelo está basado en el sistema de coordenadas cartesianas y utiliza las componentes primarias rojo, verde y azul. En la Figura 2-16 se muestra un cubo que representa el modelo RGB.

Los valores *RGB* están en tres vértices. Los valores cyan, magenta y amarillo se sitúan en otros tres vértices. El negro corresponde al origen y el blanco se sitúa en el vértice más alejado del origen. La escala de grises se extiende desde el negro al blanco a lo largo de la diagonal punteada. Los colores son puntos dentro del tetraedro definidos por vectores desde el origen.

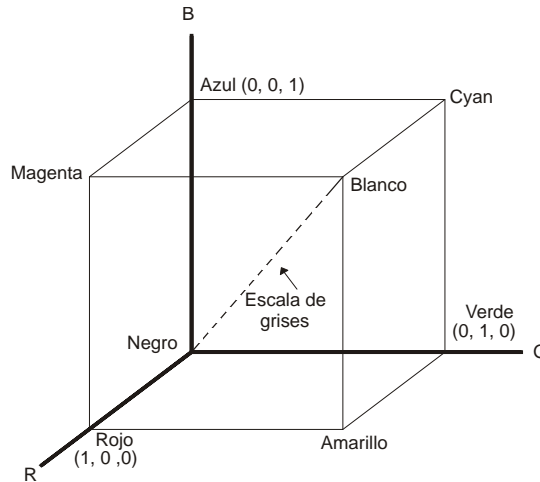


Figura 2-16: Tetraedro de color para el modelo *RGB*.

Los valores representados en el tetraedro están normalizados en el rango $[0, 1]$, siendo válidos para cualquier sensor.

Modelo *CMY*

Este modelo se basa en los colores secundarios (cyan, magenta y amarillo). Dispositivos, tales como las impresoras o copadoras a color, requieren de una entrada *CMY*, por lo que internamente realizan una conversión de *RGB* a *CMY*. Esta conversión se lleva a cabo mediante la ecuación 2-158.

$$\begin{bmatrix} C \\ M \\ Y \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} - \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (2-158)$$

La ecuación 2-158 demuestra que la luz reflejada por una superficie amarilla ($Y = 1$, $C = 0$, $M = 0$) no contiene azul, ya que en este caso $B = 0$, $R = 1$, y $G = 1$, como corresponde realmente el amarillo.

Modelo *YIQ*

El modelo *YIQ* separa la información de intensidad (*Y*) de la información de color (*I*, *Q*). Este modelo es una combinación de los valores *RGB* o viceversa, cuyas expresiones son las siguientes:

$$\begin{bmatrix} Y \\ I \\ Q \end{bmatrix} = \begin{bmatrix} 0,299 & 0,587 & 0,114 \\ 0,596 & -0,274 & -0,322 \\ 0,211 & -0,523 & 0,312 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (2-159)$$

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1,000 & 0,956 & 0,621 \\ 1,000 & -0,272 & -0,647 \\ 1,000 & -1,106 & 1,703 \end{bmatrix} \begin{bmatrix} Y \\ I \\ Q \end{bmatrix} \quad (2-160)$$

Este modelo se utiliza en la televisión comercial, ya que el uso de mayor ancho de banda (bits) para esta información, es más importante para la percepción humana.

Modelo *HSV*

El modelo *HSV* (*Hue*, *Saturation*, *Value*) se obtiene deformando el cubo *RGB*, de modo que se convierte en una pirámide hexagonal invertida. La Figura 2-17 ilustra de forma geométrica el modelo.

En cada esquina del hexágono se encuentran los colores primarios y secundarios. En el centro se encuentra el blanco y en el vértice se encuentra el negro. El eje vertical representa la brillantez o valor (*V*), el eje horizontal la saturación (*S*) y el ángulo de proyección horizontal el croma (*H*). La conversión de *RGB* a *HSV* se realiza con las ecuaciones 2-161 a 2-165.

$$V = M; [0, 1] \quad (2-161)$$

$$\text{Si : } M = m, S = 0; \text{ si no , } S = \frac{(M - m)}{M}; [0, 1] \quad (2-162)$$

$$\text{Si : } m = R, H = \frac{120(B - m)}{(B + G - 2m)}; [0, 360] \quad (2-163)$$

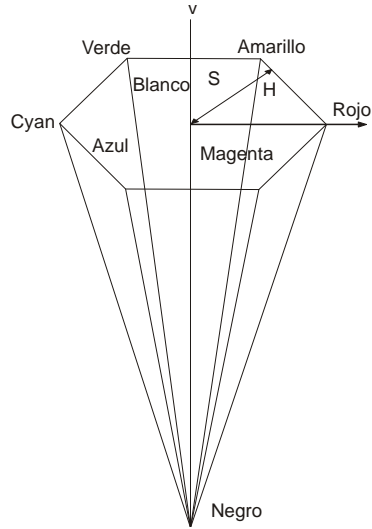


Figura 2-17: Hexágono de color para el modelo *HSV*

$$\text{Si } : m = G, H = \frac{120(R - m)}{(R + B - 2m)}; [0, 360] \quad (2-164)$$

$$\text{Si } : m = B, H = \frac{120(G - m)}{(R + G - 2m)}; [0, 360] \quad (2-165)$$

donde $m = \text{Min}(R, G, B)$ y $M = \text{Max}(R, G, B)$. La brillantez (V) y saturación (S) están normalizadas entre cero y uno. El croma (H) esta normalizado entre 0 y 360 grados.

Modelo HLS

El vértice inferior corresponde al negro, el vértice superior al blanco; el eje vertical representa la brillantez (L), el horizontal la saturación (S) y el ángulo de la proyección horizontal el croma (H).

La transformación del modelo *RGB* al *HLS* se puede obtener a través de las siguientes ecuaciones:

$$L = \frac{(M + m)}{2} \quad (2-166)$$

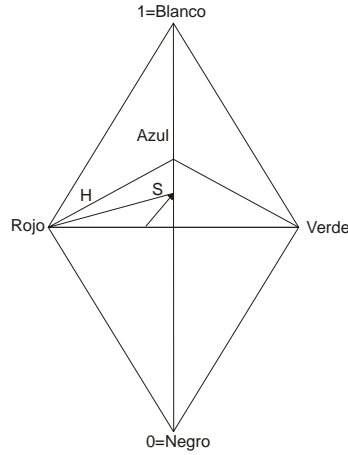


Figura 2-19: Modelo de color HSI

componente de intensidad en este modelo se define como:

$$I = \frac{1}{3}(R + G + B) \quad (2-170)$$

Estos valores pueden escalarse dependiendo de la resolución en número de bits del sensor. Para obtener a H y S se tienen las siguientes expresiones:

$$H = \cos^{-1} \left[\frac{\frac{1}{2}(R - G) + (R - B)}{\sqrt{(R - G)^2 + (R - B)(G - B)}} \right] \quad (2-171)$$

$$S = 1 - \frac{3}{R + G + B} [\text{mín}(R, G, B)] \quad (2-172)$$

2.4.3. Descriptores de color

En la literatura actual [30] [59] [37] [58], se han propuesto descriptores de color que han mostrado éxito en la clasificación de imágenes. Estos descriptores se dividen en tres grupos: descriptores basados en histogramas de color, descriptores basados en *SIFT* y descriptores basados en momentos de color.

Histogramas

Los descriptores locales basados en histogramas, en general, todos aplican a la idea de construir un histograma para cada uno de los canales presentes en la imagen a color y concatenarlos formando un descriptor compuesto. Así, la diferencia entre un descriptor y otro, consiste en el número de *bins*, dimensiones de cada histograma, y del espacio de color en el que se exprese la imagen. Los descriptores basados en histogramas mencionados en la literatura [25] [21] [37] son: Histograma *RGB*, Histograma opuesto, Histograma *Hue*, Histograma-rg, Transformada de distribución del color y un descriptor que ha sobresalido en el reconocimiento de patrones, es el *Hog* piramidal.

Histograma *RGB* El histograma *RGB* es una combinación de tres histogramas 1 – *D* basados en los canales *R*, *G* y *B* del modelo de color *RGB*. Este histograma no posee propiedades de invariancia.

Histograma opuesto El histograma opuesto es una combinación de tres histogramas 1 – *D* basado en los canales opuestos de un espacio de color, la 2-173 muestra el cálculo de para estos espacios.

$$\begin{pmatrix} O1 \\ O2 \\ O3 \end{pmatrix} = \begin{pmatrix} \frac{R-G}{\sqrt{2}} \\ \frac{R+G-2B}{\sqrt{6}} \\ \frac{R+G+B}{\sqrt{3}} \end{pmatrix} \quad (2-173)$$

La intensidad está representada en el canal *O3* y la información del color esta en los canales *O1* y *O2*. Debido a la operación resta en *O1* y *O2*, las compensaciones se cancelan si estos son iguales para todos los canales (Por ejemplo, una fuente de luz blanca). Por lo tanto, este modelo de color es invariante a la traslación respecto a la intensidad de luz. Sin embargo el canal *O3* no tiene propiedad de invariancia [37].

Histograma *Hue* Este histograma está basado en la tonalidad del modelo de color *HSV*. Sin embargo, el tono se vuelve inestable alrededor de los ejes grises. Por esta razón, en [25] el autor aplica un análisis del error de tonalidad. Dicho análisis utiliza los valores de saturación para ponderar los contenedores del histograma de tonalidad. Por lo tanto el histograma de tonalidad

se hace más robusto. Los espacios de color H y S son invariantes a la traslación y escala respecto a la intensidad de luz.

Histograma-rg En el modelo RGB normalizado, las componentes r y g describen la información del color de la imagen:

$$\begin{pmatrix} r \\ g \\ b \end{pmatrix} = \begin{pmatrix} \frac{R}{R+G+B} \\ \frac{G}{R+G+B} \\ \frac{B}{R+G+B} \end{pmatrix} \quad (2-174)$$

Debido a la normalización, r y g son invariantes a la escala respecto a los cambios de luz y sombreado.

Transformada de distribución del color Un histograma RGB no es invariante a cambios en las condiciones de iluminación. Sin embargo, con la normalización en la distribución de valores de pixel, se logra una invariancia a escala y traslación respecto a la intensidad de luz. Debido a que cada canal se normaliza de manera independiente, el descriptor también es normalizado contra los cambios de color de luz. Por lo tanto, la transformación de distribución de color se puede obtener mediante:

$$\begin{pmatrix} R' \\ G' \\ B' \end{pmatrix} = \begin{pmatrix} \frac{R-\mu_R}{\sigma_R} \\ \frac{G-\mu_G}{\sigma_G} \\ \frac{B-\mu_B}{\sigma_B} \end{pmatrix}$$

donde μc es la media y σc es la desviación estándar de cada distribución de canal C .

Histograma de gradientes orientados (HOG) Otro descriptor basado en histogramas utilizado en la literatura actual [21] [30] [45], es el Histograma de Gradientes Orientados (HOG por sus siglas en inglés). HOG es una ventana basada en descriptores locales para detectar puntos de interés. La ventana es centrada sobre el punto de interés y dividida en una cuadrícula ($n \times n$). Una vez centrada la ventana, se calcula una frecuencia de histograma dentro de cada celda de la cuadrícula, para representar las distribuciones de orientación de borde. Las orientaciones de borde se calculan como el arco tangente $\left(\frac{\delta I}{\delta y} / \frac{\delta I}{\delta x}\right)$ y son cuantificadas en q bins

(contenedores). Cada histograma es concatenado para formar un vector $q - D$ por cada celda, que a la vez es concatenado para formar un vector $qn^2 - D$ para la ventana. Posteriormente todas las ventanas son mostradas en una cuadrícula local $w \times w$ no solapadas para cada punto de interés y de nuevo concatenadas para la salida del descriptor final.

HOG piramidal Para implementar *HOG* piramidal, cada imagen se divide en una secuencia de cuadros para k niveles como lo muestra la Figura 2-20. Entonces el descriptor consiste en realizar un Histograma de Gradientes Orientados, en cada subregión de la imagen en cada nivel de resolución, formando una pirámide de Histogramas de gradientes orientados (*PHOG*). La distancia entre dos descriptores de imagen *PHOG* refleja el grado en que las imágenes contienen formas similares y corresponden a su disposición espacial.

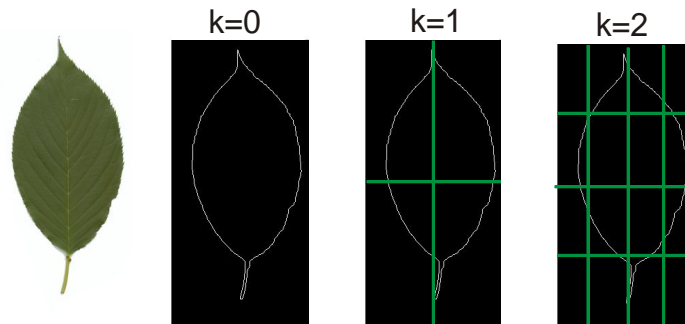


Figura 2-20: Imagen dividida en cuadros para los niveles $k = 0$ a $k = 2$.

Descriptores de color *SIFT*

El grupo de descriptores basados en *SIFT*, consiste en descriptores compuestos a partir de la concatenación de descriptores *SIFT* extraídos de los canales individuales de cada color. En este grupo se encuentran los descriptores *HSV-SIFT* [32], *Hue-SIFT* [25] y *Opponent-SIFT*[37].

SIFT En [19], Lowe propone un modelo llamado *SIFT* (*Scale- Invariant Feature Transform*, por sus siglas en ingles). Su nombre es, debido a que transforma los datos de una imagen en coordenadas de escala, invariantes con respecto a las características locales. Para que el modelo sea invariante a la escala, lo que se hace es construir una pirámide de imágenes, donde en cada

nivel se aplica un filtro (basado en Laplaciano de Gaussiana) cada vez más grueso y se aumenta el valor de σ de la función Gaussiana, hasta pasar todas las escalas. De este modo, se obtienen los puntos de interés cuantificados mediante histogramas. El descriptor toma la forma de un vector de 128 dimensiones que codifica valores del gradiente de brillo.

Un aspecto importante de este modelo es que genera un gran número de características que cubren densamente la imagen sobre toda la gama de escalas y ubicaciones. Una imagen de 500×500 píxeles dará lugar a aproximadamente 2000 características estables (aunque el número depende del contenido de la imagen y otros parámetros). Sin embargo a pesar de ser un descriptor lento ha demostrado cumplir con su objetivo adecuadamente.

HSV-SIFT Un enfoque en [32], computa descriptores *SIFT* sobre los tres canales del modelo *HSV*, produciendo un descriptor por cada canal, que da lugar a 128 descriptores por canal. Este descriptor es totalmente invariante a escala. Sin embargo, dado que los descriptores *SIFT* no son invariantes a los cambios de condiciones de iluminación en la imagen, hacen que este descriptor sea solo parcialmente invariante a los cambios de color de luz. Comúnmente se utiliza el modelo *HSV* sobre los otros, debido a las similitudes con la forma en que los seres humanos tienden a percibir el color y porque es menos sensible a la sombra.

Hue-SIFT Esta técnica combina el descriptor *SIFT* con un histograma de tonalidad del modelo *HSV*. Este canal de tonalidad presenta un comportamiento inestable para los píxeles de color que se encuentran cerca del eje de grises. Para solucionar esto, el modelo utiliza los valores de saturación para ponderar los contenedores del histograma de tonalidad. Así, esa tonalidad mejora su fiabilidad.

Opponent-SIFT Esta técnica describe todos los canales en el espacio de color oponente (ver Ec. 2-173) usando descriptores *SIFT*. La información en el canal *O3* es igual a la información de la intensidad, mientras que los otros canales describen la información del color en la imagen. Los canales *O1* y *O2* contienen algo de información de intensidad por lo tanto no son invariantes a cambios de intensidad de luz.

Momentos de color y momentos de color invariantes

En [20], Mindru propone un conjunto de momentos generalizados de color. Considerando una imagen a color en formato *RGB* como una función:

$$I : (x, y) \mapsto (R(x, y), G(x, y), B(x, y))$$

De acuerdo a la función de la imagen, es posible aplicar el concepto matemático de momentos *I*. Mindru en [20] definió momentos generalizados de color M_{pq}^{abc} :

$$M_{pq}^{abc} = \int \int x^p y^q [I_R(x, y)]^a [I_G(x, y)]^b [I_B(x, y)]^c dx dy. \quad (2-175)$$

M_{pq}^{abc} se conoce como un momento generalizado de color de orden $p + q$ y de grado $a + b + c$. Los momentos de orden 0, son invariantes a la rotación, mientras que los de orden más alto no lo son. Generalmente se consideran momentos generalizados de primer y segundo orden.

Momentos de color Los momentos de color, abarcan los momentos de color generalizados hasta el segundo grado y de primer orden. Estos pueden tener nueve combinaciones posibles para el grado, las cuales son: M_{pq}^{100} , M_{pq}^{010} , M_{pq}^{001} , M_{pq}^{200} , M_{pq}^{110} , M_{pq}^{020} , M_{pq}^{011} , M_{pq}^{002} , y M_{pq}^{101} . Realizando las tres combinaciones posibles para el orden: M_{00}^{abc} , M_{10}^{abc} y M_{01}^{abc} se forman 27 dimensiones. Estos momentos de color solo son invariantes a la traslación.

Momentos de color invariantes Los momentos de color invariantes, se pueden construir a partir de los momentos de color generalizados. Aquí se consideran todos los invariantes de las 3 bandas. Se consideran los invariantes C_{02} , dando un total de 24 momentos de color invariantes a todas las propiedades.

Capítulo 3

Clasificadores

La idea básica de la clasificación, consiste en detectar o reconocer un objeto en una o más imágenes de una escena en términos de propiedades o rasgos [44]. El reconocimiento de patrones es una de las tareas más importantes en el análisis de imágenes. Sin embargo, también es una de las tareas más complejas. En la literatura actual, no existe un clasificador que sea capaz de afrontar todas las situaciones que se pudieran presentar en una imagen. El nivel de reconocimiento humano, sigue siendo inalcanzable para los métodos creados. Aunque la aportación de los métodos propuestos han sido bastante significativos.

En este Capítulo se describen algunos de los clasificadores más importantes en la literatura, que han demostrado ser efectivos. Por un lado, se describen las técnicas estadísticas que son clásicas en la clasificación de patrones, ya que cuentan con un fundamento sólido y han sido utilizadas durante varias décadas en la solución de múltiples problemas, tales como la clasificación de documentos o filtros de mensajes en un correo electrónico. Por otro lado, se describen las técnicas basadas en el funcionamiento de redes neuronales artificiales. Al igual que las técnicas estadísticas, las redes neuronales artificiales han sido aplicadas en la detección de reconocimiento de patrones, aunque estas también se han utilizado en otros campos. Continuamos con los árboles de decisión que también han mostrado ser parte importante en la clasificación. Por último se describen las máquinas de vectores de soporte que son bastante populares dado a que abordan problemas con un conjunto de datos que no son linealmente separables o que contienen mucho ruido.

3.1. Clasificadores estadísticos

La clasificación es sencilla cuando las clases presentan una buena separación o cuando a pesar de estar cerca una de otras, sus desviaciones son pequeñas. Sin embargo, muchas clases presentan una dispersión notable o una dispersión significativa respecto a su media. Para estos casos, es conveniente adoptar una hipótesis estadística. Dentro de los clasificadores estadísticos, los más utilizados son los clasificadores Bayesianos.

3.1.1. Clasificación Bayesiana

La teoría de decisión de Bayes, está basada en el supuesto de que el problema de la decisión se enfoca en términos probabilísticos y que todas las probabilidades relevantes resultan conocidas [31].

Desde un enfoque Bayesiano, el problema de clasificación consiste en asignar a un objeto descrito por un conjunto de características, $X_1, X_2, X_3, \dots, X_n$, a una de m clases posibles, $C_1, C_2, C_3, \dots, C_m$, tal que la probabilidad de la clase dados los atributos se maximiza:

$$\text{Max } P(C | X) \quad (3-1)$$

X es el conjunto de atributos y se denota como: $X = \{X_1, X_2, X_3, \dots, X_n\}$.

Utilizando el teorema de Bayes para calcular la probabilidad de la clase dados los atributos tenemos:

$$P(C | X) = \frac{P(C)P(X | C)}{P(X)} \quad (3-2)$$

donde

$$P(X) = \sum_{j=1}^C P(X | C)P(C) \quad (3-3)$$

Para determinar la clase donde debe clasificarse un patrón dado X , la regla de decisión viene establecida por:

$$X \longrightarrow C_i \text{ si y solo si } P(C_i | X) > P(C_j | X), \quad \forall i \neq j, j = 1, 2, 3, \dots, n \quad (3-4)$$

En otras palabras, el patrón X es asignado a la clase C_i en caso de que su probabilidad a posteriori sea máxima.

3.1.2. Naive Bayes

El clasificador Naive Bayes se basa en la suposición de que todos los atributos son independientes dada la clase.

La aplicación directa de la ecuación 3-2, resulta en un sistema muy complejo al implementarlo en una computadora, ya que el término $P(X_1, X_2, X_3, \dots, X_n | C)$, incrementa exponencialmente de tamaño en función del número de atributos; resultando impráctico para un conjunto con gran número de ejemplos. Por esta razón, se recurrió a la hipótesis de independencia condicional con el objetivo de poder factorizar la probabilidad. Esta hipótesis dice lo siguiente:

“Los valores X_i que describen un atributo de un ejemplo cualquiera Z , son independientes entre sí, si se conoce el valor de la categoría a la que pertenecen. Así, la probabilidad de observar la conjunción de atributos X_i dada una categoría a la que pertenecen, es justamente el producto de las probabilidades de cada valor por separado” [60]. Bajo estas consideraciones, la ecuación 3-2 puede reescribirse como:

$$P(C | X) = \frac{P(C) \prod_i P(X_i | C_j)}{P(X)} \quad (3-5)$$

El clasificador Naive Bayes reduce drásticamente la complejidad del clasificador Bayesiano en espacio y tiempo de cálculo.

3.2. Redes Neuronales

Los clasificadores estadísticos son útiles para el caso de patrones linealmente separables. Sin embargo, la mayoría de problemas no presentan esa separación lineal. Aunque otro factor

que pudiera presentarse, es que las propiedades estadísticas de los clasificadores lineales no se conozcan o no puedan ser estimadas. En ambos casos se debe adoptar esquemas que puedan ser entrenados en base a muestras de cada una de las clases. Uno de los esquemas que permite esto, son las redes neuronales artificiales.

Los primeros modelos de redes neuronales datan de principios de los 40's cuando los investigadores Warren McCulloch y Walter Pitts propusieron el primer modelo simple de una neurona. Más adelante, en las décadas de los 50's y 70's, el movimiento de las redes neuronales fue retomado por B. Widrow y M. E. Hoff, quienes trabajaron una maquina llamada *Adaline* [38].

3.2.1. Elementos de una red neuronal.

Una red neuronal artificial (RNA) es una sociedad de pequeños elementos de cálculo llamados neuronas que en su conjunto, integran la red neuronal. Estos elementos pueden ser usados de forma adaptable e iterada para obtener los pesos de las funciones de decisión que han de permitir clasificar patrones de clases lineal o no linealmente separables [44]. Los elementos básicos de una RNA son los siguientes:

- Conjunto de unidades de procesamiento (neuronas).
- Conexiones entre unidades.
- Funciones de salida o activación.

Para ejemplificar el esquema, la Figura 3-1 muestra el esquema básico de una RNA.

3.2.2. El Perceptrón

Los perceptrones de una capa pueden clasificar correctamente los conjuntos de datos que son linealmente separables, esto es, que se pueden separar por un hiperplano. Esto puede verse como una línea que separa dos clases Figura 3-2.

Un perceptrón convencional tiene un función de no linealidad binaria. El algoritmo de aprendizaje del perceptrón solo funciona para aprender funciones binarias linealmente separables ya que de otra manera no convergería a una mejor solución. La Figura 3-3 muestra un esquema

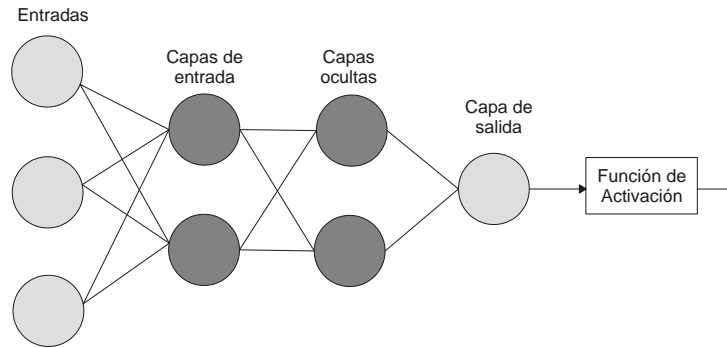


Figura 3-1: Esquema general de una RNA

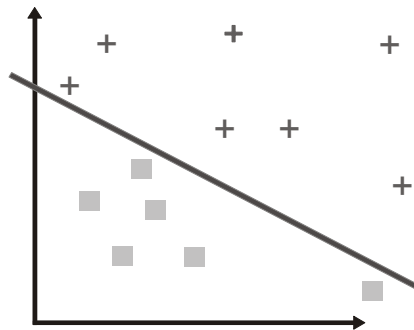


Figura 3-2: Ejemplo de separación lineal para dos clases

del modelo del perceptrón para dos clases, c_1 y c_2 . La respuesta del perceptrón está basado en la suma de *pesos* con sus entradas x_i , $i = 1, 2, \dots, n$. Esto es:

$$fd(x) = \sum_{i=1}^n w_j x_i + \theta w_{n+1} \quad (3-6)$$

La sumatoria \sum define un hiperplano que cruza el origen. Los *pesos* w_j , $j = 1, 2, \dots, n$, modifican las entradas antes de que sean sumadas y suministradas al elemento umbral. En este sentido, los pesos son similares a las sinapsis en el sistema neuronal humano. La introducción de un término θ nombrado *bias* en la sumatoria del hiperplano tiene mayor movilidad y permite la separación de los conjuntos de datos, con lo que se obtienen mejores clasificaciones. La función que trasforma la salida de la sumatoria en salida final se denomina *función de activación*:

$$y = \begin{cases} 1 & \text{si } fd(x) \geq 0 \\ 0 & \text{si } fd(x) < 0 \end{cases} \quad (3-7)$$

De la ecuación 3-7, se puede ver que cuando $fd(x) \geq 0$, la función de activación del perceptrón produce una salida cuyo valor es 1, indicando que el patrón de entrada x ha sido reconocido como perteneciente a la clase c_1 , y viceversa, cuando $fd(x) < 0$, la función de activación del perceptrón ocasiona que la salida sea un valor de 0, indicando que el patrón de entrada x ha sido identificado como perteneciente a la clase c_2 .

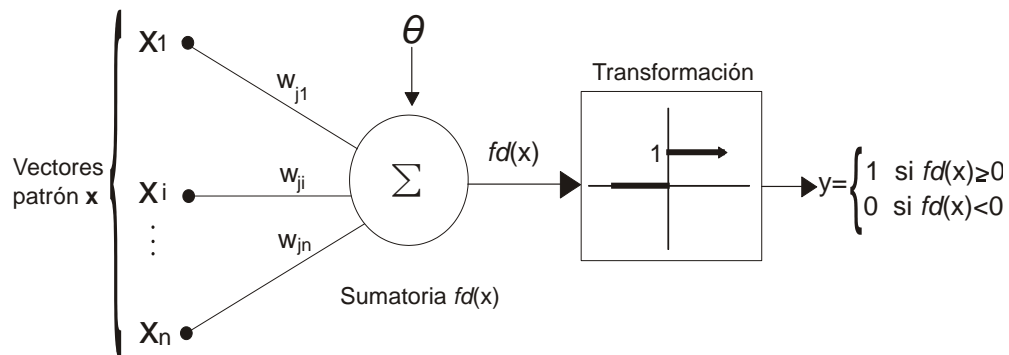


Figura 3-3: Modelo de un perceptrón para dos clases.

El objetivo del método de aprendizaje consiste en obtener los pesos w_j , con el fin de dis-

criminar entre dos clases c_1 o c_2 . Uno de las técnicas utilizadas, es la denominada *regla delta de mínimos cuadrados*.

En síntesis el algoritmo del perceptrón tal y como se deduce del proceso anterior es el siguiente:

1. Fijar valores aleatorios a cada uno de los pesos w_j , $j = 1, 2, \dots, n$ y al *bias* θ .
2. Presentar un nuevo patrón de entrada $x_i = \{x_1, x_2, \dots, x_n\}$ junto con la salida esperada $fd(x)$.
3. Calcular la salida de la neurona $y(k) = f(w^t x_i - w_{n+1})$.
4. Actualizar los pesos $w_i(k+1) = w_i(k) + \alpha(k)[fd_i(k) - y(k)]x_i$.
5. Regresar al paso 2 hasta la convergencia.

Al actualizar los pesos en cada iteración, se mejora la exactitud de clasificación y el sistema comete menos errores. Por lo tanto, los cambios en los pesos se vuelen menos frecuentes. La convergencia se da cuando todos los patrones de ambas clases han sido clasificados correctamente.

3.2.3. El Perceptrón Multicapa

Cuando varios perceptrones son acomodados en varias capas sucesivas se obtiene lo que se conoce como una red neuronal multicapa de perceptrones [44]. La Figura 3-4 muestra el modelo de una red multicapa.

En esta arquitectura, cada capa se compone de una matriz de pesos w , un vector de *bias* b , y un vector de salida s . Cada capa puede contener un número diferente de neuronas. Se puede notar, que desde la primera hasta la penúltima capa constituyen la entrada de la siguiente capa. La matriz de pesos entre la capa k y la capa j viene dada como:

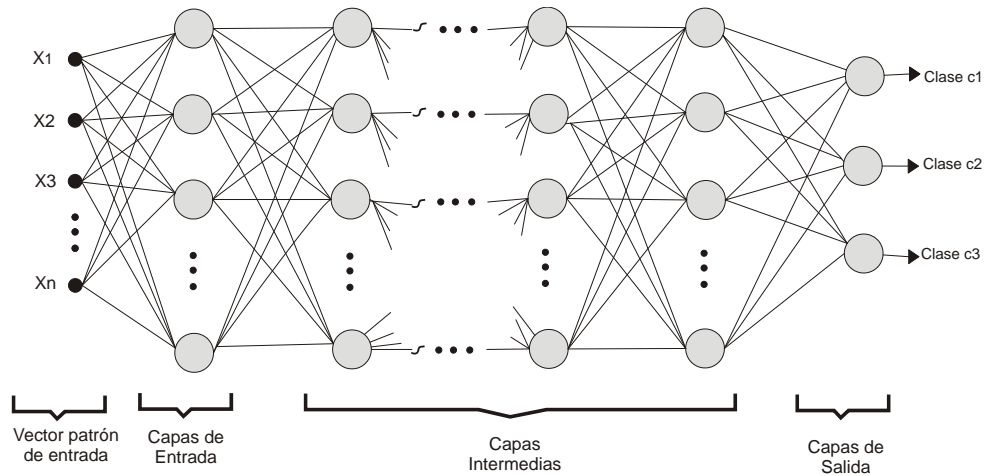


Figura 3-4: Modelo de una red neuronal multicapa

$$w_{jk} = \begin{bmatrix} w_{11} & w_{12} & \cdot & \cdot & \cdot & w_{1n_k} \\ w_{21} & w_{22} & \cdot & \cdot & \cdot & w_{2n_k} \\ \cdot & \cdot & \cdot & & & \cdot \\ \cdot & \cdot & & \cdot & & \cdot \\ \cdot & \cdot & & & \cdot & \cdot \\ p_{n_j1} & p_{n_j2} & & & & p_{n_jn_k} \end{bmatrix}$$

Cada una de las capas juega un papel singular. Por ejemplo, la primera capa, recibe como entrada el vector x , y se denomina capa de entrada. La última capa entrega el resultado final de la red completa y se denomina capa de salida. Las capas entre la de entrada y salida procesan resultados intermedios y se denominan capas ocultas.

Las redes neuronales multicapa son muy útiles para resolver muchos tipos de problemas, ya que estas, pueden obtener regiones de decisión mucho más complejas que las obtenidas con un simple perceptrón. Por ejemplo, con dos capas se pueden resolver problemas de separación convexas como los mostrados en la Figura 3-5 y claramente se puede notar que no pueden ser resueltos con un perceptrón.

En teoría, las redes neuronales con tres capas y un número apropiado de neuronas por capa, puede modelar cualquier superficie de separación.

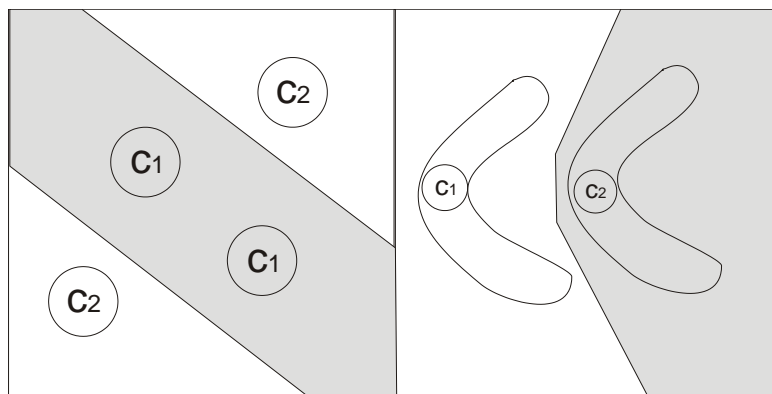


Figura 3-5: Problemas de clasificación que pueden ser resueltos con una RNA multicapa con dos capas.

3.3. Árboles de decisión

Un árbol de decisión es un conjunto de condiciones o reglas organizadas en una estructura jerárquica, de tal forma que la decisión final se puede determinar siguiendo las condiciones que se cumplen desde la raíz hasta alguna de sus hojas. El árbol tiene entradas que pueden ser un conjunto de descriptores y a partir de estos devuelve una respuesta que es una decisión tomada a partir de las entradas. Las entradas pueden ser valores discretos o continuos, pero normalmente se utilizan valores discretos por simplicidad [33].

El árbol de decisión lleva a cabo un test a medida que recorre las hojas para alcanzar así una decisión. Dentro del árbol de decisión existen nodos internos, nodos de probabilidad, nodos hojas y arcos. Cada nodo en el árbol especifica una prueba de algún atributo de los casos, y cada rama descendente de ese nodo, corresponde a uno de los valores posibles para este atributo. La Figura 3-6 ilustra un árbol de decisión típico. Este árbol de decisión, clasifica los días en función de si son adecuados para la práctica de Tennis.

Normalmente, los arboles de decisión se usan en los sistemas expertos porque son más precisos que el hombre para poder desarrollar un diagnostico con respecto a una situación, ya que el hombre puede dejar pasar sin querer algunos detalles. En cambio la maquina mediante un sistema experto con un árbol de decisión puede dar un resultado exacto. Sin embargo, la desventaja del árbol, es que puede llegar a ser más lento ya que analiza todas las posibilidades posibles.

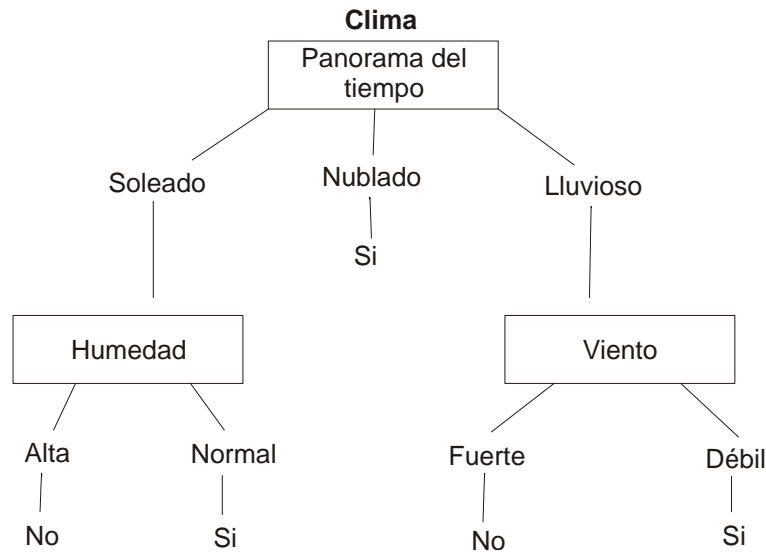


Figura 3-6: Árbol de decisión para determinar si es un día apropiado para jugar Tennis

A continuación se describen algunos árboles de decisión que influyen en este proyecto.

3.3.1. ID3

ID3 es un algoritmo iterativo que elige al azar un subconjunto de datos a partir del conjunto de datos de entrenamiento y construye un árbol de decisión a partir de ello. El árbol debe clasificar de forma correcta todos los casos de entrenamiento. Usando este árbol intenta clasificar a todos los demás casos en el conjunto completo. Si el árbol consigue clasificar el subconjunto, entonces será correcto para todo el conjunto de datos y el proceso termina. En caso contrario, se incorpora al subconjunto una selección de los casos que ha conseguido clasificar correctamente y se repite el proceso [33].

ID3 puede hallar el árbol correcto con pocas iteraciones, procesando un conjunto de datos. Para elaborar un árbol de decisión ID3 se tienen que tomar en cuenta las siguientes reglas.

- Cada nodo corresponde a un atributo y cada rama al valor posible de ese atributo. Una hoja del árbol, especifica el valor esperado de la decisión de acuerdo con los ejemplos dados. La explicación de una determinada decisión, viene dada por la trayectoria desde la raíz a la hoja representativa de esa decisión.

- A cada nodo es asociado aquel atributo más informativo que aún no haya sido considerado en la trayectoria desde la raíz.
- Para medir cuán informativo es un atributo, se emplea el concepto de *entropía*. Cuanto menor sea el valor de la entropía, menor será la incertidumbre y más útil será el atributo para la clasificación.

La *entropía* se puede definir como sigue: Dada una colección S , que contiene ejemplos positivos y negativos de algún concepto, la entropía de S en relación con esa clasificación binaria es:

$$Entropia(S) = -p_{\oplus} \log_2 p_{\oplus} - p_{\ominus} \log_2 p_{\ominus} \quad (3-8)$$

donde p_{\oplus} es la proporción de ejemplos positivos de S y p_{\ominus} , es la proporción de ejemplos negativos de S .

En términos más generales, si el atributo de destino puede asumir n valores diferentes, entonces la entropía de S está definida como:

$$Entropia(S) = \sum_{i=1}^n -p_i \log_2 p_i \quad (3-9)$$

donde p_i es la proporción de S que pertenece a cada clase i .

La ganancia de información de un atributo A , relativa a una colección de ejemplos S , se define como:

$$Ganancia(S, A) = Entropia(S) - \sum_{v \in Valores(A)} \frac{|S_v|}{|S|} Entropia(S_v) \quad (3-10)$$

donde $Valores(A)$ es el conjunto de todos los valores posibles para el atributo A , y S_v , es el subconjunto de S para el que el atributo A tiene valor v , es decir, $S_v = \{s \in S | A(s) = v\}$.

El árbol ID3 es capaz de tratar con atributos discretos o continuos. Sin embargo, cuando los atributos son continuos, el ID3 no clasifica correctamente los ejemplos dados. Por esta razón, se propuso el árbol J48 como una extensión del ID3.

3.3.2. J48

El árbol J48 también conocido como C4.5, es un árbol de decisión a partir de los datos mediante particiones realizadas recursivamente. El algoritmo considera todas las pruebas posibles que pueden dividir el conjunto de datos y selecciona la prueba que resulta con mayor ganancia de información. Para cada atributo discreto, se considera una prueba con n resultados, siendo n el número de valores posibles que puede tomar el atributo. Para cada atributo continuo, se realiza una prueba binaria $(1, 0)$ de los valores que toma el atributo en los datos. En cada nodo, el sistema debe decidir cuál prueba escoge para dividir los datos [23]. Las pruebas propuestas para el J48 son:

- Prueba estándar: Esta prueba es para variables discretas, con un resultado y una rama para cada valor posible de la variable.
- Prueba compleja: Basada en una variable discreta, los valores posibles son asignados a un número variable de grupos con un resultado posible para cada grupo, en lugar de para cada valor.
- Prueba binaria: Si una variable A tiene valores numéricos continuos, se realiza esta prueba binaria con resultados $A \leq Z$ y $A > Z$, donde se debe determinar el valor límite Z .

Todas estas pruebas se evalúan observando la ganancia resultante de la división de datos que producen.

Características del algoritmo J48

- Permite trabajar con valores continuos para los atributos.
- Los árboles son menos frondosos, ya que cada hoja cubre una distribución de clases, no una clase en particular.
- Utiliza el método “divide y vencerás” para generar el árbol de decisión inicial a partir de un conjunto de datos de entrenamiento.

- Se basan en la utilización del criterio de proporción de ganancia. De esta manera se consigue evitar que las variables con mayor número de categorías salgan beneficiadas en la selección.
- Es recursivo.

3.3.3. *Random Forest*

Random Forest (RF) es una combinación de muchos árboles de clasificación. Para clasificar un objeto, se procesa su vector de entrada en cada uno de los árboles del bosque. Cada árbol genera una clasificación y el bosque elige la clasificación tomando en cuenta el árbol más votado sobre todos los del bosque [14].

Cada árbol se desarrolla como sigue:

- Si el número de casos en el conjunto de entrenamiento es N , prueba N casos aleatoriamente, pero con sustitución, de los datos originales. Esto es el conjunto de entrenamiento para la creación del árbol.
- Si hay M variables de entrada, un número $m < M$ es especificado para cada nodo, m variables son seleccionadas aleatoriamente del conjunto M y la mejor participación de este m es usada para dividir el nodo. El valor de m se mantiene constante durante el crecimiento del bosque.
- Cada árbol crece de la forma más extensa posible, sin ningún tipo de poda.

Una ventaja de RF es que puede manejar cientos de variables de entrada sin eliminación de otras variables. Además es un método eficaz para estimar datos perdidos y mantiene la exactitud cuando una proporción grande de los datos falla.

3.4. Clasificación basada en SVM

Las Maquinas de Vectores de Soporte (SVM por sus siglas en inglés, *Support Vector Machines*), son sistemas de aprendizaje que utilizan funciones lineales en espacios característicos

de dimensión muy alta, ensayando algoritmos de aprendizaje de la teoría del aprendizaje estadístico [16]. Inicialmente las SVMs fueron pensadas solamente para resolver problemas de 2 clases, pero a lo largo de los años se han propuesto alternativas para extender el uso a problemas de 3 o más clases. En muchas aplicaciones, las SVM han mostrado tener gran desempeño, más que las máquinas de aprendizaje tradicional como las RNA [41].

Inicialmente una SVM mapea los puntos de entrada a un espacio de características de una dimensión mayor, es decir, si los puntos de entrada están en \mathbb{R}^2 entonces son mapeados por la SVM a \mathbb{R}^3 y encuentra un hiperplano que los separe y maximice el margen m entre las clases en este espacio. La Figura 3-7 muestra el hiperplano que maximiza el margen m entre dos clases. Maximizar el margen m es un problema de programación cuadrática (QP) y puede ser resuelto introduciendo multiplicadores de Lagrange.

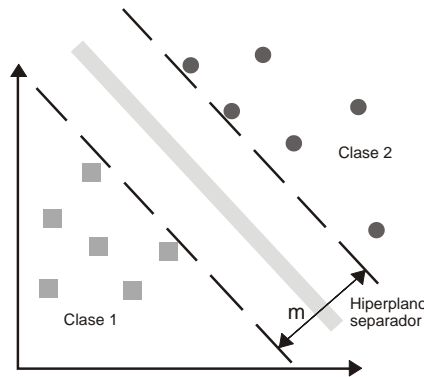


Figura 3-7: Hiperplano que separa a dos clases

La SVM encuentra el hiperplano óptimo utilizando el producto punto con funciones en el espacio de características que son llamados *Kernels*. La solución del hiperplano óptimo puede ser escrita como la combinación de unos pocos puntos de entrada que son llamados vectores de soporte.

De acuerdo a la naturaleza de los datos, la implementación de las SVM no siempre es la misma. Las SVM cuentan con dos opciones de trabajo. Una para el caso de los datos linealmente separables y otra para el caso de los datos linealmente no separables.

3.4.1. Caso linealmente separable

La SVM lineal es el modelo más sencillo, aunque tiene menos uso para resolver problemas de clasificación debido a que supone que el conjunto de datos de entrada es linealmente separable en el espacio de entrada como se muestra en la Figura 3-8.

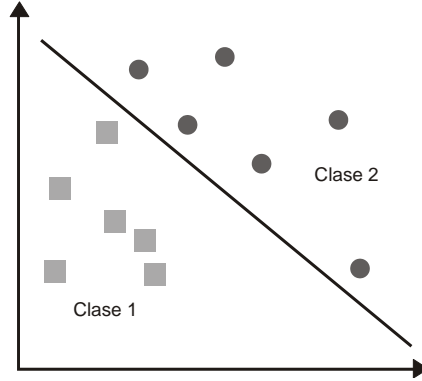


Figura 3-8: Clases linealmente separables.

Normalmente esta SVM etiqueta cada una de las clases como $+1$ y -1 . Dado un conjunto separable existe al menos un hiperplano que separa los vectores $x_i, i = 1, 2, \dots, n$. Entonces las SVM busca entre todos los hiperplanos separadores, aquel que maximice la distancia m de separación entre los conjuntos de las dos clases posibles [16].

Básicamente una SVM lineal se puede definir como sigue: Sea $z = \phi(x)$, el vector correspondiente en el espacio de características con un mapeo ϕ de \mathfrak{R} a un espacio de características Z . Se desea encontrar el hiperplano siguiente:

$$w \cdot z + b = 0 \tag{3-11}$$

donde $w \in Z$ y $b \in \mathfrak{R}$, tal que podamos separar el punto x_i de acuerdo a la función:

$$(w \cdot z_i + b) \geq +1, \quad y_i = +1 \tag{3-12}$$

$$(w \cdot z_i + b) \leq -1, \quad y_i = -1 \tag{3-13}$$

Al obtener la máxima distancia m entre los dos hiperplanos esto quedaría como:

$$\frac{2}{\|w\|^2} \tag{3-14}$$

Luego, para encontrar el máximo margen es necesario minimizar w como sigue:

$$\min_w \|w\|^2 \quad \text{sujeto a } y_i(w^T x_i + b) \geq 1, \quad i = 1, 2, \dots, n \tag{3-15}$$

La región de generalización de una SVM es el espacio dentro de los hiperplanos 3-12 y 3-13.

3.4.2. Caso linealmente no separable

En la práctica de clasificación, es habitual encontrarse conjuntos de datos linealmente no separables como se muestra en la Figura 3-9, y por tanto nunca podrían ser separados por medio de un hiperplano.

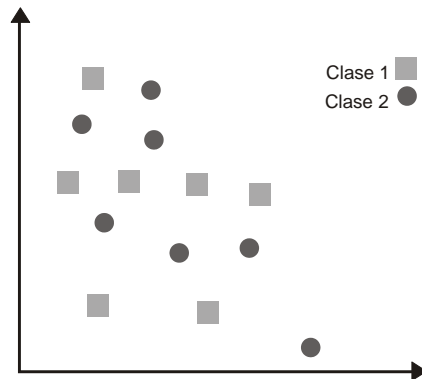


Figura 3-9: Clases linealmente no separables

Para este caso, las SVMs amplían las ideas generales de caso separable al caso no separable introduciendo una variable ξ de holgura en las restricciones, replanteando un nuevo conjunto de restricciones:

$$(w \cdot z_i + b) \geq +1, \quad y_i = +1$$

$$(w \cdot z_i + b) \leq -1, \quad y_i = -1$$

$$\xi_i \geq 0$$

Si una entrada no es ubicada en la clase correcta, es necesario que el valor correspondiente ξ_i sea superior a la unidad, es decir, si en el vector x_i se comete un error entonces $\xi_i \geq 1$ y por tanto $\sum \xi_i$ es una cuota del número de errores que se cometen dentro del conjunto de entrenamiento. Debido a que en el conjunto de datos no separables es común encontrar demasiados errores, es lógico que se pueda plantear el problema de minimizar:

$$\text{mín} \left\{ \frac{1}{2} w \cdot w + C \sum \xi_i \right\}$$

donde C es una constante y puede ser definida como un parámetro de regularización.

Cuando la dimensión del espacio de características para separar un conjunto de datos es muy grande y no se tiene ningún conocimiento de ϕ . Existe una propiedad efectiva de la SVM donde solo se necesita una función llamada *Kernel* (K), que calcula el producto punto de los puntos de entrada en el espacio de características Z , esto es:

$$Z_i Z_j = \phi(x_i) \phi(x_j) = K(x_i, x_j)$$

Esto permite realizar una separación de los datos en el espacio de características como muestra la Figura 3-10.

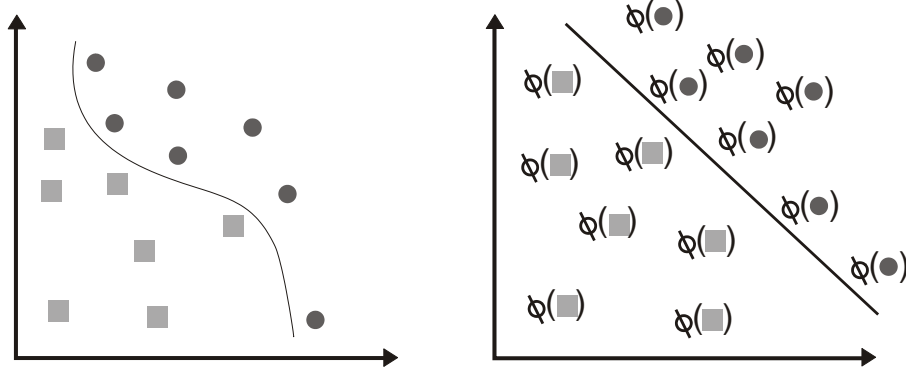


Figura 3-10: Uso de un Kernel para transformación de un espacio de datos.

En la actualidad existen diversos *Kernels* tales como el lineal, polinomial, gaussiano y base

radial que se pueden analizar con detalle en la referencia [51].

3.4.3. SMO (Sequential Minimal Optimization)

SMO (*Sequential Minimal Optimization*) es un método que resuelve rápidamente problemas generados por el entrenamiento de las SVM. Dado que el entrenamiento de una SVM requiere la solución a un gran problema de QP, SMO particiona este gran problema en una serie de problemas más pequeños QP y emplea el teorema de Osuna para garantizar una convergencia. Estos problemas más pequeños son resueltos de forma analítica, lo que reduce significativamente el tiempo del ciclo interno de procesamiento. A diferencia de los otros métodos, SMO selecciona el menor problema de optimización posible en cada paso del algoritmo utilizando 2 multiplicadores de Lagrange. La cantidad de memoria requerida para SMO es lineal en el tamaño del conjunto de entrenamiento, lo cual permite entrenar con grandes conjuntos de datos [11].

Capítulo 4

Algoritmos Genéticos

Los algoritmos genéticos (AGs), parten de la premisa de emplear la evolución natural como un procedimiento de optimización. Se caracterizan por representar las soluciones al problema que abordan en forma de cadenas binarias. Esas representaciones binarias les aportan características muy importantes de eficiencia. Sin embargo, es necesario disponer de un método para pasar esa representación binaria al espacio de búsqueda natural de cada problema.

4.1. Elementos de un algoritmo genético

Para ejecutar un AG, se requiere de una población de individuos. Cada individuo, es un candidato a ser la solución del problema tratado, o permite llegar a la solución a partir de este.

Cada individuo de la población se representa con una cadena binaria y se denomina *genotipo* del individuo que es análoga al *cromosoma* en el sistema biológico. Cada genotipo representa a puntos x del espacio de búsqueda del problema. A cada punto x se le denomina *fenotipo*. Se usa el término *gen* para referirse a la codificación de una determinada característica del individuo. Cada *gen* puede tomar distintos valores que son llamados *alelos*. Para referirse a una determinada posición de la cadena binaria se usa el término locus. La Tabla 4-1 muestra estas expresiones que se usan comúnmente en la genética y su estructura equivalente en un algoritmo genético:

Tabla 4-1: Expresiones que se utilizan en la genética con su estructura equivalente en un algoritmo genético

Evolución natural	Algoritmo genético
cromosoma	cadena
genotipo	código de cadena
fenotipo	punto sin codificar
gen	posición de cadena
alelo	valor en una posición determinada
aptitud	valor de la función objetivo

4.2. Algoritmo genético básico

En [34] se propuso un algoritmo genético básico, con el objetivo de explicar con claridad el funcionamiento de un AG. El termino básico o simple, es debido a que en cada una de sus etapas se aplican las elecciones más sencillas posibles. El algoritmo inicia con una población generada aleatoriamente. La función de adaptación, es una función matemática para la que se busca el valor óptimo en un determinado intervalo. El algoritmo entra a un ciclo donde el primer paso es una selección de individuos. Esta selección se realiza de tal manera que solo permanezcan los individuos mejor adaptados. Los individuos a cruzar se eligen de forma consecutiva, ya que se supone que el proceso de selección ha reubicado a los individuos de forma eficiente. Se aplica una mutación aleatoria y se determina el nivel de adaptación de la nueva generación de individuos. El criterio de paro, es un número máximo de generaciones en las que no hubo mejora de aptitudes.

El esquema general de un algoritmo genético básico es el siguiente:

	Entrada: <i>Conjunto de datos de entrada X.</i>
	Salida: <i>Conjunto de los mejores datos aptos para resolver el problema</i>
	1: <i>Crear población inicial</i>
	2: <i>Computar población inicial</i>
Algoritmo 1	3: WHILE <i>condición de paro no se cumple</i> Do
	4: <i>Selección de individuos para la reproducción</i>
	5: <i>Cruza de individuos</i>
	6: <i>Mutación de individuos</i>
	7: <i>Computar la nueva generación</i> END

La estructura se describe con más detalle a continuación:

4.2.1. Población inicial

Los individuos de la población inicial suelen ser cadenas de ceros y unos generados de forma completamente aleatoria. Es decir, se va generando cada gen, con una función que devuelve un cero o un uno con igual probabilidad. Es importante dotar al algoritmo genético de población con suficientemente variedad, para poder explorar todas las zonas del espacio de búsqueda.

4.2.2. Selección de individuos

La idea básica de selección, es utilizar una distribución de probabilidad de selección de una cadena, donde la probabilidad es directamente proporcional a la función de aptitud. Es decir, el proceso de selección debe favorecer la cantidad de copias de los individuos más adaptados. Las técnicas de selección usadas pueden clasificarse en tres grupos: selección proporcional, selección mediante torneo y selección de estado uniforme. Sin embargo, en este trabajo solo se analizarán algunas técnicas del grupo selección proporcional, para un estudio más a fondo sobre las demás técnicas puede consultar la referencia [34].

Dos técnicas conocidas dentro de las técnicas de selección proporcional son la ruleta y sobranste estocástico. Estas se describen a continuación.

La Ruleta

Está método ha sido el más comúnmente utilizado desde los inicios de los AGs. El algoritmo presenta el problema de que el individuo menos apto puede ser seleccionado más de una vez. Sin embargo, su popularidad se debe a su simplicidad. El algoritmo de la Ruleta es el siguiente:

- Calcular la suma de valores esperados T .
- Repetir N veces (N es el tamaño de la población)
 - Generar un número aleatorio r entre 0.0 y T
 - Ciclar a través de los individuos de la población sumando los valores esperados hasta que la suma sea mayor o igual a r .

- El individuo que haga esta suma exceda el límite es el seleccionado.

Sobrante Estocástico

El sobrante estocástico reduce los problemas de la ruleta, pero puede causar convergencia prematura al introducir una mayor precisión de selección. La idea principal es asignar determinísticamente las partes enteras de los valores esperados para cada individuo y luego usar otro esquema para la parte fraccionaria. El algoritmo es el siguiente:

- Asignar de manera determinística el conteo de valores esperados a cada individuo (valores enteros)
- Los valores restantes (sobrantes del redondeo) se usan probabilísticamente para rellenar la población.

4.2.3. Cruza

Este es un método de fusión sobre la información genética de dos individuos. Este proceso provee un mecanismo para heredar características a su descendencia donde intervienen ambos padres.

La forma más simple del operador de cruce es el cruce mono punto, que consiste en seleccionar una única posición en la cadena de ambos padres e intercambiar las partes divididas por dicha posición. La Figura 4-1 muestra un ejemplo de cruce.

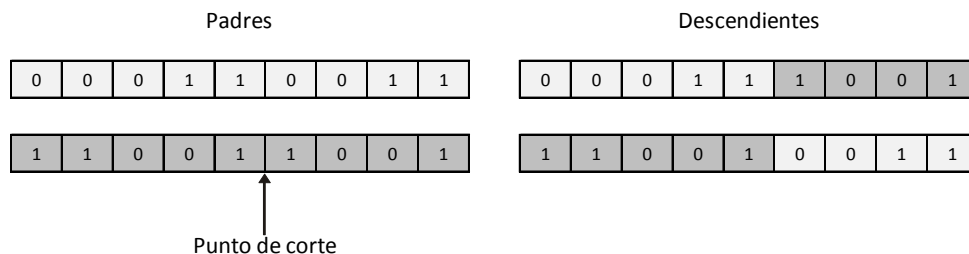


Figura 4-1: Cruza de dos cadenas binarias y sus descendientes correspondientes.

4.2.4. Mutación

La mutación es un proceso donde el material genético puede ser alterado en forma aleatoria, debidamente a un error en la reproducción o la deformación de genes. A diferencia de la genética humana, la probabilidad en un algoritmo genético es mayor. De hecho en un algoritmo genético, la mutación es una forma de evitar caer en mínimos locales.

La forma más sencilla de mutación consiste en cambiar el valor de una de las posiciones de la cadena. Si el valor es cero pasa a uno, y si es uno pasa a cero. La Figura 4-2 muestra un ejemplo:

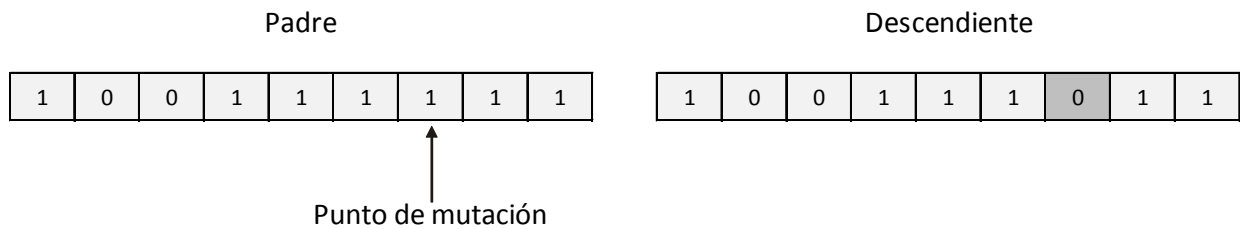


Figura 4-2: Mutación de una cadena binaria

4.2.5. Condición de paro

Es necesario especificar las condiciones en las que el algoritmo deja de evolucionar y se presenta la mejor solución encontrada. La condición de paro más sencilla, se presenta al detectar que la mayor parte de la población ha convergido a una forma similar, careciendo de la suficiente diversidad para que tenga sentido continuar con la evolución.

4.2.6. Ejemplo de un genético básico para la búsqueda de un óptimo

A continuación generamos un ejemplo para una función sencilla. Este con el objetivo de clarificar el funcionamiento de un genético básico y así adaptarlo en aplicaciones más complejas.

El genético se utilizara para llegar al valor óptimo en el intervalo $[0, 60]$ la siguiente función:

$$f(x) = \frac{x}{1+x}$$

El primer paso es determinar el tamaño m_j de la cadena binaria. El tamaño debe cubrir el intervalo $[0, 60]$ y se debe tomar en cuenta que $2^{L-1} < m_j < 2^L$. En este caso $m_j = 6$ cumple la condición.

Una vez elegida la longitud de las cadenas binarias de la población podemos generar la población inicial. Esta población se crea de manera aleatoria y para nuestro caso será únicamente de 7 individuos que son los siguientes:

No. Individuo	Cadena binaria
1	101001
2	010100
3	100010
4	011100
5	111011
6	010010
7	101101

Para cada individuo, se presenta su genotipo (cadena binaria), su fenotipo (valor real que le corresponde a la cadena binaria) y computamos su aptitud o valor de adaptación dada la función.

No. Individuo	Cadena binaria	Valor	Aptitud
1	101001	41	0,952
2	010100	20	0,952
3	100010	34	0,971
4	011100	28	0,965
5	111011	59	0,983
6	010010	18	0,947
7	101101	45	0,978

Siguiendo los pasos del algoritmo, la siguiente fase entra el ciclo del AG donde comenzamos con el proceso de selección de individuos supervivientes para reproducirse. Solo tomamos el individuo con mejor aptitud (número 5) que pasara automáticamente a la siguiente generación

y utilizamos un método de selección para los individuos que se reproducirán y generara la nueva población. Para este caso utilizamos el método de la ruleta. Así que iniciamos calculando la suma de valores esperados (Ve).

No. Individuo	Cadena binaria	Valor	Aptitud	Valor esperado (Ve)
1	101001	41	0,976	1,0089
2	010100	20	0,952	0,9841
3	100010	34	0,971	1,0037
4	011100	28	0,965	0,9975
5	111011	59	0,983	1,0161
6	010010	18	0,947	0,9789
7	101101	45	0,978	1,0109
			$\sum = 6,772$	$\sum = 7$
$\bar{f} = \frac{6,772}{7} = 0,967$			$Ve = 7$	

Entonces $T =$ suma de Ve y $r \in [0,0, T]$

Generando el primer valor aleatorio $r \in [0,0, 7,0]$

$r = 1,3$

(ind1) suma = 1,0089 < r

(ind2) suma = 1,9930 > r

Seleccionamos a ind2 que corresponde a la cadena 2.

Generamos un nuevo valor aleatorio r

$r = 2,8$

(ind1) suma = 1,0089 < r

(ind2) suma = 1,9930 < r

(ind3) suma = 2,9967 > r

Seleccionamos a ind3 que corresponde a la cadena 3

Generamos un nuevo valor aleatorio r

$r = 4,9$

$$(\text{ind1}) \text{ suma} = 1,0089 < r$$

$$(\text{ind2}) \text{ suma} = 1,9930 < r$$

$$(\text{ind3}) \text{ suma} = 2,9967 < r$$

$$(\text{ind4}) \text{ suma} = 3,9942 < r$$

$$(\text{ind5}) \text{ suma} = 5,0103 > r$$

Seleccionamos a ind5 que corresponde a la cadena 5

Generamos un nuevo valor aleatorio r

$$r = 0,6$$

$$(\text{ind1}) \text{ suma} = 1,0089 > r$$

Seleccionamos a ind1 que corresponde a la cadena 1

Generamos un nuevo valor aleatorio r

$$r = 6,3$$

$$(\text{ind1}) \text{ suma} = 1,0089 < r$$

$$(\text{ind2}) \text{ suma} = 1,9930 < r$$

$$(\text{ind3}) \text{ suma} = 2,9967 < r$$

$$(\text{ind4}) \text{ suma} = 3,9942 < r$$

$$(\text{ind5}) \text{ suma} = 5,0103 < r$$

$$(\text{ind6}) \text{ suma} = 5,9892 < r$$

$$(\text{ind7}) \text{ suma} = 7,0000 > r$$

Seleccionamos a ind7 que corresponde a la cadena 7

Generamos un nuevo valor aleatorio r

$$r = 4,7$$

$$(\text{ind1}) \text{ suma} = 1,0089 < r$$

$$(\text{ind2}) \text{ suma} = 1,9930 < r$$

$$(\text{ind3}) \text{ suma} = 2,9967 < r$$

$$(\text{ind4}) \text{ suma} = 3,9942 < r$$

$$(\text{ind5}) \text{ suma} = 5,0103 > r$$

Seleccionamos a ind5 que corresponde a la cadena 5

Tras el proceso de selección, se obtienen los siguientes individuos:

No. Individuo	Cadena binaria
2	010100
3	100010
5	111011
1	101001
7	101101
5	111011

Se puede observar que el individuo más apto (numero 5) recibió más copias en la nueva población, mientras que el individuo de baja adaptación (numero 6) desapareció. Sin embargo, el método de la ruleta, es un proceso probabilístico, por lo que también los individuos de baja adaptación tienen oportunidades.

El siguiente paso de la evolución es la reproducción de nuevos individuos mediante el operador cruce monopunto. Dado que son 6 individuos, se realizan las 3 cruza siguientes:

Cruza 1

Padres	Descendientes
2 010100 →	010010
3 100010 →	100100

Cruza 2

Padres	Descendientes
5 111011 →	111001
1 101001 →	101001

Cruza 3

Padres	Descendientes
7 101101 →	101011
5 111011 →	111101

Obtenemos la nueva generación:

No. Individuo	Cadena binaria
1	010010
2	100100
3	111001
4	101001
5	101011
6	111101

En el proceso de mutación seleccionamos un individuo aleatoriamente.

Padre	Descendiente
6 111101	→ 101101

Computamos la nueva generación

No. Individuo	Cadena binaria	Valor	Aptitud
1	111011	59	0,983
2	010010	18	0,947
3	100100	36	0,972
4	111001	57	0,982
5	101001	41	0,976
6	101011	43	0,977
7	101101	45	0,978

Se puede observar como a medida que avanzan las generaciones existen valores mejores adaptados. Sin embargo la mejora de adaptación de una generación a otra no está garantizada. Por lo tanto, el algoritmo continúa hasta que existan varias generaciones sin mejora.

Capítulo 5

Metodología

En este Capítulo se describe la metodología y estrategias con que se llevó a cabo el análisis comparativo de las características que influyen en una buena clasificación sobre los distintos conjuntos de datos. El trabajo comenzó con la búsqueda de bibliografía relacionada con el procesamiento de imágenes. El análisis se encaminó a los tipos de segmentación y métodos de extracción de características, donde se incluyeron características texturales, cromáticas y geométricas. Además se analizaron algunos de los clasificadores utilizados en la investigación.

El pre procesamiento de imágenes es una etapa muy importante en PDI. Sin embargo, no fue necesario implementarla para la gran mayoría del conjunto de datos que se utilizó, ya que las imágenes obtenidas fueron tomadas en un entorno controlado (Fondo Blanco). Así que comenzamos a tratar las imágenes desde la etapa de segmentación, posteriormente se trabajó con la extracción de características y por ultimo con la clasificación. El conjunto de datos de hojas de planta que se utilizó consta de 220 familias.

Las etapas de este proyecto se muestran en el diagrama de la Figura 5-1. Este diagrama muestra un esquema general de las etapas que se llevaron a cabo durante el proyecto. Cada etapa se analiza de forma independiente, permitiendo comprender a detalle su implementación.

5.1. Creación de los conjuntos de datos

De las 220 familias de hojas de plantas, se formaron dos conjuntos de hojas, un conjunto de hojas llamado conjunto trivial y otro conjunto llamado conjunto complejo. El conjunto de

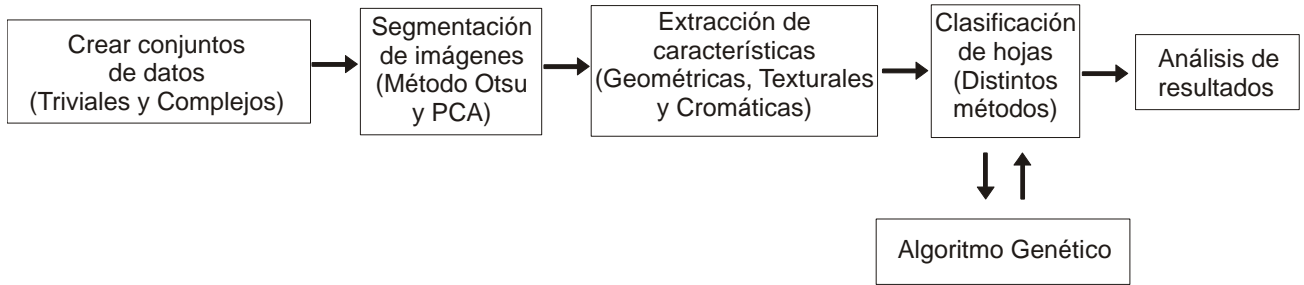


Figura 5-1: Etapas para la metodología propuesta.

hojas complejo se refiere al conjunto de hojas que tienen gran similitud entre sí. El conjunto de hojas trivial, se refiere al conjunto de hojas que son muy distintas entre sí. La Figura 5-2 muestra un ejemplo de tres hojas consideradas triviales y tres hojas consideradas complejas. Se denota así, al término trivial y complejo dado a la facilidad o complejidad que pudiera tener un clasificador para reconocer la imagen. Es decir, el conjunto de hojas trivial que contiene hojas muy distintas entre sí, se le puede atribuir que dada la gran diferencia en características geométricas, el clasificador pueda distinguir entre una y otra hoja muy fácilmente. Por otro lado, en el conjunto de hojas complejo que tiene hojas muy similares entre sí, podría ser un reto entre varios clasificadores distinguir una hoja con otras dado a sus características muy similares. Sin embargo, es cierto que cada clasificador utiliza técnicas distintas que pueden tener variaciones en los resultados finales. Además, al mezclar características de la hoja y técnicas de segmentación, se puede realizar un análisis mejor detallado en cada clasificador.

El proceso de separación de conjuntos, se realizó de forma manual de acuerdo a las semejanzas y diferencias notables a simple vista. Se tomó como muestra una imagen de hoja de cada clase y se colocaron en una carpeta donde se pudieran apreciar mejor esas semejanzas y diferencias. Para el conjunto de hojas trivial se realizó un filtrado de imágenes de hoja que tenían diferencias muy marcadas, es decir, si alguna hoja tenía cierta similitud con otra, se eliminaba del conjunto. Finalmente, el total de familias asociadas al conjunto trivial fue de 90 familias de las 220 con las que se contaba. La Figura 5-3 muestra un ejemplo de hoja de cada familia asociada al conjunto trivial.

Para el conjunto de hojas complejas, se tomaron en cuenta varias razones de similitud. Este proceso no fue una tarea fácil dado al análisis meticuloso de escoger el parecido de cada hoja

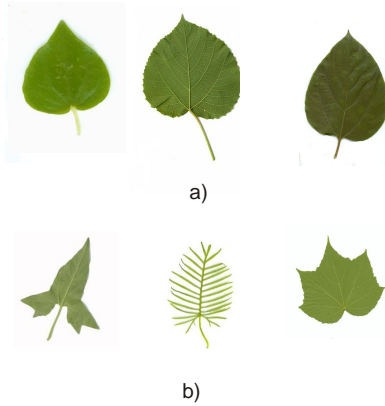


Figura 5-2: Ejemplo de imágenes de hojas complejas y triviales a) Imágenes denominadas complejas (muy similares entre sí). b) Imágenes denominadas triviales (muy distintas entre sí).

con el resto. Inicialmente se tomaron en cuenta veinte razones de similitud. Sin embargo, con el objetivo de minimizar estas, se eliminaron las que tenían muy poca diferencia entre una otra, quedando solamente once subconjuntos de familias de hojas complejas. Las Figuras 5-4 y 5-5 muestran cada subconjunto creado con algunos ejemplos de familias de hojas que se tomaron para formar cada subconjunto, además se describe la razón de similitud que se tomó en cuenta para asignar cada hoja al grupo y se muestran tres hojas como ejemplo. El total de familias de hojas asociadas en cada subconjunto se describe en el encabezado "Familias Asociadas". Se puede notar en las Figuras que el tamaño de cada subconjunto es distinto y varía mucho de un subconjunto a otro. El subconjunto más pequeño se formó con tan solo tres familias y el subconjunto con más familias se formó con treinta y siete.

Las Figuras 5-6, 5-7, 5-8, 5-9, 5-10, 5-11, 5-12, 5-13, 5-14, 5-15 y 5-16 muestran un ejemplo de una imagen de hoja asociada a cada subconjunto complejo. Las Figuras se muestran en el mismo orden que se muestran los subconjuntos de las Figuras 5-4 y 5-5. Se puede notar que algunas familias de hoja pueden pertenecer a 1 o más subconjuntos debido a que su razón de similitud cubre ambas clases.



Figura 5-3: Ejemplo de imágenes asociadas al conjunto de hojas trivial.







Subconjunto	Familias Asociadas	Ejemplos	Razon de Similitud
1	19		Orbicular
2	8		Lineal
3	14		Lanceolada
4	27		Elíptica
5	12		Aovada
6	3		Lacerada con forma de pentágono

Figura 5-4: Razón de similitud de subconjuntos complejos 1-6






Subconjunto	Familias Asociadas	Ejemplos	Razon de Similitud
7	7		Lineal dentada
8	9		Espatulada
9	13		Aovada con cuspide
10	20		Elongada
11	37		Obovada

Figura 5-5: Razón de similitud de subconjuntos complejos 7-11

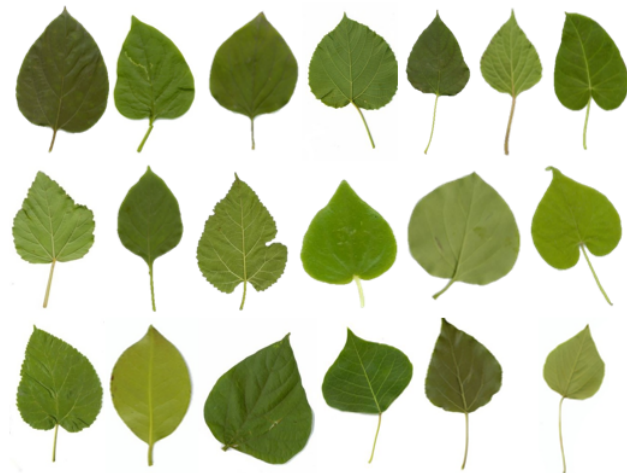


Figura 5-6: Subconjunto complejo 1



Figura 5-7: Subconjunto complejo 2



Figura 5-8: Subconjunto complejo 3



Figura 5-9: Subconjunto complejo 4



Figura 5-10: Subconjunto complejo 5



Figura 5-11: Subconjunto complejo 6



Figura 5-12: Subconjunto complejo 7



Figura 5-13: Subconjunto complejo 8



Figura 5-14: Subconjunto complejo 9



Figura 5-15: Subconjunto complejo 10



Figura 5-16: Subconjunto complejo 11

5.2. Segmentación de imágenes

La etapa de segmentación es muy importante, ya que ayuda a detectar y visualizar fácilmente bordes y forma de la hoja. Las técnicas de segmentación empleadas en los conjuntos de datos fueron: segmentación basada en PCA y segmentación basada en el método de Otsu. La razón por la que se eligieron estos métodos fue por la efectividad que cada uno de estos genera. Cuando se trata de segmentar hojas con geometría fácil, cada tipo de segmentación devuelve resultados muy similares. Sin embargo al segmentar imágenes de hojas con geometría compleja cada tipo de segmentación puede tener pequeñas diferencias para cada hoja. En la Figura 5-17, se puede ver la similitud de resultados al segmentar imágenes de hoja con geometría trivial. La Figura 5-18 muestra los resultados de segmentación de imágenes de hojas con geometría compleja. Un ejemplo claro de diferencias de segmentación, se puede notar en la Figura 5-18 parte b) donde la segmentación realizada con el método Otsu omitió el tallo de la hoja, mientras que la segmentación basada en PCA si incluyó el tallo.

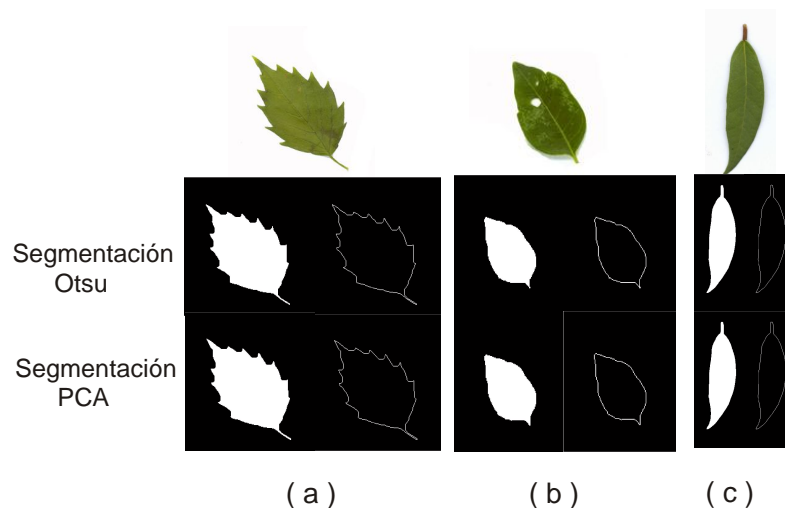


Figura 5-17: Imágenes de hoja de planta fácil de segmentar.

Un aspecto importante en la etapa de segmentación es que antes de implementarla, se aplicó un algoritmo que rellena aquellos orificios que se encuentran dentro de la imagen. Un ejemplo de ello se muestra en la Figura 5-17 parte b), donde se puede notar que el orificio que se encuentra en la imagen original es rellenado en las dos segmentaciones realizadas. Este rellenado de orificios

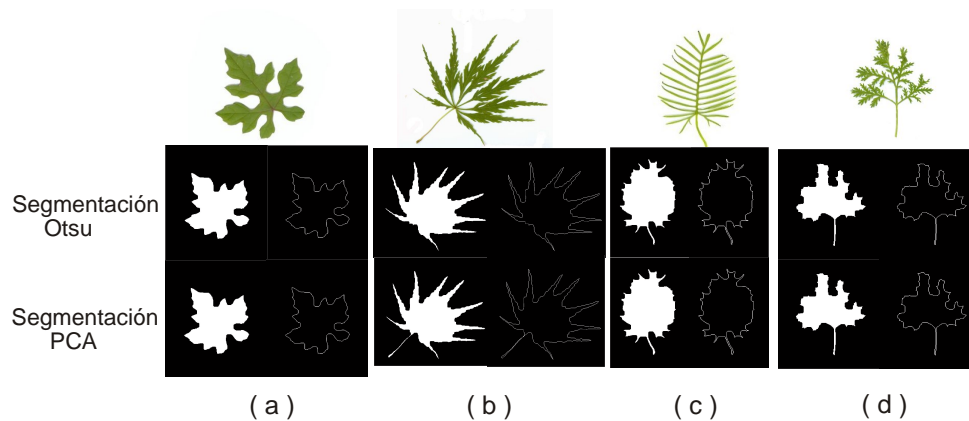


Figura 5-18: Imágenes de hoja de planta difícil de segmentar.

se implementó con el fin de que estos no influyeran en la extracción de características de la hoja, ya que algunas técnicas toman en cuenta los orificios de la imagen.

La etapa de segmentación fue una tarea sencilla ya que el tipo de imágenes de hojas utilizadas tenían un entorno controlado y no se encontraban hojas solapadas. Sin embargo, en muchos casos las imágenes de hojas de planta son obtenidas del medio ambiente lo que hace que interfiera mucho ruido en la imagen. Para esos casos, es necesario aplicar la etapa de pre procesamiento a cada imagen antes de segmentarla. De las distintas etapas de pre procesamiento existentes, se aplican solo las que más se adapten al tipo de imagen a tratar. Los factores que más afectan la segmentación pueden ser: una imagen borrosa o una imagen con mucho ruido de fondo. Específicamente en las hojas de planta, pueden existir imágenes con hojas solapadas que pudieran afectar la identificación de la hoja de interés. La segmentación de hojas con un fondo complejo en un entorno natural está fuera de alcance en este proyecto, pero se puede analizar con detalle en [52].

5.3. Extracción de características

Una vez terminada la etapa de segmentación, se extrajeron características texturales, cromáticas y geométricas. En las características texturales se utilizaron los 28 descriptores de Haralick. Se utilizaron 23 tipos de características geométricas, que forman un total de 54 características. En cuanto a las características cromáticas se utilizaron los momentos de Hu con intensidad

para rojo, verde y azul, momentos de Fourier, descriptores Gabor y HOG piramidal donde finalmente se obtuvieron 861 características cromáticas. Las Tablas 5-1, 5-2 y 5-3 muestran los tipos de características que extrajeron en cada conjunto de hojas junto con el número de características asociadas. La Tabla 5-1 muestra las características geométricas, la Tabla 5-2 muestra las características texturales y la Tabla 5-3 muestra las características cromáticas.

Tabla 5-1: Tipo de características geométricas y número de características asociadas

Características Geométricas	
Tipo de características	Número de características
Descriptores de Fourier	8
Momentos de Hu	7
Momentos	8
Características Gupta	3
Centro de gravedad	2
Altura	1
Anchura	1
Area	1
Perimetro	1
Redondez	1
Factor Danielsson	1
Numero Euler	1
Diametro Equivalente	1
Longitud de eje mayor	1
Longitud de eje menor	1
Orientacion	1
Solidez	1
Extension	1
Excentricidad	1
Area convexa	1
Area Rellenada	1
Momentos de Flusser	3
Elipse	7
Características Totales 54	

Tabla 5-2: Tipo de características texturales y número de características asociadas

Características Texturales	
Tipo de características	Número de características
Descriptores Haralick	28
Características Totales 28	

Tabla 5-3: Tipo de características cromáticas y número de características asociadas

Características Cromáticas	
Tipo de características	Numero de caraterísitcas
Momentos de Hu Rojo	7
Momentos de Hu Verde	7
Momentos de Hu Azul	7
Hog Piramidal	765
Gabor	67
Momentos de Fourier	8
Características Totales 861	

Con el objetivo de identificar el impacto de cada característica y cada técnica de segmentación en los distintos clasificadores, en cada familia se obtuvieron resultados de extracción de características mezcladas con cada método de segmentación, es decir, de las imágenes segmentadas con la técnica basada en PCA se obtuvieron las características cromáticas y se almacenaron los datos en un archivo de texto, después, en otro archivo distinto con las mismas imágenes segmentadas con la técnica basada en PCA se obtuvieron las características texturales, y posteriormente se realizó lo mismo con las características geométricas y con la combinación de estas. Para las imágenes segmentadas con el método Otsu se realizó el mismo método. En la Tabla 5-4 se muestran las distintas combinaciones de técnicas de segmentación y características extraídas que se realizó en los conjuntos de hojas.

Tabla 5-4: Combinaciones de las técnicas de segmentación con los tipos de características extraídas en cada conjunto de hojas

Técnica de Segmentación	Características
PCA	Cromáticas
PCA	Texturales
PCA	Geométricas
PCA	Cromáticas y Texturales
PCA	Cromáticas y Geométricas
PCA	Geométricas y Texturales
PCA	Cromáticas, Geométricas y Texturales
Otsu	Cromáticas
Otsu	Texturales
Otsu	Geométricas
Otsu	Cromáticas y Texturales
Otsu	Cromáticas y Geométricas
Otsu	Geométricas y Texturales
Otsu	Cromáticas, Geométricas y Texturales

Las etapas de segmentación y extracción de características se realizaron con la ayuda del programa Matlab. Se generó un programa que realizó la tarea de segmentación y extracción de características según la combinación que se requería de la Tabla 5-4. El programa almacenaba los resultados en archivos de texto con nombres que ayudaban a identificar la técnica de segmentación utilizada y el tipo de características que se extraían. Por ejemplo, para el archivo que contiene características cromáticas y texturales con una segmentación PCA, lo nombro 'PCA_CTX'. La 'X' al final denotaba que las características geométricas estaban ausentes en ese archivo.

Inicialmente, por cada una de las 220 familias se generaron 14 archivos que representaban cada una de las combinaciones. La estructura de cada archivo se formó con una matriz en la que cada fila representa una hoja en particular y cada columna los resultados de una característica extraída. Es decir, de una carpeta que contenía una familia con 20 imágenes de hojas como muestra y a la que se le extrajeron solo las características texturales, formó un archivo con 20 filas que representan a cada imagen de la hoja y 29 columnas. Las primeras 28 columnas representan los 28 descriptores Haralick y la última representaba el número asignado a la familia de hoja.

Una vez obtenidos los resultados de cada familia, se procedió a juntar todos los archivos generados según el conjunto al que pertenecía cada familia (Trivial o Complejo). Es decir, para el conjunto de familias triviales se creó un archivo que contenía todos los resultados generados de las familias que pertenecen a dicho conjunto. En caso de pertenecer al conjunto de hojas complejas se generaron archivos por cada subgrupo. Cada archivo se nombró según el conjunto al que pertenece. Por ejemplo, todos los archivos que contenían los resultados de una segmentación PCA y que contenían características cromáticas y texturales nombrados como 'PCA_CTX' que pertenecían al conjunto de datos triviales, se juntaron en un nuevo archivo nombrado 'PCA_CTX_T'. Para aquellos resultados que pertenecían al grupo 2 del conjunto de datos complejo, se juntaron en un nuevo archivo nombrado 'PCA_CTX_CG2'. Finalmente se generaron 168 archivos (14 archivos por cada uno de los 11 subconjuntos complejos y un conjunto trivial) a los que se le aplicó la etapa de clasificación y el algoritmo genético.

5.4. Clasificación de resultados

Para la etapa de clasificación, se utilizó el programa Weka versión 3.7.5 que tiene integrados los clasificadores que se utilizaron en este proyecto. Los clasificadores utilizados fueron los siguientes:

- Bayes Net
- Naive Bayes
- Perceptron Multicapa
- SMO
- J48
- Random Forest

El programa Weka maneja ficheros con un formato denominado arff (Acronimo de *Attribute-Relation File Format*). Sin embargo, los archivos generados al obtener las características se guardaron con extensión mat. Por lo tanto, se realizó una conversión del formato mat al formato arff para que los datos fueran aceptados por Weka y así se pudiera realizar la etapa de clasificación.

La estructura de los ficheros arff se compone de tres partes: cabecera, declaración de atributos y sección de datos:

Cabecera: En esta sección se define el nombre de la relación con un formato como el siguiente:

@relation <nombre-de-la-relacion>

Declaración de atributos: Aquí se incluye una línea por cada atributo o columna con que se compondrá nuestro conjunto de datos. La sintaxis para este apartado es como sigue:

@attribute <nombre-del-atributo> <tipo>

donde *nombre-del-atributo* es de tipo String. Los tipos varían dependiendo del tipo de atributo y pueden ser numeric, integer, date, String o enumerar entre llaves los posibles valores que pueda tomar el atributo.

Sección de datos: Esta sección comienza con una sintaxis @data y continúa con todos los datos que componen a la relación separando los atributos con comas y con saltos de línea cada relación.

Un ejemplo de la conversión resultante de archivos arff se puede apreciar en la Figura 5-19, donde se muestra el nombre del encabezado como ‘OTSUTexturalesCG6’ que hace referencia a los resultados obtenidos del subconjunto complejo 6 con una segmentación Otsu y características texturales. Continúa con la lista de características que se incluyen en el conjunto de datos. En este caso las características corresponden a los descriptores Haralick y todos los datos son de tipo real. Por último, En la sección de datos se muestran tres vectores de características como ejemplo.

```

@relation OTSUTexturalesCG6

@attribute Haralick1Tx1 real
@attribute Haralick1Tx2 real
@attribute Haralick1Tx3 real
@attribute Haralick1Tx4 real
@attribute Haralick1Tx5 real
@attribute Haralick1Tx6 real
@attribute Haralick1Tx7 real
@attribute Haralick1Tx8 real
@attribute Haralick1Tx9 real
@attribute Haralick1Tx10 real
@attribute Haralick1Tx11 real
@attribute Haralick1Tx12 real
@attribute Haralick1Tx13 real
@attribute Haralick1Tx14 real

@attribute Clase {27,44,88}

@data
0.2589,0.4197,630.5853,0.4875,0.8333,9.5694,8.1032,1.5
007,1.8941,0.0597,0.7342,-0.2925,0.4751,0.4807,27
0.2607,0.3995,580.6571,0.4095,0.8441,9.6614,8.1548,1.5
093,1.7872,0.0637,0.7081,-0.2802,0.4392,0.5714,27
0.2528,0.4222,559.3563,0.5022,0.8410,9.2189,7.8252,1.4
127,1.7768,0.0625,0.7224,-0.2546,0.4027,0.4036,27
-
-
-
-

```

Figura 5-19: Ejemplo de resultados de clasificación con estructura de un fichero arff correspondiente a los resultados del subconjunto complejo 6 con segmentación OTSU.

Una vez adaptados cada uno de los archivos a formato arff, se importaron en el programa Weka para realizar las pruebas de clasificación. En cada uno de los 168 archivos de características

se implementaron los 6 clasificadores. Por cada ejecución de clasificación se extrajeron tres datos relevantes que fueron: la precisión de instancias clasificadas correctamente, F- Measure, Area ROC y su matriz de confusión.

5.5. Algoritmo genético para reducción de características

Los resultados obtenidos en la etapa de clasificación, se extrajeron utilizando todo el conjunto de características (geométricas, texturales o cromáticas). Sin embargo, es bien sabido que muchas de esas características no benefician e incluso añaden ruido colocando valores que confunden al clasificador. Es decir, se puede llegar a una misma precisión de clasificación correcta, utilizando solo algunas de todo el conjunto de características, e incluso se puede aumentar la precisión eliminando aquellas que solo aportan ruido como descriptores.

El uso adecuado de características para una buena clasificación ha sido tratado por varios autores [12] [26] y lo llaman, el curso de la dimensionalidad. El curso de la dimensionalidad analiza las características de alta dimensión y nos dice que del número de características de entrada hay un número óptimo de características que se pueden seleccionar en relación con el tamaño de la muestra para maximizar el rendimiento de un clasificador, ya que una dimensión grande de características solo aporta ruido y confunde a los métodos de aprendizaje. La Figura 5-20 muestra el ejemplo del curso de dimensionalidad y se ve claramente que al aumentar las características, aumenta la precisión de clasificación, sin embargo, solo hasta cierto punto, ya que después la empeora con mayor impacto. Así que el desafío de reducir la dimensionalidad de características es fundamental en la etapa de clasificación.

Un factor importante al momento de reducir características, es eliminar aquellas que no son clave importante en el clasificador. Aunque, el hecho de decir que cierto descriptor aporta ruido en un clasificador, no quiere decir que no funcione o que sea inútil en otras clasificaciones, más bien significa que tal vez es un rasgo o características que comparte el mismo valor con otras hojas de planta. Si dicho descriptor se utilizará para identificar otros objetos, tal vez pueda ser un buen elemento y tener buena efectividad en su identificación.

Con el objetivo de identificar las características menos eficientes, se implementó un algoritmo genético básico sobre las características de cada uno de los conjuntos de datos. La forma en que

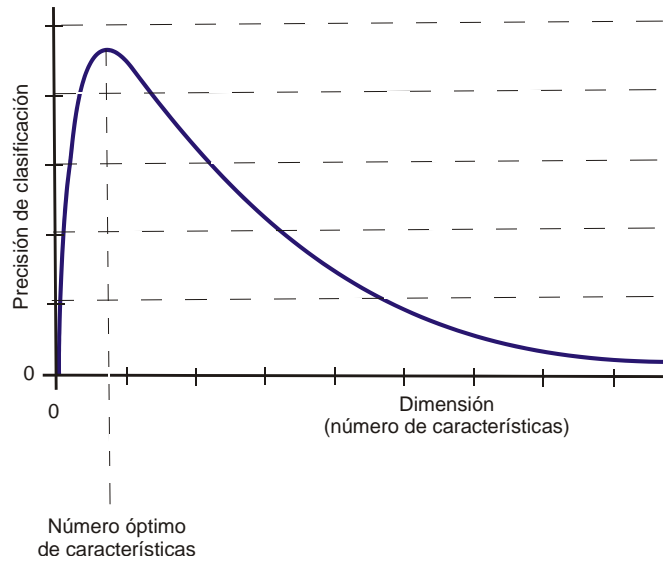


Figura 5-20: Curso de la dimensionalidad.

se adaptó el genético (AG) se describe a continuación:

5.5.1. Algoritmo Genético propuesto

Cada conjunto de características por hoja, forma un vector de acuerdo al número de descriptores extraídos. Por ejemplo, si las características extraídas son geométricas, se forma un vector de 54 datos (ver Tabla 5-1). Si las características son texturales se forma un vector de 28 datos (ver Tabla 5-2). En cada archivo arff generado, se encuentran los vectores de características de todas las hojas que pertenecen a la misma familia y que además corresponden a la mismo tipo de segmentación y tipo de características extraídas, es decir, en el archivo 'PCA_CTX_CG2' se encuentran los vectores característicos de todas las familias de hoja que pertenecen al subconjunto complejo 2 y que además su segmentación fue basada en PCA con las características cromáticas y texturales.

De acuerdo al número de características de cada conjunto de datos, se formó el tamaño de cada cadena binaria que se necesitó para implementar el algoritmo genético. Tomando como ejemplo el conjunto de vectores formados por características texturales de un archivo, este forma una cadena binaria de 28 elementos. La relación que existe entre cada cadena binaria con el conjunto de características, es que el 1 se toma como característica empleada y el 0 como

ausencia de esa característica. En la Figura 5-21 se muestra un ejemplo de adaptación de la cadena binaria con un conjunto de características

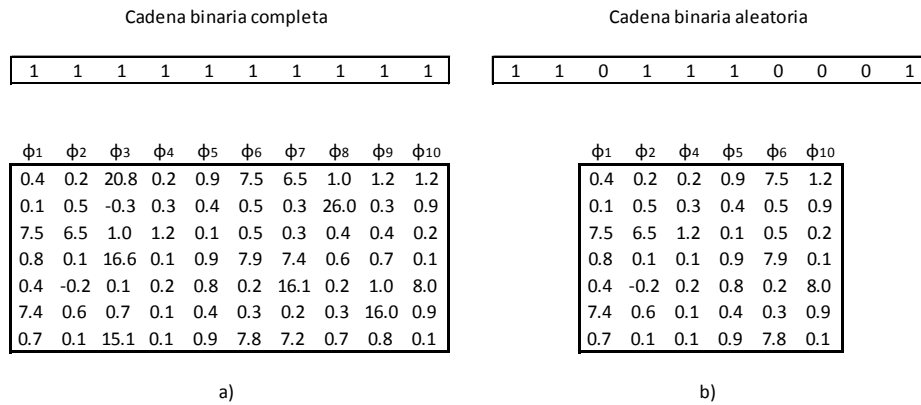


Figura 5-21: Adaptación de un conjunto de características con una cadena binaria de un AG.

La aptitud de cada individuo se toma de la precisión que se obtiene al clasificar el conjunto correspondiente a dicha cadena. La forma en cómo funciona el AG es similar al ejemplo descrito anteriormente en el Capítulo 4.

Como primer paso se crea aleatoriamente una población de 20 individuos. Entonces, del archivo tratado que contiene todas las características se eliminan todas aquellas donde la cadena binaria contiene un 0 y se mantienen todas aquellas donde la cadena binaria contiene un 1. Obteniendo un total de 20 filas distintas. El proceso de adaptación se realiza introduciendo cada archivo en el API de Weka, donde se obtienen distintas precisiones de clasificación correcta. La Figura 5-22 muestra un conjunto de cadenas que obtienen distintas precisiones según la cadena binaria.

De toda la población, se toma el individuo con mejor resultado de clasificación y pasa intacto a la siguiente generación. Se continúa con un método de selección y se realiza el método de cruce con los individuos seleccionados para la nueva generación. Se realiza el mismo procedimiento para la siguiente generación, y si ahora se obtiene un mejor resultado que el anterior, esté reemplaza al individuo que tenía la mejor precisión. Si no hubo cadena que mejore la aptitud, la cadena anterior vuelve a permanecer intacta en la nueva generación. En la Figura 5-23 se muestran 2 iteraciones como ejemplo del AG.

El algoritmo finaliza cuando después de 10 iteraciones no hubo mejora. La cadena con

[1, 0, 1, 1, 1, 0, 0, 1, 0, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 1, 1, 1, 0] 87.4213836
 [0, 0, 0, 0, 1, 0, 1, 0, 1, 1, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 1, 1, 1, 0, 0, 1, 0, 1] 86.7924528
 [1, 0, 0, 1, 0, 0, 1, 0, 1, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 0, 0] 91.1949686
 [1, 0, 1, 0, 1, 1, 0, 1, 1, 1, 0, 0, 0, 1, 0, 1, 0, 1, 1, 1, 1, 1, 1, 1, 0, 1, 0, 0] 86.163522
 [1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 1, 1, 1, 0, 1, 1, 1] 90.5660377
 [1, 1, 1, 0, 0, 1, 0, 1, 1, 0, 1, 1, 0, 0, 0, 0, 0, 0, 1, 0, 1, 1, 0, 0, 0, 1, 1, 1, 0] 89.3081761
 [0, 1, 1, 0, 0, 1, 1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 1, 1, 0, 0, 0] 88.0503145
[0, 0, 0, 1, 0, 0, 0, 1, 1, 1, 0, 0, 0, 1, 0, 1, 0, 1, 0, 0, 1, 0, 1, 0, 1, 0, 1, 1] 93.7106918
 [0, 1, 0, 0, 0, 1, 1, 0, 0, 0, 0, 1, 1, 1, 1, 0, 0, 0, 0, 1, 1, 0, 1, 0, 1, 0, 1, 1] 87.4213836
 [1, 1, 1, 1, 0, 0, 0, 1, 0, 0, 0, 1, 0, 1, 0, 0, 0, 1, 0, 0, 1, 1, 1, 0, 0, 1, 0, 0] 91.8238994

Figura 5-22: Conjunto de cadenas binarias y las precisiones obtenidas de un clasificador

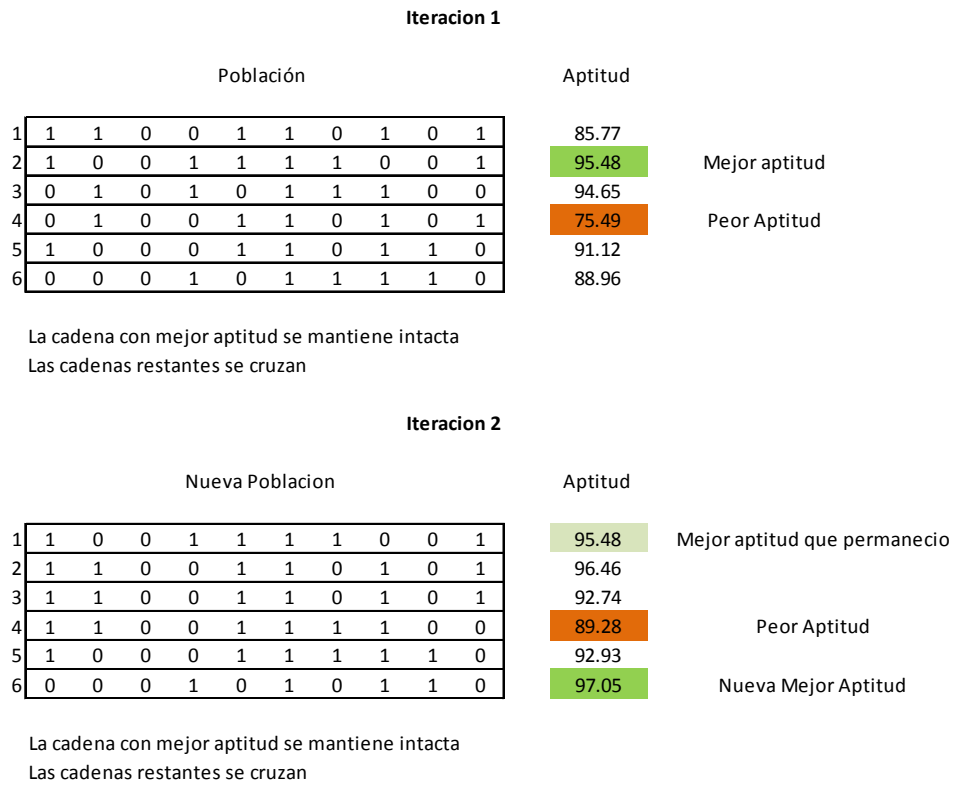


Figura 5-23: Ejemplo de dos iteraciones en el AG propuesto para reducción de características.

mejor precisión con la que termina el algoritmo se almacena en un archivo de texto junto con la precisión obtenida.

Capítulo 6

Resultados experimentales

En este Capítulo se presentan los resultados obtenidos de cada uno de los conjuntos de datos creados. Se muestran las precisiones de clasificación obtenidas de cada clasificador y se discuten las características que influyeron en la correcta o incorrecta clasificación. Se analizan los resultados haciendo una comparativa entre cada uno de ellos. Primero con los datos obtenidos al utilizar todas las características y después con los datos obtenidos al implementar el AG. En base a los resultados obtenidos por el AG, se muestran las características que sobresalieron e influyeron en una buena precisión de clasificación y las características que se despreciaron en la mayoría de datos.

6.1. Técnicas de validación

Para poder predecir cuál será el índice de error de un clasificador al momento que este se aplique a datos sin clasificar, existen distintas medidas de desempeño. Estas medidas, son reconocidas como un elemento importante en toda gestión de calidad. Para validar nuestros resultados utilizamos varias medidas de desempeño. A continuación se mencionan las que fueron utilizadas en este proyecto.

6.1.1. Cross-validation

La validación cruzada (*Cross-validation*), es un método para evaluar y comparar los algoritmos de aprendizaje mediante la división de datos en dos segmentos. El primero se utiliza para

entrenar el modelo y el segundo para validar el modelo.

El algoritmo primero divide los datos en k partes iguales. Después realiza k iteraciones de entrenamiento, tomando en cada iteración como conjunto de prueba un subconjunto diferente y construyendo el modelo con los subconjuntos restantes. La Figura 6-1 muestra un ejemplo con 4 iteraciones. El índice de error estimado es la media de todos los errores obtenidos en cada entrenamiento.

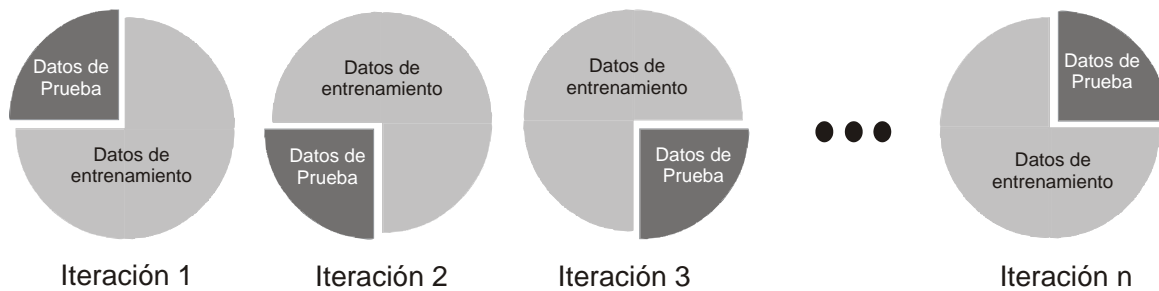


Figura 6-1: Validación cruzada de k iteraciones

La ventaja de evaluar a partir de k combinaciones de datos de entrenamiento y prueba hace que el método sea más preciso. Sin embargo, en una evaluación con un valor alto en k el proceso es lento al momento de computar. La elección del número de iteraciones depende de la medida del conjunto de datos, aunque lo más común es utilizar la validación cruzada de 10 iteraciones.

6.1.2. F-Measure

F-Measure no es más que la media armónica entre precisión y exhaustividad. La precisión representa el nivel de confianza del clasificado ya que es el porcentaje de datos clasificados correctamente. La exhaustividad representa la cobertura del clasificador, es decir, la cantidad de datos que clasifica frente a los no clasificados y clasificados. Cuando un sistema clasifica todos los datos en una sola categoría, este puede tener una exhaustividad alta, sin embargo, si la clasificación es incorrecta la precisión será baja. Tanto la precisión como exhaustividad están basadas en la matriz de confusión que está formada por cuatro casos:

Verdaderos positivos (TP): Es el caso de los datos positivos que han sido clasificados como positivos, es decir, es el total de datos que han sido clasificados correctamente.

Verdaderos negativos (TN): Es el caso de los casos negativos que ha sido clasificados como negativos, es decir, representan el número de datos clasificados en otra categoría correctamente.

Falsos Positivos (FP): Es el caso de los datos negativos que ha sido clasificados como positivos.

Falsos Negativos (FN): Es el caso de los datos positivos que han sido clasificados como negativos.

Partiendo de estos casos se puede formar la matriz de confusión como se muestra en la Tabla 6-1

Tabla 6-1: Estructura de la Matriz de Confusión		
	Positivos	Negativos
Positivos	TP: Verdaderos Positivos	FN: Falsos Negativos
Negativos	FP: Falsos Positivos	TN: Verdaderos Negativos

La técnica de matrices de confusión, no solo permite conocer el error del modelo predictivo, sino que también muestra el tipo de predicciones correctas e incorrectas cuando se aplica el modelo sobre el conjunto de prueba. Las predicciones correctas estas representadas por la diagonal principal, sin en cambio los elementos ubicados fuera de la diagonal principal, indican los errores de asignación.

Dada la matriz de confusión se pueden obtener la precisión y exhaustividad con las siguientes ecuaciones:

$$precision = \frac{TP}{TP + FP} \tag{6-1}$$

$$exhaustividad = \frac{TP}{TP + FN} \tag{6-2}$$

Para calcular F-Measure de una clase j con otra clase i primero se define la ecuación siguiente:

$$F_{ij} = \frac{2 * precision(i, j) * exhaustividad(i, j)}{precision(i, j) + exhaustividad(i, j)} \tag{6-3}$$

entonces F-Measure de un conjunto dado es calculado como sigue:

$$F - Measure = \sum \frac{n_i}{n} \text{máx}(F_{ij}) \quad (6-4)$$

donde n es el número de todo el conjunto de datos y n_i es el número de datos de la clase i .

El rango de los valores calculados esta entre 0 y 1. Un valor F-measure alto indica una mayor calidad de clasificación.

6.1.3. Área ROC

Cuando los errores llevan asociada una perdida que puede cuantificarse, es posible aplicar otra técnica de validación como el análisis de la curva ROC (por sus siglas en inglés, *Receiver Operating Characteristics*). Los gráficos ROC son útiles para visualizar el desempeño de los clasificadores y se utilizan comúnmente en la toma de decisiones médicas, aunque en los últimos años se han utilizado cada vez más en el aprendizaje automático [24]. El método consiste en un gráfico que ayuda a visualizar la disyuntiva entre la tasa de verdaderos positivos y la tasa de falsos positivos de un clasificador. La tasa de verdaderos positivos se representa en el eje las y , y la tasa de falsos positivos se representa en el eje de las x . La Figura 6-2 muestra un ejemplo de similitud que puede existir entre dos clases.

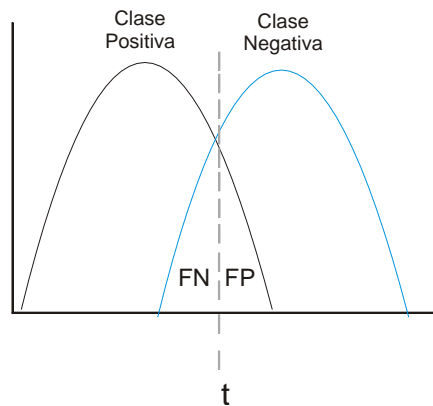


Figura 6-2: Representación de similitud entre dos clases. El punto de corte t determina el comportamiento del clasificador.

El comportamiento de pruebas depende del punto de corte t . Si este se desplaza a la derecha (Clase Negativa) disminuye la tasa de falsos positivos y aumenta la tasa de falsos negativos.

Inversamente si se desplaza a la izquierda (Clase Positiva) aumenta la tasa de falsos positivos pero disminuye la tasa de falsos negativos. Entonces, para caracterizar el comportamiento entre estas dos clases se utilizan las curvas ROC. Un ejemplo de curva ROC se muestra en la Figura 6-3 .

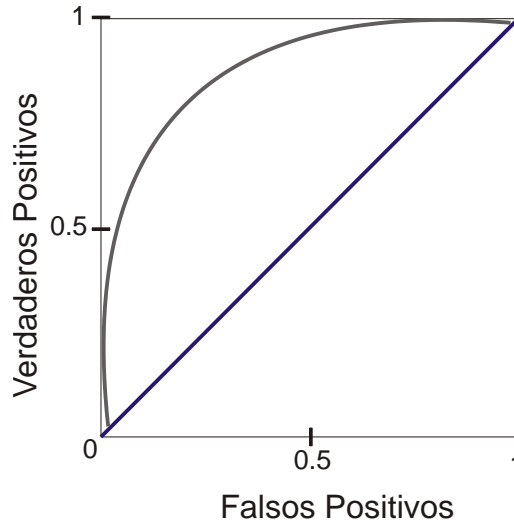


Figura 6-3: Curva ROC

Si la prueba fuera perfecta, es decir que no exista solapamiento entre clases, la curva solo tiene un punto $(0,1)$. Sin embargo, si la prueba fuera mala, la curva sería una diagonal de $(0,0)$ a $(1,1)$. La Figura 6-4 muestra un ejemplo de distintos tipos de solapamiento y los tipos de curvas ROC que se generan.

El parámetro para evaluar la bondad de la prueba, es el área bajo la curva ROC que toma valores entre 1 (prueba perfecta) y 0.5 (prueba fallida). Esta área puede interpretarse como la probabilidad de que un conjunto de datos ante un clasificador funcione correctamente.

6.2. Resultados

El objetivo primordial de esta investigación fue encontrar la mejor precisión de clasificación entre distintos clasificadores, distintos métodos de segmentación, distintos métodos de extracción de características y el impacto entre dos conjuntos de datos, un conjunto complejo y un conjunto trivial.

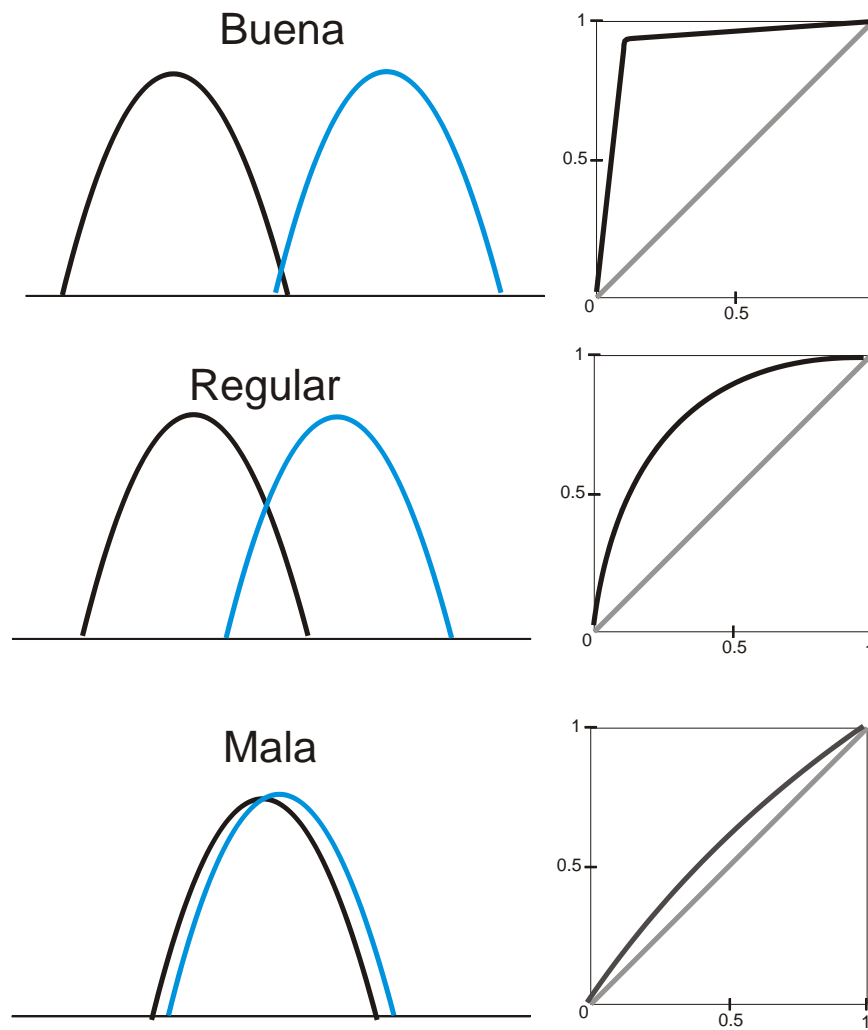


Figura 6-4: Tipos de curvas ROC

Los resultados obtenidos se analizan primero sobre el conjunto de datos complejo. Posteriormente sobre el conjunto de datos trivial, aunque también se hace una comparativa de resultados de ambos conjuntos al final.

Cada clasificador fue aplicado en todas las combinaciones de características y tipos de segmentación mencionados en la Tabla 5-4. Entonces, a continuación se discuten los resultados obtenidos sobre las 14 combinaciones. Además de analizar la precisión de clasificación, también se analiza el área ROC y F-Measure.

6.2.1. Conjunto de datos Complejo

Todos los resultados obtenidos de cada uno de los 11 subconjuntos que componen al conjunto complejo se muestran en las Tablas A-1 a A-14 del Apéndice A. Cada Tabla muestra una de las combinaciones entre características y tipos de segmentación de la Tabla 5-4. Las siglas SC1 hasta SC11 hacen referencia al subconjunto complejo 1 hasta el subconjunto complejo 11 respectivamente. Comenzando con la Tabla A-1, muestra los resultados de cada subconjunto al que se le aplicó una segmentación basada en Otsu y el conjunto de todas las características (Cromáticas, Geométricas y Texturales) hasta la Tabla A-14 que tiene los resultados de los subconjuntos a los que se les aplicó una segmentación basada en el método PCA y solo las características geométricas. Los resultados que muestran las mejores precisiones se enmarcan con letra Negrita.

En cada una de las tablas, se puede ver que cada clasificador se comportó de manera distinta dependiendo de cada una de las combinaciones entre características y segmentaciones. Los clasificadores SMO y Perceptron Multicapa fueron los que obtuvieron mejores resultados de clasificación. El clasificador SMO fue efectivo cuando se aplicó sobre características cromáticas no importando si estas se complementaban con características geométricas o texturales. Sin embargo, el Perceptron Multicapa fue el mejor clasificador cuando se aplicó a características geométricas y texturales.

El hecho de incluir cada vez más tipos de características en el conjunto de datos, permitía una mejor clasificación. El tipo de características con menor número, fueron las características texturales con solo 28 descriptores. Al implementar cada una de las características de manera independiente, se pudo notar que las características texturales fueron las que devolvieron

precisiones más bajas incluso menores al 50%. Las características geométricas y cromáticas devolvieron resultados similares e influyo el tipo de clasificador utilizado. Los clasificadores Bayes Net, y SMO se comportaron mejor con las características cromáticas, mientras que los clasificadores Navi Bayes, Perceptrón Multicapa, J48 y Random Forest trabajaron mejor con características geométricas. Aunque las características texturales mostraron un desempeño bajo en la clasificación, esto pueden no atribuirse al número de características sino más bien al tipo de características, ya que la mayoría de hojas cuenta con una textura similar. Las características texturales pueden ser eficientes cuando se trata de clasificar objetos con texturas antagónicas.

La diferencia de resultados entre características geométricas y texturales fue demasiada en todos los clasificadores. Tan solo comparando los resultados del SC1 donde el mejor clasificador fue el Perceptrón Multicapa, la precisión en características texturales fue de 69.55 y con características geométricas se elevó hasta 91.07. Sin embargo, se pudo ver una mejora al combinar estas características. El aumento del mismo SC1 con el mismo Perceptrón Multicapa fue de 92.07. Aunque hubo otros subconjuntos donde el aumento fue significativo, por ejemplo, en el SC2 donde con solo características geométricas y Perceptrón Multicapa la precisión fue de 87.35, al incluir las características texturales aumento a 92.85.

Tomando en cuenta solo las características cromáticas, donde el clasificador con mejor desempeño fue SMO, se pudo notar una mejor precisión con respecto a la combinación de características geométricas con texturales. En el mismo SC1, la precisión con SMO fue de 93.75, mayor que la obtenida con la combinación de características geométricas y texturales. Esta comparación es del Perceptrón Multicapa con SMO que devolvieron los mejores resultados. Ahora, hablando solo del clasificador SMO, la diferencia de precisión entre características texturales y geométricas con características cromáticas fue de 90.22 a 93.75, una diferencia más notable. Sin embargo, un punto importante es que las características texturales no influyeron en una mejor clasificación para SMO, ya que al combinar las características cromáticas con texturales la precisión bajo. Aunque el desempeño que afecto fue poco, es importante tomarlo en cuenta ya que se busca mejorar la clasificación y no empeorarla. Otra situación donde se puede notar que las características texturales afectan al clasificador, es al comparar los resultados de clasificación utilizando solo características cromáticas con los resultados donde se utilizan características cromáticas, texturales y geométricas. Cuando se comparan estas dos situaciones se puede notar

que la precisión aumenta. Esto se ve en el SC1, donde al utilizar solo características cromáticas con SMO se obtiene una precisión de 93.75 y al utilizar todas las características aumenta a 95.54. Sin embargo, al omitir las características texturales, quedando solo las características cromáticas con geométricas la precisión aumenta hasta 97.14. En el análisis hasta este punto se puede ver que las características que mejor impacto tuvieron, fueron las características cromáticas y geométricas combinadas y el mejor clasificador fue SMO. Aunque el Perceptrón Multicapa se comportó mejor con características geométricas y texturales, se pudo notar que fue el peor clasificador al añadir características cromáticas.

Ahora, analizando los clasificadores que obtuvieron el peor desempeño, fueron Navi Bayes y J48. El clasificador Navi Bayes mostro deficiencia principalmente con las características texturales, aunque sus resultados aumentaron con la combinación de otras características. Las situaciones donde principalmente Navi Bayes obtuvo un desempeño bajo, fue cuando se utilizaron las características individualmente y cuando se hizo la combinación de las tres. El Clasificador J48 se comportó deficiente cuando se combinaron las características cromáticas con texturales, y texturales con geométricas. Por otro lado se pudo ver que obtuvo un buen desempeño cuando se utilizaron solo características geométricas.

El impacto que tuvo la segmentación fue poca aunque relevante. El comportamiento de desempeño en los clasificadores fue demasiado similar entre las hojas segmentadas con Otsu y las hojas segmentadas con PCA. Sin en cambio, al realizar la comparativa de las mejores clasificaciones se pudo ver que los resultados de hojas segmentadas con PCA aumentaron respecto a las hojas segmentadas con Otsu. Tomando en cuenta el mejor resultado del SC1 con segmentación Otsu, que fue la situación, donde se utilizaron características cromáticas y geométricas con SMO la precisión fue de 97.14. Comparando la misma situación pero ahora con una segmentación basada en PCA la precisión aumento a 97.24. Aunque la diferencia es poca, la segmentación basada en PCA ayudo al clasificador a obtener un mejor desempeño en la mayoría de los subconjuntos.

Cada uno de los subconjuntos complejos se compone de distintos números de familias de hojas (ver Figura 5-4 y 5-5). El subconjunto con menor número de hojas es el SC6 que se compone con solo tres familias y en muchos casos alcanzo el 100 % de precisión. Esto pudiera llevar a concluir que el número de familias asociadas en cada subconjunto influye en una buena

clasificación. Sin embargo, los resultados mostraron que el número no influye. Por ejemplo, tomando en cuenta los mejores resultados de la Tabla A-10 en las clasificaciones con SMO el SC2 que cuenta con 8 familias asociadas tiene una precisión de 90.58, sin en cambio, el SC11 que cuenta con 37 familias asociadas tiene una precisión de 94.88. El desempeño es mayor aun cuando el número de familias entre subconjunto es alto. Tomando en cuenta el desempeño de los clasificadores respecto a los subconjuntos complejos, se notó que el SC11 fue el que más afecto en el desempeño de Bayes Net y Navi Bayes. El SC2 y SC3 fueron los que más afectaron el desempeño de los clasificadores SMO y J48.

6.2.2. Conjunto de datos Trivial

Los resultados de clasificación obtenidos para el conjunto de datos trivial, se muestran en la Tabla 6-2.

El análisis de estos datos se plasma en una sola tabla incluyendo las 14 combinaciones entre tipos de segmentación y tipos de características. La tabla 6-2 muestra los primeros 7 resultados con una segmentación Otsu y los siguientes con segmentación basada en PCA. El comportamiento de cada clasificador fue similar al comportamiento en los subconjuntos complejos. El clasificador con mejor precisión en características texturales y geométricas fue el Perceptrón Multicapa y en características cromáticas fue el clasificador SMO. De igual manera, las características texturales afectaron la precisión en SMO. Además los resultados de la segmentación basada en el método PCA tuvieron mejora en comparación con los resultados de segmentación con el método OTSU.

El conjunto trivial está compuesto de 90 familias de hojas de planta. Sin embargo, se puede ver en los resultados, que el hecho de ser un conjunto trivial si influye en una buena clasificación.

Si comparamos el mejor resultado de la Tabla 6-2 que se encuentra en la segmentación basada en PCA con características Texturales y Geométricas y la Tabla A-10 donde se encuentra los mejores resultados de los subconjuntos complejos se puede ver que la precisiones están en un rango alto aun cuando le conjunto trivial se compone de 90 familias. Incluso la precisión del conjunto trivial que es 93.95 es más alta la del SC2 con 90.58 que solo se compone de 8 familias.

Tabla 6-2: Resultados del desempeño de clasificación, F-Measure y Área-ROC con cada uno de los clasificadores para el conjunto Trivial

		BayesNet	NaviBayes	Perceptrón Multicapa	SMO	J48	Random Forest
Otsu	Precisión	88.8413	73.4204	6.2325	92.3752	76.2262	81.3665
CTG	F-Measure	0.889	0.745	0.098	0.923	0.76	0.807
	Área ROC	0.998	0.917	0.457	0.993	0.889	0.975
Otsu	Precisión	81.3062	68.5653	6.6595	89.935	64.6467	72.334
CTX	F-Measure	0.811	0.696	0.091	0.898	0.645	0.709
	Área ROC	0.995	0.916	0.576	0.935	0.834	0.955
Otsu	Precisión	87.8337	73.7861	4.7482	91.2381	77.065	80.8457
CXG	F-Measure	0.879	0.751	0.038	0.912	0.769	0.8
	Área ROC	0.998	0.927	0.122	0.994	0.896	0.973
Otsu	Precisión	79.8716	68.4436	6.1151	87.8409	66.7855	72.1697
CXX	F-Measure	0.798	0.698	0.061	0.876	0.667	0.71
	Área ROC	0.994	0.925	0.384	0.984	0.843	0.954
Otsu	Precisión	85.9037	81.0481	91.9786	88.1925	78.8877	88.7273
XTG	F-Measure	0.857	0.804	0.918	0.877	0.787	0.884
	Área ROC	0.998	0.99	0.991	0.996	0.911	0.99
Otsu	Precisión	26.5825	22.4979	55.2823	28.8922	42.2797	48.9307
XTX	F-Measure	0.244	0.191	0.54	0.227	0.421	0.479
	Área ROC	0.896	0.88	0.928	0.915	0.739	0.88
Otsu	Precisión	79.6972	73.7845	88.7444	77.3998	81.2289	90.7391
XXG	F-Measure	0.793	0.73	0.883	0.759	0.81	0.904
	Área ROC	0.995	0.984	0.983	0.99	0.926	0.992
PCA	Precisión	88.4886	73.2877	5.0727	93.6429	75.9846	79.3022
CTG	F-Measure	0.885	0.744	0.051	0.937	0.76	0.786
	Área ROC	0.998	0.919	0.121	0.995	0.889	0.974
PCA	Precisión	81.0788	68.1507	4.1755	88.9554	64.2123	71.8108
CTX	F-Measure	0.81	0.692	0.056	0.892	0.64	0.708
	Área ROC	0.995	0.916	0.254	0.964	0.832	0.982
PCA	Precisión	87.7853	44.0068	5.0641	94.0263	77.1041	80.4387
CXG	F-Measure	0.878	0.441	0.05	0.94	0.77	0.796
	Área ROC	0.998	0.716	0.123	0.994	0.897	0.973
PCA	Precisión	79.373	69.1486	4.9875	87.5133	66.637	73.4592
CXX	F-Measure	0.793	0.704	0.049	0.875	0.666	0.725
	Área ROC	0.994	0.926	0.215	0.986	0.844	0.956
PCA	Precisión	86.3403	81.5306	93.9504	89.2048	79.7349	88.9483
XTG	F-Measure	0.862	0.809	0.938	0.887	0.797	0.887
	Área ROC	0.998	0.991	0.996	0.996	0.917	0.991
PCA	Precisión	28.2386	22.6165	56.5413	29.5853	40.3005	49.9786
XTX	F-Measure	0.26	0.193	0.56	0.233	0.403	0.491
	Área ROC	0.899	0.883	0.938	0.917	0.728	0.886
PCA	Precisión	79.9858	73.0413	89.1026	79.0954	79.7899	91.4708
XXG	F-Measure	0.796	0.721	0.888	0.779	0.797	0.913
	Área ROC	0.995	0.984	0.985	0.992	0.925	0.994

6.2.3. Mejora de resultados con Algoritmo Genético

Con la implementación del AG en los datos anteriores se pudieron obtener mejoras en el desempeño de los clasificadores. Aunque anteriormente se discutió sobre las características que más influyeron y los clasificadores que obtuvieron mejores resultados, se aplicó el AG a todos los conjuntos de datos con el fin de hacer un análisis más profundo sobre las características que más influyeron para una buena clasificación.

Como se había previsto la mayoría de las precisiones aumentaron con el AG. La comparación entre los resultados previos al aplicar el AG y los resultados posteriores al AG se pueden visualizar en las Tablas B-1 a B-14 del Apéndice B. Sin embargo esta misma comparación de resultados también se puede visualizar en las gráficas de las Figuras 6-5 a 6-18 que se encuentran al final de este apartado. Estas gráficas no muestran el aumento de precisión de forma individual, más bien muestran el promedio de precisiones de los 6 clasificadores de cada subconjunto complejo y el conjunto trivial. En total, se presentan 14 graficas que representan a cada una de las combinaciones de la Tabla 5-4. Cada grafica muestra el promedio de precisiones obtenidas con todas las características con barras azules y el promedio de precisiones después de aplicar el AG con barras verdes. En este caso, se juntaron los datos del conjunto trivial con los datos de los subconjuntos complejos.

El AG mejoro los resultados de forma distinta en cada combinación, esto se puede ver en la gráficas, ya que en algunas se muestra una diferencia más significativa que otras como es el caso de las gráficas 6-5,6-9 y 6-10 con las gráficas 6-6,6-7,6-8 y 6-11. Las primeras gráficas muestran más diferencias en resultados a comparacion de las otras. Aunque algunas gráficas muestran más diferencias en resultados que otras, no se presentó el caso en que la precisión disminuyera. En la mayoría de los casos hubo aumento de precisión. La peor situación que presentó el AG fue mantener la misma precisión, aunque aun así, se obtuvo un beneficio, ya que esta misma precisión se obtuvo con solo la mitad de características empleadas.

Con la implementación del AG se logró mostrar que muchas características solo aportan ruido al clasificador, que en vez de beneficiarlo, lo confunden. La gran mayoría de datos redujo en promedio la mitad o menos de las características iniciales después de aplicar al AG. Esto beneficia el costo computacional cuando las características son de gran número como es el caso de las características cromáticas donde de 889 características se redujo a 442 en promedio.

Al disminuir gran cantidad de características el proceso de entrenamiento de un clasificador se vuelve más rápido. La Tabla 6-3 muestra el total de características promedio que utilizó el AG para mejorar la precisiones.

Tabla 6-3: Reducción de características promedio en cada combinación de características después de aplicar el Algoritmo Genético

Reducción de características promedio al aplicar AG		
	Características iniciales	Características reducidas
Cromáticas, Texturales y Geométricas	943	466
Cromáticas y Texturales	889	442
Cromáticas y Geométricas	915	454
Cromáticas	861	427
Texturales y Geométricas	82	42
Texturales	28	10
Geométricas	54	27

El conjunto de características donde la precisión aumento mejor, fueron los datos con características texturales. La forma en como aumentaron las precisiones en cada combinación de características fue similar en cuanto al tipo de segmentación, es decir, la precisión promedio de características cromáticas, texturales y geométricas fue similar en los datos segmentados con PCA y los datos segmentados con Otsu. La Tabla 6-4 muestra la precisión promedio que aumento en cada combinación de características y tipos de segmentación.

Tabla 6-4: Precisión promedio alcanzada en cada combinación de características y tipo de segmentación

Mejora de precision promedio con AG en tipos de segmetación		
	PCA	Otsu
Cromáticas, Texturales y Geométricas	1.6	1.5
Cromáticas y Texturales	2.4	2.3
Cromáticas y Geométricas	1.4	2.0
Cromáticas	2.7	2.8
Texturales y Geométricas	2.0	1.9
Texturales	3.7	3.6
Geométricas	2.0	1.7

La forma en como aumentaron las precisiones de manera individual en la mayoría de datos fue de 2 a 5 por ciento sobre la precisión anterior. Salvo algunos casos donde la precisión subió hasta un 30% más, como el caso del subconjunto complejo 2 con segmentación OTSU y características cromáticas entrenado con el clasificador Random Forest. En este conjunto

inicialmente se obtuvo una precisión de 52.08 y logró aumentarse hasta 84.50 con el AG (ver Tabla B-4 del apéndice B). En muy pocas ocasiones la precisión subió de 10 a 15 por ciento.

Para encontrar la relación de características se realizaron dos comparativas, la primera entre clasificadores y la segunda entre conjuntos. La relación de características que se eliminaron y se mantuvieron constantemente fue mínima de un clasificador a otro. Sin embargo si se encontró fuerte relación entre los distintos conjuntos (subconjuntos complejos y conjunto trivial). La comparativa se realizó analizando las características que se mantuvieron o eliminaron frecuentemente en un mismo clasificador o un mismo conjunto. Se tomaron como características utilizadas aquellas que se mantuvieron en al menos 5 de los 6 clasificadores y se tomaron como características eliminadas aquellas que se omitieron en al menos 5 de los 6 clasificadores. La relación de características entre conjuntos se realizó tomando en cuenta los 12 conjuntos (11 subconjuntos complejos y 1 conjunto trivial). En este caso se tomaron como características utilizadas aquellas que permanecieron en al menos 10 conjuntos y como características eliminadas aquellas que fueron omitidas en al menos 10 conjuntos. El hecho de que una característica no se encuentre dentro de estos dos tipos es debido a que su uso fue muy aleatorio en todas las situaciones. El resultado de este análisis se puede ver en la Tabla 6-5 donde se mencionan las características que más destacaron por ser eliminadas o utilizadas por el AG y así obtener una buena precisión.

Los descriptores de Hog piramidal forman casi el 90 % de las características cromáticas por lo que era de esperar, estos descriptores fueron utilizados y eliminados en gran proporción de los conjuntos de datos donde se utilizaron características cromáticas. Los momentos Hu Azul fueron eliminados constantemente donde había características cromáticas, mientras que los momentos de Hu verde fueron los más utilizados en este mismo tipo de características. La elipsidad (descriptor geométrico) fue la característica más eliminada cuando no se utilizaron características cromáticas, mientras que los descriptores geométricos más utilizados en todas las combinaciones con este tipo de características fueron los descriptores de Fourier, el centro de gravedad y el número Euler.

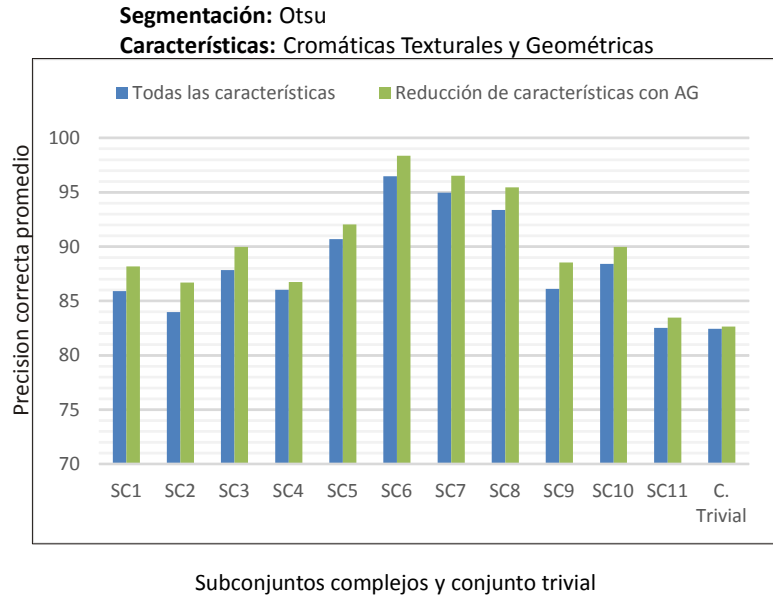


Figura 6-5: Representación gráfica del aumento de precisión promedio al disminuir características con el AG. Los resultados se basan en segmentación Otsu y características cromáticas, texturales y geométricas.

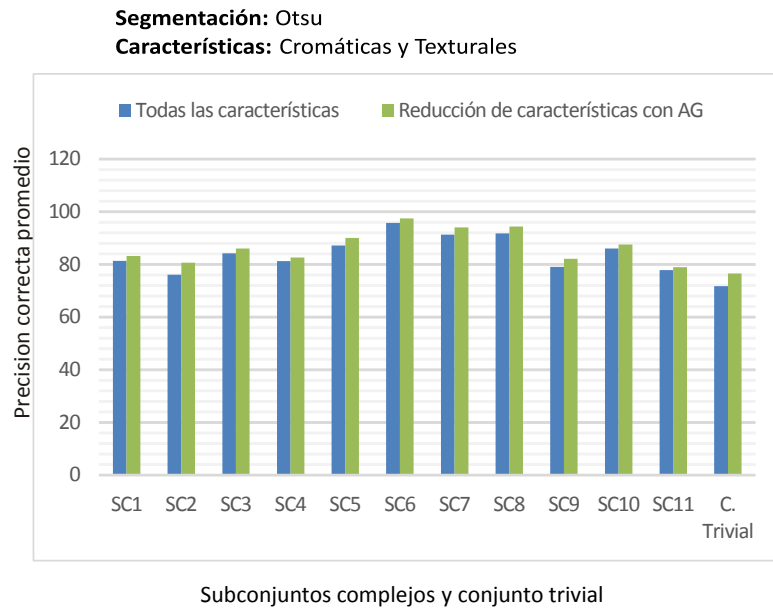


Figura 6-6: Representación gráfica del aumento de precisión promedio al disminuir características con el AG. Los resultados se basan en segmentación Otsu y características cromáticas y texturales.

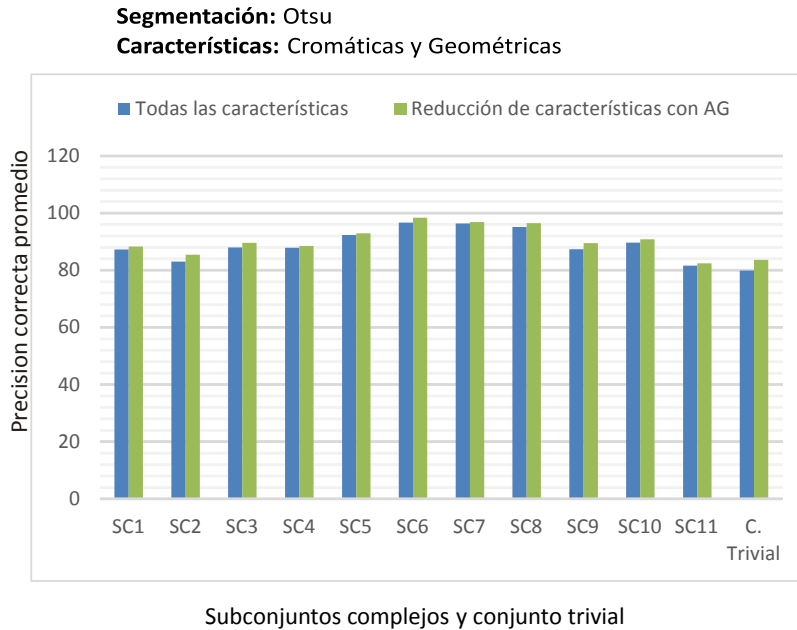


Figura 6-7: Representación gráfica del aumento de precisión promedio al disminuir características con el AG. Los resultados se basan en segmentación Otsu y características cromáticas y geométricas.

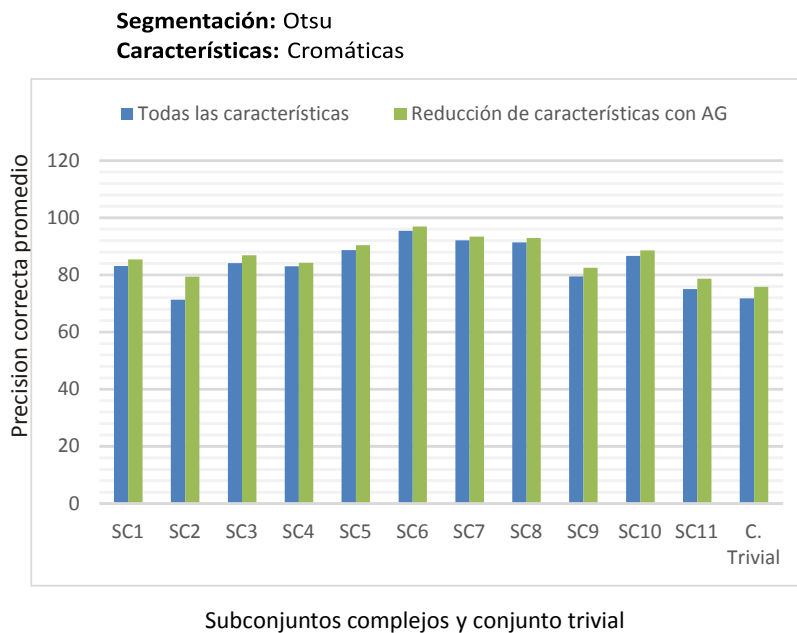


Figura 6-8: Representación gráfica del aumento de precisión promedio al disminuir características con el AG. Los resultados se basan en segmentación Otsu y características cromáticas.

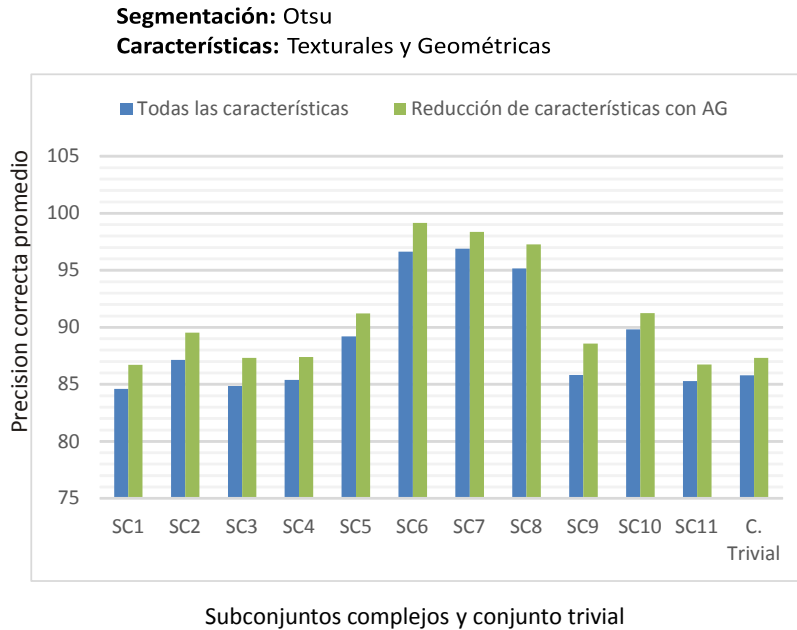


Figura 6-9: Representación gráfica del aumento de precisión promedio al disminuir características con el AG. Los resultados se basan en segmentación Otsu y características texturales y geométricas.

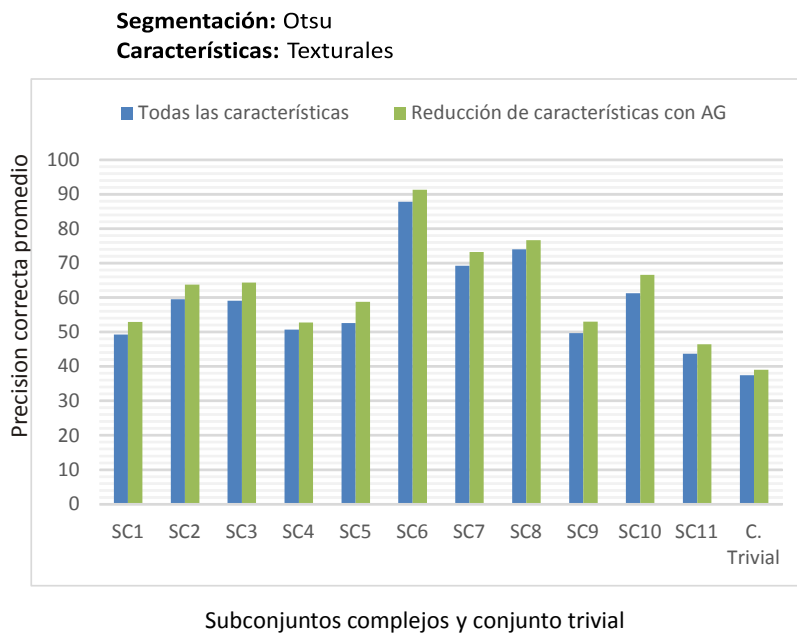


Figura 6-10: Representación gráfica del aumento de precisión promedio al disminuir características con el AG. Los resultados se basan en segmentación Otsu y características texturales.

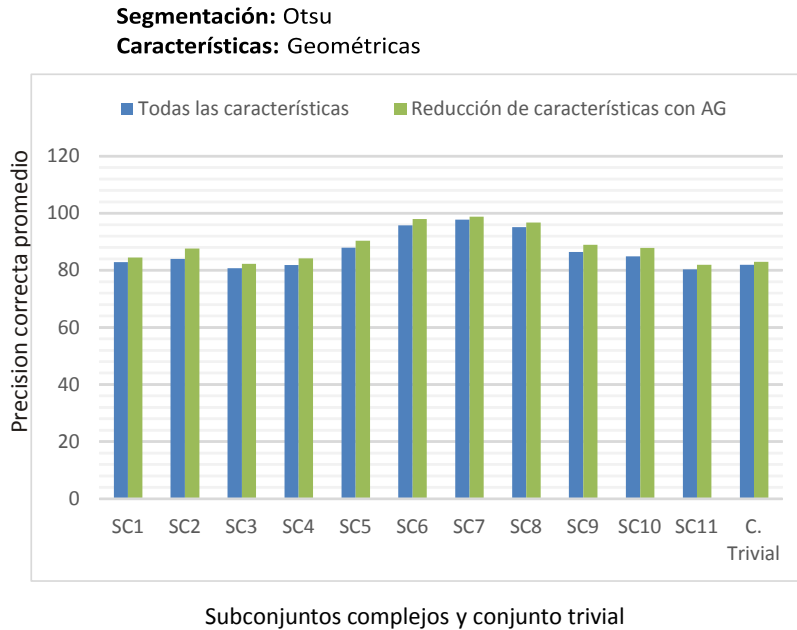


Figura 6-11: Representación gráfica del aumento de precisión promedio al disminuir características con el AG. Los resultados se basan en segmentación Otsu y características geométricas.

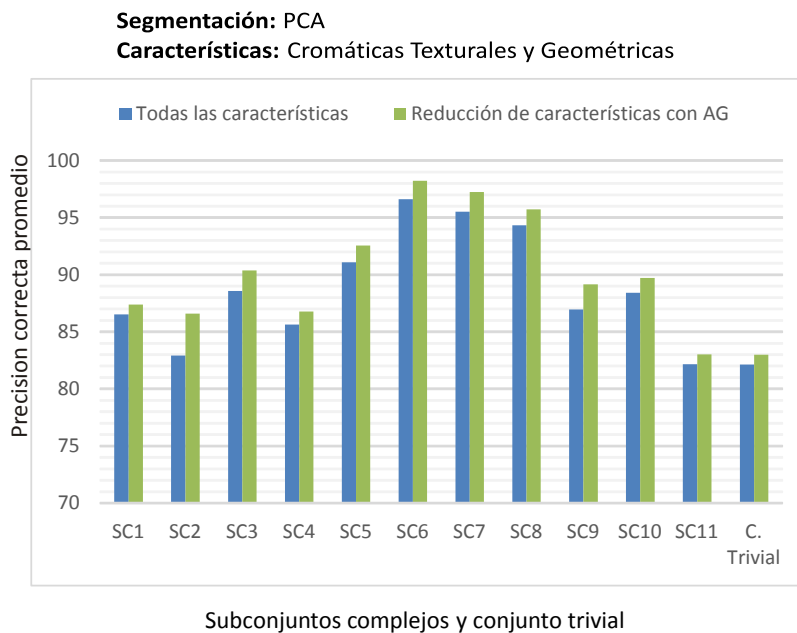


Figura 6-12: Representación gráfica del aumento de precisión promedio al disminuir características con el AG. Los resultados se basan en segmentación PCA y características cromáticas, texturales y geométricas.

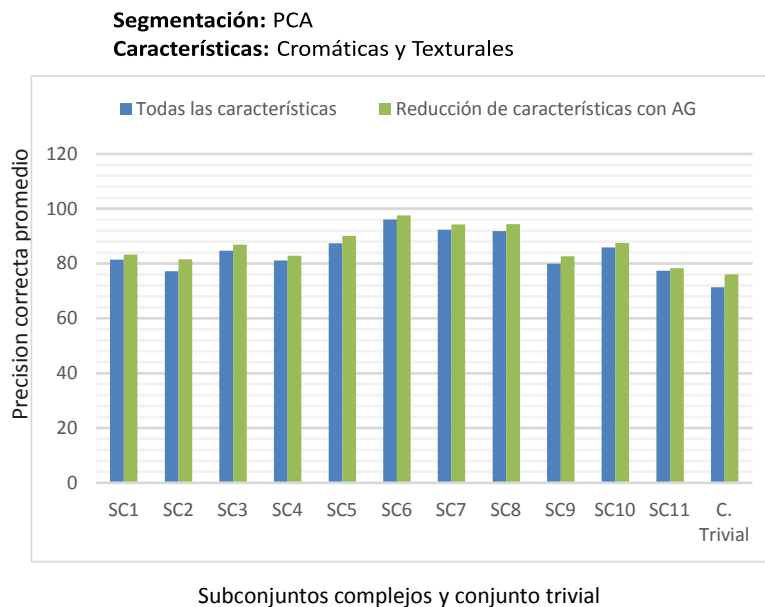


Figura 6-13: Representación gráfica del aumento de precisión promedio al disminuir características con el AG. Los resultados se basan en segmentación PCA y características cromáticas y texturales.

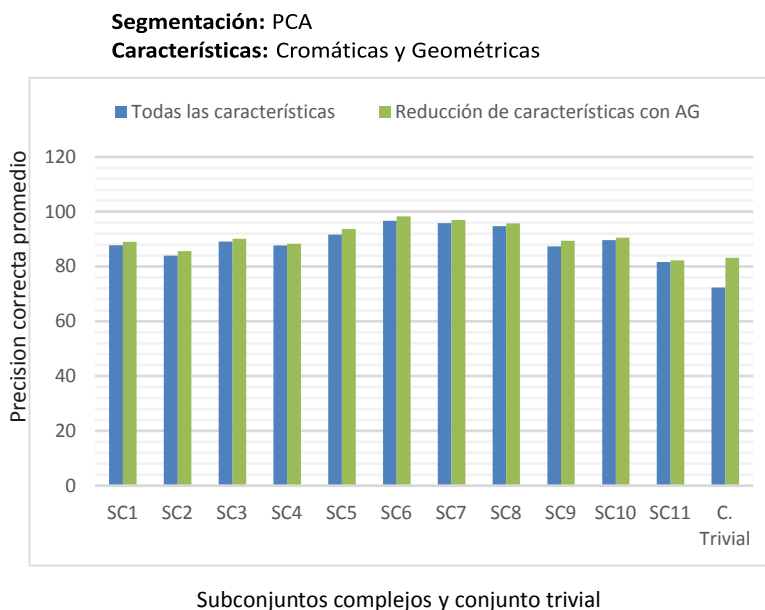


Figura 6-14: Representación gráfica del aumento de precisión promedio al disminuir características con el AG. Los resultados se basan en segmentación PCA y características cromáticas y geométricas.

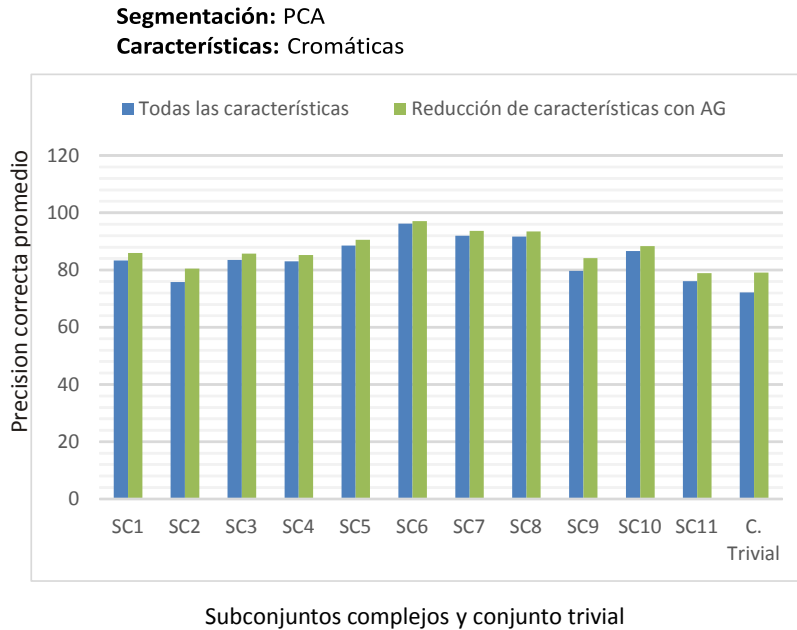


Figura 6-15: Representación gráfica del aumento de precisión promedio al disminuir características con el AG. Los resultados se basan en segmentación PCA y características cromáticas.

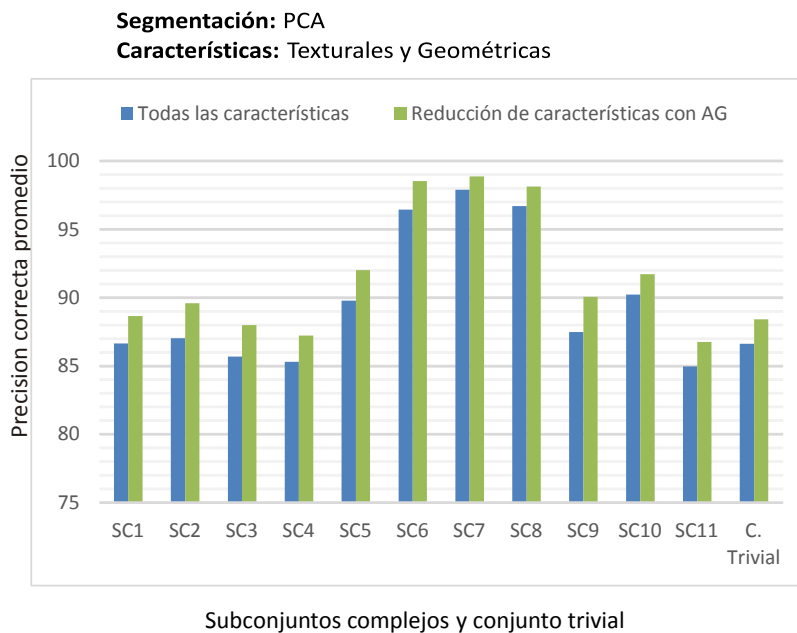


Figura 6-16: Representación gráfica del aumento de precisión promedio al disminuir características con el AG. Los resultados se basan en segmentación PCA y características texturales y geométricas.

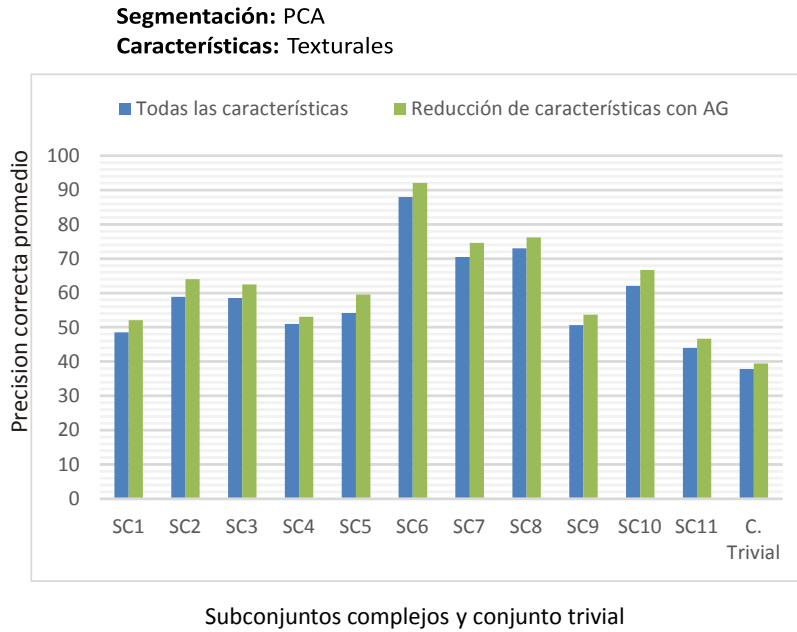


Figura 6-17: Representación gráfica del aumento de precisión promedio al disminuir características con el AG. Los resultados se basan en segmentación PCA y características texturales.

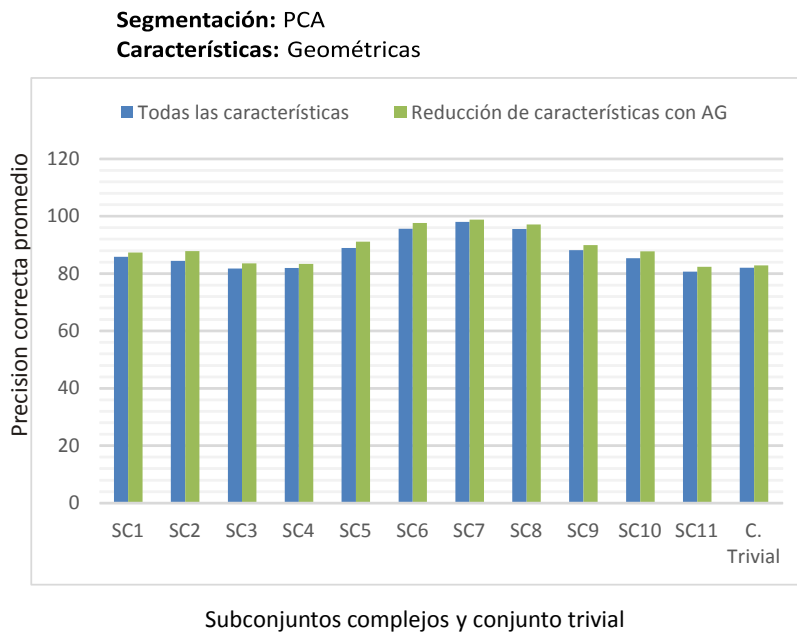


Figura 6-18: Representación gráfica del aumento de precisión promedio al disminuir características con el AG. Los resultados se basan en segmentación PCA y características geométricas.

Tabla 6-5: Características más eliminadas y utilizadas por el Algoritmo Genético para obtener mejores precisiones

Relación de características		
	Características eliminadas	Características utilizadas
Cromáticas Texturales Geométricas	* 20 de Hog piramidal (Cromáticas) * Momentos de Hu (Geométricas)	* 20 de Hog piramidal, momentos de Hu verde (Cromáticas) * Descriptores de Fourier 1,4 y 5, centro de gravedad y Redondez (Geométricas)
Cromáticas Texturales	* 27 de Hog piramidal, momento de Hu azul 7 y 20 Gabor (Cromáticas) * 5 Descriptores Haralick (Texturales)	* 20 de Hog piramidal, momentos de Hu azul, verde y descriptores de Fourier 2 y 3(Cromáticas). * Descriptores Haralick 6,7, 11 y 12 (Texturales).
Cromáticas Geométricas	* 27 de Hog piramidal y momentos de Hu azul (Cromáticas)	* 16 de Hog piramidal, 8 descriptores Gabor y momentos Fourier 3, momento de Hu rojo 3 y momentos de Hu Verde (Cromáticas). * Momento de Hu, centro de gravedad, elipsidad y número euler (Geométricas)
Cromáticas	* 46 de Hog piramidal y 2 descriptores Gabor.	* Momentos de Hu verde y azul, 25 descriptores Hog piramidal, 10 descriptores Gabor y momento de Fourier 4.
Texturales Geométricas	* Descriptores Haralick 1 y 9 (Texturales). * Elipsidad (Geométricas)	* Descriptores de Fourier 6 y 7, área, número euler y diametro (Geométricas).
Texturales	* Descriptor Haralick 1	* Descriptor Haralick 7.
Geométricas	* Descriptor de Fourier 1, anchura y elipsidad.	* Descriptores de Fourier 4, 5, 6 y 8, Centro de gravedad, altura, número euler y área.

Capítulo 7

Conclusiones

En esta tesis se desarrolló un sistema de identificación de hojas de plantas y se realiza un análisis comparativo utilizando diferentes métodos de segmentación y extractores de características. Los resultados obtenidos por el sistema están basados en hojas con ambiente totalmente controlado. Por lo que si se introducen al sistema imágenes de hojas solapadas o con un fondo complejo, puede influir en la precisión de clasificación. A partir de los resultados obtenidos se puede deducir lo siguiente:

Primero, el tiempo de entrenamiento fue muy enorme en conjuntos de datos con información de demasiadas familias de hojas y conjuntos de datos con características de gran tamaño. El conjunto más grande fue el conjunto trivial que contenía los datos de 90 familias. Otros conjuntos con gran tamaño pertenecieron al conjunto de hojas complejas y fueron los subconjuntos 11 y 4 que contenían datos de 37 y 27 familias respectivamente. Los demás subconjuntos tuvieron un tamaño medio o pequeño. El conjunto más pequeño fue el subconjunto complejo 6, que contaba con solo datos de 3 familias. Las características de gran tamaño fue otro factor que influyó en el desempeño del clasificador. Las características geométricas y texturales fueron las de menor tamaño, una tuvo 54 otra 28 características respectivamente, por lo que no hubo problemas durante el entrenamiento. Incluso al combinar características texturales y geométricas no existió demora, ya que juntas formaban un total de 82 características. Sin embargo, el número de características cromáticas fueron 861 que aumentó al combinarlas con las geométricas y las texturales. La combinación de características cromáticas con texturales tuvo un total de 889, las características cromáticas con geométricas un total de 915 y la combinación de todas las

características fue de 943. Los conjuntos de datos donde los clasificadores tardaron más en devolver resultados fue cuando trabajaron con archivos que contenían datos del conjunto trivial junto con cualquier combinación de características cromáticas. De todos los clasificadores, se notó que el Perceptrón multicapa fue el clasificador con mayor costo computacional ya que tardó 10 veces más en devolver resultados. Los clasificadores restantes tardaron el mismo tiempo promedio para entrenar los datos.

Segundo, sobre el desempeño de los clasificadores. A partir de los resultados obtenidos se pudo observar que los mejores clasificadores fueron el Perceptron multicapa y SMO. Sin embargo, cada uno de estos clasificadores trabajó mejor dependiendo de los datos de entrada. Es decir, el Perceptron multicapa devolvió precisiones bajas cuando trabajó con características cromáticas, pero devolvió las precisiones altas cuando trabajó con descriptores geométricos y texturales. Por otro lado, se notó que el clasificador SMO devolvió las precisiones más altas cuando trabajó con características cromáticas, y devolvió precisiones bajas cuando trabajó con características geométricas y texturales. Aun así se notó que de los dos clasificadores el mejor fue SMO ya que aunque el Perceptron multicapa tuvo mejores resultados en datos que omitieron las características cromáticas, SMO logró mejores resultados al combinar características cromáticas y geométricas. A pesar de que SMO junto con características cromáticas se comportó de mejor manera que los demás, este demandó un costo computacional mucho mayor a los demás clasificadores (sin tomar en cuenta el Perceptron multicapa, ya que este tardó más tiempo en entrenarse que SMO).

La mejora entre clasificadores que se esperaba al trabajar con hojas triviales y hojas complejas no fue significativa, es decir, no fue posible inferir que la similaridad entre hojas afecte la precisión de los clasificadores ya que el desempeño de los clasificadores utilizados con imágenes muy similares entre si y disimilares no fueron contrastantes. Esto se le puede atribuir al que el conjunto de datos de hojas trivial fue demasiado grande y como se mencionó anteriormente el tamaño de los conjuntos de datos influyó en la precisión de los clasificadores.

En este trabajo también se pudo demostrar que no es necesario utilizar demasiadas características para una mejor clasificación. Ya que del conjunto de todas las características (Cromáticas, Texturales y Geométricas) se notó que la mejor combinación de características para clasificar hojas de planta fueron las cromáticas con geométricas.

El tipo de segmentación utilizado antes de extraer características también influye en la precisión de clasificación. En este trabajo de investigación se utilizaron dos tipos de segmentación, a saber, el método Otsu y otra segmentación basada en el método PCA. De los resultados obtenidos de estos dos tipos de segmentación, se pudo ver que la segmentación basada en el método PCA logró que se obtuvieran mejores resultados de clasificación. Por lo tanto, como conclusión, en un sistema de identificación de plantas, se puede recomendar utilizar una segmentación basada en el método PCA, utilizar características cromáticas junto con características geométricas y utilizar un clasificador SMO. Solo restaría analizar el tipo de hojas a identificar para eliminar aquellas características que no influyen en la precisión de clasificación.

Tercero, la mejora en el desempeño del clasificador. Otro punto importante en este proyecto fue el análisis de características bajo la implementación de un Algoritmo genético. El Algoritmo genético implementado en las características mostro que se puede llegar a una mejor clasificación omitiendo muchas de esas características. En nuestro caso se demostró que de todas las características utilizadas, solo se necesita el 50% de estas, con lo que se logra la misma precisión de clasificación o una mayor. Aunque cada conjunto mantuvo y eliminó características distintas se pudo ver que al momento de utilizar características cromáticas con geométricas, los momentos de Hu azul no influían para una buena clasificación mientras que si era importante utilizar la mayoría de características geométricas.

7.1. Trabajo futuro

Se está obteniendo una base de datos con imágenes de hojas de plantas con gran similitud, es decir, conjuntos complejos. A diferencia de las imágenes utilizadas para formar los conjuntos complejos en el trabajo de tesis, estas hojas tienen una similitud aun mayor, ya que forman parte de una misma especie. En el campo de la botánica, es común encontrar hojas de planta con mucho parecido cuando estas forman parte de la misma especie. El parecido de estas hojas es enorme y es una tarea compleja para los sistemas de identificación incluso para los especialistas en la materia, ya que sus diferencias son estrechamente pequeñas. Para el análisis de este tipo de hojas se requiere de un estudio más a fondo de esas características que diferencian una especie de otra.

En esta base de datos se pretende utilizar los algoritmos implementados en el trabajo de tesis y solo las técnicas con las que se obtuvieron mejores resultados. Además se pretende añadir una comparativa entre técnicas utilizadas por el Algoritmo Genético. Realizar este análisis resultara de gran impacto para los especialistas en la botánica.

7.2. Publicaciones

Durante el desarrollo de este trabajo, se presentaron los resultados en dos publicaciones científicas. El primero artículo llamado ‘Mejorando la Clasificación en Conjuntos de Datos No-balanceados Utilizando un Algoritmo Genético para Selección de Atributos’ fue publicado en el Quinto Congreso Internacional de Computación México-Colombia y XV Jornada Académica en Inteligencia Artificial que se celebró los días 24, 25 y 26 de septiembre de 2015 en Cartagena de Indias (Colombia). El segundo artículo llamado ‘Análisis Comparativo de las técnicas utilizadas en un Sistema de Reconocimiento de Hojas de Planta’ fue publicado en la Revista Iberoamericana de Automática Industrial. Ambos artículos se encuentran en el Apéndice D de esta tesis.

Bibliografía

- [1] Villasenor. J.L. and Murguía. M. (1992) La computadora en la identificación botánica: una grata experiencia. *Ciencia y Desarrollo* 18,130-137.
- [2] Mitchell, T.M. *Machine Learning*, McGraw-Hill 1997
- [3] Robert M. Haralick, K Shanmugam and Its'Hak Dinstein. "Textural Features for Image Classification." *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. SMC-3, No. 6, November 1973.
- [4] Robert M. Haralick. "Statistical and Structural Approaches to Texture", *Proceedings of the IEEE*, Vol.67, No. 5, May 1979.
- [5] A. Taza and C. Suen, "Discrimination of planar shapes using shape matrices", *IEEE Trans. System, Man, and Cybernetics*, vol. 19(5), pp. 1281-1289, 1989.
- [6] J. Flusser, "Invariant shape description and measure of object similarity", in *Proc. 4th International Conference on Image Processing and its Application*, 1992, pp. 139-142.
- [7] J. Flusser and Tomas Suk, "Pattern recognition by affine moment invariants", *Institute of Information Theory and Automation, Czechoslovak Academy of Sciences*, June 1992.
- [8] M. Peura and J. Iivarinen, "Efficiency of simple shape descriptors", in *Proc, 3rd International Workshop on Visual Form*, May 1997.
- [9] Villasenor. J.L. and Murguía. M. (1998) *Gencomex: a Computerized Key to Identify the Genera of Asteraceae of Mexico*.

- [10] K. Chakrabarti, M. Binderberger, K. Porkaew, and S. Mehrotra, "Similar shape retrieval in mars", in Proc. IEEE International Conference on Multimedia and Expo, 2000.
- [11] Platt J.C., "Fast Training of Support Vector Machines using Sequential Minimal Optimization", Microsoft Research, EUA, 2000.
- [12] Mario Koppen. The Curse of Dimensionality. (held on the internet), September 4-18 2000. 5th Online World Conference on Soft Computing in Industrial Applications (WSC5).
- [13] D. S. Zhang and G. Lu, ".A comparative study on shape retrieval using fourier descriptors with different shape signatures", in Proc. International Conference on Intelligent Multimedia and Distance Education, 2001.
- [14] Leo Breiman, Random Forests, Statistics Department, University of California Berkeley, January 2001.
- [15] Guillermo Sampallo. Reconocimiento de tipos de hojas. Inteligencia Artificial. Revista Iberoamericana de Inteligencia Artificial, vol. 7, núm. 21, 2003, pp. 55-62, Asociación Española para la Inteligencia Artificial España.
- [16] L. Gonzáles Abril, Modelos de Clasificación basados en Máquinas de Vectores Soporte, Departamento de Economía Aplicada I Universidad de Sevilla. (2003).
- [17] Dengsheng Zhang, Guojun Lu, Review of shape representation and description techniques, Pattern Recognition, Volume 37, Issue 1, January 2004, Pages 1-19, ISSN 0031-3203.
- [18] Miriam Presutti, "La matriz de co-ocurrencia en la clasificación multispectral: Tutorial para la enseñanza de medidas texturales en cursos de grado universitario"4ta Jornada de educación en sensoriamiento remoto, Agosto 2004.
- [19] D. G. Lowe. Distinctive image features from scale-invariant keypoints. Int'l. J. of Computer Vision, 60(2):91-110, 2004.
- [20] F. Mindru, T. Tuytelaars, L. Van Gool, and T. Moons. Moment invariants for recognition under changing viewpoint and illumination. Computer Vision and Image Understanding, 94(1-3):3-27, 2004.

- [21] N.Dalal, B.Triggs, Histograms of oriented gradients for human detection, in: CVPR, 2005, pp.886–893.
- [22] Gustavo A. Betancourt, Las Máquinas de Soporte Vectorial (SVMs), Scientia et Technica Año XI, No 27, Abril 2005.
- [23] Bruno López Takeyas, Algoritmo C4.5, Instituto Tecnológico Nuevo Laredo, Noviembre 2005.
- [24] Tom Fawcett, An introduction to ROC analysis, Pattern Recognition Letters, Volume 27, Issue 8, June 2006, Pages 861-874, ISSN 0167-8655.
- [25] J. van deWeijer, T. Gevers, and A. Bagdanov. Boosting color saliency in image feature detection. IEEE Trans. Pattern Analysis and Machine Intell., 28(1): 150–156, 2006.
- [26] Evangelista, Paul F., Mark J. Embrechts, and Boleslaw K. Szymanski. "Taming the curse of dimensionality in kernels and novelty detection." In Applied soft computing technologies: The challenge of complexity, pp. 425-438. Springer Berlin Heidelberg, 2006.
- [27] Ji-Xiang Du, Xiao-Feng Wang, Guo-Jun Zhang, Leaf shape based plant species recognition, Applied Mathematics and Computation, Volume 185, Issue 2, 15 February 2007, Pages 883-893, ISSN 0096-3003.
- [28] D. Chaudhuri, A. Samal, A simple method for fitting of bounding rectangle to closed regions, Pattern Recognition, Volume 40, Issue 7, July 2007, Pages 1981-1989, ISSN 0031-3203.
- [29] Yong Kui Liu, Wei Wei, Peng Jie Wang, Borut Žalik, Compressed vertex chain codes, Pattern Recognition, Volume 40, Issue 11, November 2007, Pages 2908-2913, ISSN 0031-3203.
- [30] A.Bosch, A.Zisserman, X.Munoz, Representing shape with a spatial pyramid kernel, in: CIVR, 2007, pp.401–408.
- [31] Gonzalo Pajares Martinsanz, Jesús M. de la Cruz García, Ejercicios resueltos de Visión por Computador, Alfaomega, 2008.

- [32] A. Bosch, A. Zisserman, and X. Muoz. Scene classification using a hybrid generative/discriminative approach. *IEEE Trans. Pattern Analysis and Machine Intell.*, 30(04):712–727, 2008.
- [33] Vizcaino Garzon Paula Andrea, "Aplicación de técnicas de inducción de árboles de decisión a problemas de clasificación mediante el uso de weka", Fundación universitaria Konrad Lorenz, Facultad de Ingeniería de sistemas, 2008.
- [34] Lourdes Araujo, Carlos Cevigón, *Algoritmos evolutivos un enfoque práctico*, Alfaomega Grupo Editor, S. A. de C. V., Mexico, 2009.
- [35] C. Cattaneo, L. Larcher, A. Ruggeri, A. C. Herrera, E. Biasoni, M. Escañuelas. Segmentación de imágenes digitales mediante umbralizado adaptativo en imágenes de color. *Mecánica Computacional XXIX*, 6177-6193, 2010.
- [36] Erik Cuevas, Daniel Zaldívar, Marco Pérez, *Procesamiento digital de imágenes con MATLAB y Simulink*, Alfaomega, 2010.
- [37] Koen E. A. van de Sande, Theo Gevers, Cees G. M. Snoek, "Evaluating of Color Descriptors for Object and Scene Recognition", *IEEE Transactions on Pattern Analysis & Machine Intelligence*, vol.32, no. 9, pp. 1582-1596, September 2010.
- [38] Pedro Ponce Cruz, *Inteligencia artificial con aplicaciones la ingeniería*, Primera edición, Alfaomega Grupo Editor, S. A. de C. V., Mexico, 2010.
- [39] Duber Martínez T., Humberto Loaiza C., Eduardo Caicedo B. Una propuesta para incrementar por medio de Algoritmos Genéticos la capacidad discriminante de las técnicas PCA y LDA aplicadas al Reconocimiento de Rostros con Imágenes IR. *Ingeniería y Ciencia*, vol. 7, núm. 13, enero-julio, 2011, pp. 111-130, Universidad EAFIT Colombia.
- [40] Shanwen Zhang, Ying-Ke Lei, Modified locally linear discriminant embedding for plant leaf recognition, *Neurocomputing*, Volume 74, Issues 14–15, July 2011, Pages 2284-2290, ISSN 0925-2312.
- [41] Diego Candel Contardo, Algoritmo tipo SMO para la AD-SVM aplicado a clasificación

multicategoría, Universidad Técnica Federico Santa María, Departamento de Informática, Enero 2011.

- [42] H. Muhammad Asraf, M.T. Nooritawati, M.S.B. Shah Rizam, A Comparative Study in Kernel-Based Support Vector Machine of Oil Palm Leaves Nutrient Disease, *Procedia Engineering*, Volume 41, 2012, Pages 1353-1359, ISSN 1877-7058.
- [43] Z. Husin, A.Y.M. Shakaff, A.H.A. Aziz, R.S.M. Farook, M.N. Jaafar, U. Hashim, A. Harun, Embedded portable device for herb leaves recognition using image processing techniques and neural network algorithm, *Computers and Electronics in Agriculture*, Volume 89, November 2012, Pages 18-29, ISSN 0168-1699.
- [44] Roberto Rodríguez Morales, Juan Humberto Sossa Azuela, *Procesamiento y Análisis Digital de Imágenes*, Alfaomega, 2012.
- [45] Rui Hu, John Collomosse, A performance evaluation of gradient field HOG descriptor for sketch based image retrieval, *Computer Vision and Image Understanding*, Volume 117, Issue 7, July 2013, Pages 790-806, ISSN 1077-3142,
- [46] Shanwen Zhang, Yingke Lei, Tianbao Dong, Xiao-Ping Zhang, Label propagation based supervised locality projection analysis for plant leaf classification, *Pattern Recognition*, Volume 46, Issue 7, July 2013, Pages 1891-1897, ISSN 0031-3203.
- [47] Petr Novotný, Tomáš Suk, Leaf recognition of woody species in Central Europe, *Biosystems Engineering*, Volume 115, Issue 4, August 2013, Pages 444-452, ISSN 1537-5110.
- [48] Chunlei Xia, Jang-Myung Lee, Yan Li, Yoo-Han Song, Bu-Keun Chung, Tae-Soo Chon, Plant leaf detection using modified active shape models, *Biosystems Engineering*, Volume 116, Issue 1, September 2013, Pages 23-35, ISSN 1537-5110.
- [49] Ji-xiang Du, Chuan-Min Zhai, Qing-Ping Wang, Recognition of plant leaf image based on fractal dimension features, *Neurocomputing*, Volume 116, 20 September 2013, Pages 150-156, ISSN 0925-2312.

- [50] Chih-Ying Gwo, Chia-Hung Wei, Yue Li, Rotary matching of edge features for leaf recognition, *Computers and Electronics in Agriculture*, Volume 91, February 2013, Pages 124-134, ISSN 0168-1699.
- [51] Gutiérrez Méndez Samuel, "Influencia de Kernels en clasificación de texturas usando máquinas de soporte vectorial. Universidad Autónoma del Estado de México, Agosto 2013.
- [52] Guillaume Cerutti, Laure Tougne, Julien Mille, Antoine Vacavant, Didier Coquin, Understanding leaves in natural images – A model-based approach for tree species identification, *Computer Vision and Image Understanding*, Volume 117, Issue 10, October 2013, Pages 1482-1501, ISSN 1077-3142.
- [53] Shun Li, Jin Wen, A model-based fault detection and diagnostic methodology based on PCA method and wavelet transform, *Energy and Buildings*, Volume 68, Part A, January 2014, Pages 63-71, ISSN 0378-7788.
- [54] Mónica G. Larese, Rafael Namías, Roque M. Craviotto, Miriam R. Arango, Carina Gallo, Pablo M. Granitto, Automatic classification of legumes using leaf vein image features, *Pattern Recognition*, Volume 47, Issue 1, January 2014, Pages 158-168, ISSN 0031-3203.
- [55] Roberto Oberti, Massimo Marchi, Paolo Tirelli, Aldo Calcante, Marcello Iriti, Alberto N. Borghese, Automatic detection of powdery mildew on grapevine leaves by image analysis: Optimal view-angle range to increase the sensitivity, *Computers and Electronics in Agriculture*, Volume 104, June 2014, Pages 1-8, ISSN 0168-1699.
- [56] Mónica G. Larese, Ariel E. Bayá, Roque M. Craviotto, Miriam R. Arango, Carina Gallo, Pablo M. Granitto, Multiscale recognition of legume varieties based on leaf venation images, *Expert Systems with Applications*, Volume 41, Issue 10, August 2014, Pages 4638-4647, ISSN 0957-4174.
- [57] F. Mokhtarian and Riku Suomela, "Curvature Scale Space Based Image Corner Detection", Centre for Vision, Speech, and Signal Processing, Department of Electronic and Electrical Engineering.

- [58] S. Lucey, S. Sridharan and V. Chandran, Adaptive Mouth Segmentation using Chromatic Features, School of Electrical and Electronic Systems Engineering Queensland University of Technology.
- [59] Ioannis Alexiou, Anil A. Bharath, Spatio-Chromatic Opponent Features, BICV Group, Imperial College London.
- [60] Luis Enrique Sucar, Clasificadores Bayesianos: de Datos a Conceptos, Instituto Nacional de Astrofísica, Óptica y Electrónica, Puebla, México.

Apéndice A

Resultados de clasificación del conjunto complejo.

El conjunto complejo de hojas de planta, se divide en 11 subgrupos, cada subgrupo fue analizado con cada una de las combinación de la Tabla 5-4. A continuación se muestran las 14 Tablas de resultados. Cada subconjunto complejo se presenta resultados de tres técnicas de validación que son: precisión de clasificación correcta, F-Measure y Área ROC. Además se presentan los resultados de los seis clasificadores utilizados. Las siglas ‘SC1’ hacen referencia al subconjunto complejo 1, las siglas ‘SC2’ al subconjunto complejo 2 y así sucesivamente hasta el subconjunto complejo 11. Dado que el objetivo es encontrar el clasificador con mejor precisión de clasificación correcta, se marcan en negrita las precisiones más altas que se obtuvieron en cada subconjunto complejo.

Tabla A-1: Resultados del desempeño de clasificación, F-Measure y Área-ROC con cada uno de los clasificadores para los 11 subconjuntos del conjunto complejo. Los resultados se basan en segmentación basada en Otsu y características Cromáticas, Texturales y Geométricas.

		BayesNet	NaviBayes	Perceptrón Multicapa	SMO	J48	Random Forest
SC1	Precisión	91.5842	78.9604	94.3069	95.5446	77.599	85.8911
	F-Measure	0.916	0.8	0.942	0.955	0.775	0.852
	Área ROC	0.996	0.919	0.998	0.992	0.892	0.977
SC2	Precisión	88.3459	77.4436	89.0977	89.4737	81.203	83.4586
	F-Measure	0.882	0.78	0.89	0.894	0.811	0.831
	Área ROC	0.988	0.894	0.981	0.967	0.905	0.976
SC3	Precisión	93.1166	83.174	95.7935	95.9847	77.8203	89.1013
	F-Measure	0.932	0.832	0.958	0.96	0.778	0.889
	Área ROC	0.997	0.93	0.998	0.995	0.891	0.987
SC4	Precisión	91.5606	78.0978	9.2431	95.9812	79.0355	85.5325
	F-Measure	0.916	0.786	0.108	0.96	0.789	0.851
	Área ROC	0.997	0.933	0.639	0.997	0.894	0.982
SC5	Precisión	94.859	83.2504	97.3466	98.01	85.2405	92.0398
	F-Measure	0.949	0.836	0.974	0.98	0.851	0.916
	Área ROC	0.998	0.934	0.999	0.997	0.93	0.993
SC6	Precisión	98.7421	91.195	100	100	94.9686	97.4843
	F-Measure	0.987	0.915	1	1	0.95	0.975
	Área ROC	0.996	0.95	1	1	0.972	1
SC7	Precisión	98.4816	88.5033	97.18	98.0477	93.0586	96.7462
	F-Measure	0.985	0.885	0.971	0.98	0.932	0.967
	Área ROC	0.999	0.943	0.997	0.993	0.963	0.996
SC8	Precisión	96.3731	89.3782	96.8911	97.1503	88.601	95.3368
	F-Measure	0.964	0.895	0.968	0.971	0.886	0.951
	Área ROC	0.998	0.965	0.995	0.995	0.942	0.997
SC9	Precisión	93.9394	75.974	93.9393	94.5887	78.5714	87.4459
	F-Measure	0.939	0.76	0.939	0.946	0.786	0.874
	Área ROC	0.998	0.904	0.998	0.989	0.892	0.984
SC10	Precisión	89.6867	79.6345	42.3028	97.85	83.8112	91.0574
	F-Measure	0.901	0.805	0.332	0.978	0.837	0.907
	Área ROC	0.993	0.928	0.883	0.995	0.91	0.992
SC11	Precisión	86.6001	72.2356	9.0251	94.9707	75.4048	83.4309
	F-Measure	0.87	0.737	0.109	0.949	0.752	0.829
	Área ROC	0.994	0.912	0.641	0.993	0.879	0.974

Tabla A-2: Resultados del desempeño de clasificación, F-Measure y Área-ROC con cada uno de los clasificadores para los 11 subconjuntos del conjunto complejo. Los resultados se basan en segmentación basada en Otsu y características Cromáticas y Texturales

		BayesNet	NaviBayes	Perceptrón Multicapa	SMO	J48	Random Forest
SC1	Precisión	87.2525	76.3614	90.8415	92.203	69.5545	81.1881
	F-Measure	0.872	0.774	0.906	0.921	0.694	0.806
	Área ROC	0.992	0.916	0.98	0.988	0.85	0.973
SC2	Precisión	80.5243	71.5356	82.0224	85.0187	62.1723	80.8989
	F-Measure	0.806	0.726	0.819	0.848	0.63	0.805
	Área ROC	0.97	0.884	0.981	0.942	0.779	0.952
SC3	Precisión	88.3365	79.5411	92.543	94.4651	71.3193	87.3805
	F-Measure	0.884	0.795	0.928	0.944	0.713	0.871
	Área ROC	0.994	0.927	0.998	0.991	0.857	0.984
SC4	Precisión	87.2739	74.4139	28.2652	93.704	68.7877	81.9826
	F-Measure	0.872	0.749	0.257	0.937	0.689	0.813
	Área ROC	0.995	0.93	0.812	0.994	0.889	0.97
SC5	Precisión	91.5423	81.592	93.6981	95.3566	81.0945	86.4013
	F-Measure	0.917	0.818	0.932	0.953	0.813	0.861
	Área ROC	0.996	0.932	0.991	0.993	0.903	0.984
SC6	Precisión	98.7421	88.6792	100	100	94.9686	96.2264
	F-Measure	0.987	0.889	1	1	0.95	0.62
	Área ROC	0.997	0.942	1	1	0.972	0.999
SC7	Precisión	95.6616	85.2495	95.8785	96.3124	85.2495	93.7093
	F-Measure	0.957	0.853	0.959	0.963	0.852	0.937
	Área ROC	0.998	0.931	0.995	0.991	0.923	0.991
SC8	Precisión	95.8549	89.1192	94.8186	97.4093	84.9741	91.1917
	F-Measure	0.958	0.892	0.947	0.974	0.851	0.909
	Área ROC	0.998	0.964	0.997	0.994	0.931	0.991
SC9	Precisión	87.4459	72.2944	88.0952	89.8268	67.7489	78.1385
	F-Measure	0.874	0.724	0.879	0.896	0.679	0.778
	Área ROC	0.991	0.898	0.911	0.976	0.841	0.959
SC10	Precisión	87.0757	78.1984	50.1305	96.7363	79.5692	88.5117
	F-Measure	0.875	0.793	0.499	0.967	0.797	0.88
	Área ROC	0.991	0.927	0.752	0.991	0.893	0.989
SC11	Precisión	80.0551	67.9642	3.1346	92.525	68.5153	80.0551
	F-Measure	0.808	0.697	0.042	0.925	0.686	0.792
	Área ROC	0.988	0.908	0.467	0.991	0.845	0.966

Tabla A-3: Resultados del desempeño de clasificación, F-Measure y Área-ROC con cada uno de los clasificadores para los 11 subconjuntos del conjunto complejo. Los resultados se basan en segmentación basada en Otsu y características Cromáticas y Geométricas.

		BayesNet	NaviBayes	Perceptrón Multicapa	SMO	J48	Random Forest
SC1	Precisión	92.9032	80.0922	95.2073	97.1429	78.894	87.4654
	F-Measure	0.929	0.808	0.952	0.971	0.79	0.872
	Área ROC	0.997	0.932	0.998	0.996	0.9	0.985
SC2	Precisión	86.0947	74.2604	89.6449	90.5325	77.2189	86.9822
	F-Measure	0.86	0.752	0.895	0.905	0.773	0.87
	Área ROC	0.986	0.903	0.989	0.972	0.883	0.981
SC3	Precisión	92.4581	82.1229	96.229	97.905	77.2346	90.2235
	F-Measure	0.925	0.822	0.962	0.979	0.771	0.901
	Área ROC	0.997	0.938	0.995	0.997	0.886	0.99
SC4	Precisión	91.1427	79.6696	30.335	97.1088	83.2951	87.9302
	F-Measure	0.913	0.802	0.322	0.971	0.833	0.875
	Área ROC	0.997	0.943	0.411	0.998	0.922	0.984
SC5	Precisión	96.3211	86.1761	96.8784	97.9933	88.7402	92.1962
	F-Measure	0.964	0.865	0.969	0.98	0.887	0.918
	Área ROC	0.999	0.952	0.999	0.997	0.945	0.995
SC6	Precisión	98.7421	91.195	99.371	100	94.9686	98.1132
	F-Measure	0.987	0.915	0.994	1	0.95	0.981
	Área ROC	1	0.948	0.998	1	0.972	0.998
SC7	Precisión	98.5591	90.3458	97.9827	98.7032	95.6772	98.2709
	F-Measure	0.986	0.903	0.979	0.987	0.957	0.983
	Área ROC	0.998	0.946	0.999	0.997	0.979	0.999
SC8	Precisión	97.8131	89.8608	98.8071	99.006	93.2406	95.6262
	F-Measure	0.978	0.9	0.989	0.99	0.933	0.956
	Área ROC	0.999	0.962	0.999	0.998	0.971	0.997
SC9	Precisión	94.6809	74.6201	93.465	96.0486	83.5466	87.69
	F-Measure	0.947	0.743	0.934	0.96	0.837	0.876
	Área ROC	0.998	0.909	0.995	0.992	0.928	0.963
SC10	Precisión	89.7065	81.7156	46.0496	97.3815	86.6817	93.0474
	F-Measure	0.902	0.827	0.441	0.937	0.866	0.928
	Área ROC	0.992	0.937	0.879	0.993	0.919	0.994
SC11	Precisión	83.8643	68.7128	11.0593	94.5869	76.7159	83.942
	F-Measure	0.846	0.7	0.112	0.946	0.767	0.837
	Área ROC	0.99	0.909	0.211	0.994	0.887	0.978

Tabla A-4: Resultados del desempeño de clasificación, F-Measure y Área-ROC con cada uno de los clasificadores para los 11 subconjuntos del conjunto complejo. Los resultados se basan en segmentación basada en Otsu y características Cromáticas,

		BayesNet	NaviBayes	Perceptrón Multicapa	SMO	J48	Random Forest
SC1	Precisión	89.4301	77.3897	15.3493	93.75	70.9559	84.1912
	F-Measure	0.894	0.782	0.151	0.937	0.709	0.839
	Área ROC	0.994	0.93	0.642	0.992	0.856	0.978
SC2	Precisión	81.5789	68.7135	33.3333	84.5029	69.5905	52.0877
	F-Measure	0.814	0.698	0.309	0.845	0.695	0.849
	Área ROC	0.968	0.88	0.675	0.954	0.829	0.971
SC3	Precisión	88.3171	80.1113	16.968	95.1321	69.541	87.7872
	F-Measure	0.883	0.802	0.116	0.951	0.696	0.866
	Área ROC	0.994	0.938	0.633	0.992	0.847	0.98
SC4	Precisión	86.2557	76.0731	31.8721	94.5662	74.4292	83.6986
	F-Measure	0.863	0.766	0.294	0.945	0.745	0.83
	Área ROC	0.995	0.939	0.732	0.994	0.874	0.98
SC5	Precisión	93.1111	85.2222	94.4444	95.6667	81.1111	88.2222
	F-Measure	0.932	0.855	0.944	0.957	0.811	0.879
	Área ROC	0.997	0.952	0.998	0.994	0.904	0.989
SC6	Precisión	98.7421	88.0503	100	100	94.9686	95.5975
	F-Measure	0.987	0.882	1	1	0.95	0.956
	Área ROC	0.997	0.936	1	1	0.972	0.997
SC7	Precisión	97.1347	85.6734	95.8452	96.7049	87.2493	93.9828
	F-Measure	0.971	0.857	0.959	0.967	0.871	0.94
	Área ROC	0.997	0.931	0.998	0.989	0.925	0.995
SC8	Precisión	95.858	87.3767	95.4635	98.2249	83.0375	92.5049
	F-Measure	0.959	0.876	0.954	0.982	0.83	0.925
	Área ROC	0.997	0.958	0.994	0.996	0.914	0.99
SC9	Precisión	87.6133	71.997	90.3323	91.6918	67.9758	78.0967
	F-Measure	0.877	0.709	0.902	0.915	0.679	0.777
	Área ROC	0.989	0.906	0.945	0.983	0.84	0.962
SC10	Precisión	86.21	78.6841	45.0202	96.0342	82.5146	90.0406
	F-Measure	0.868	0.798	0.379	0.96	0.825	0.896
	Área ROC	0.989	0.931	0.881	0.99	0.902	0.988
SC11	Precisión	77.5168	63.0872	11.5126	92.0496	67.9143	74.8849
	F-Measure	0.786	0.645	0.115	0.92	0.678	0.783
	Área ROC	0.982	0.904	0.345	0.991	0.644	0.968

Tabla A-5: Resultados del desempeño de clasificación, F-Measure y Área-ROC con cada uno de los clasificadores para los 11 subconjuntos del conjunto complejo. Los resultados se basan en segmentación basada en Otsu y características Texturales y Geométricas.

		BayesNet	NaviBayes	Perceptrón Multicapa	SMO	J48	Random Forest
SC1	Precisión	82.9208	80.8168	92.0792	90.2228	75.495	86.1386
	F-Measure	0.83	0.811	0.92	0.901	0.756	0.859
	Área ROC	0.988	0.974	0.992	0.988	0.889	0.988
SC2	Precisión	89.8496	83.4586	92.8571	92.1053	78.1955	86.4662
	F-Measure	0.9	0.835	0.928	0.92	0.781	0.863
	Área ROC	0.989	0.974	0.994	0.981	0.89	0.988
SC3	Precisión	84.3212	82.0268	91.7782	88.5277	77.0554	85.4685
	F-Measure	0.843	0.819	0.915	0.882	0.769	0.853
	Área ROC	0.987	0.979	0.993	0.983	0.886	0.986
SC4	Precisión	83.858	81.8486	92.4983	88.1447	80.71	85.2646
	F-Measure	0.839	0.819	0.925	0.88	0.806	0.851
	Área ROC	0.993	0.985	0.995	0.989	0.915	0.983
SC5	Precisión	89.5222	87.3964	93.2007	92.3715	81.592	91.2106
	F-Measure	0.898	0.876	0.931	0.923	0.816	0.912
	Área ROC	0.996	0.984	0.994	0.992	0.917	0.993
SC6	Precisión	96.2264	91.195	99.3711	99.3711	96.2264	97.4843
	F-Measure	0.962	0.91	0.994	0.994	0.962	0.975
	Área ROC	0.996	0.98	0.999	0.997	0.978	0.993
SC7	Precisión	98.9293	96.788	98.2869	96.788	92.7195	97.8587
	F-Measure	0.989	0.968	0.983	0.968	0.927	0.978
	Área ROC	1	0.997	1	0.992	0.967	0.999
SC8	Precisión	96.114	94.3005	98.1865	97.9275	88.0829	96.3731
	F-Measure	0.961	0.943	0.982	0.975	0.881	0.963
	Área ROC	0.999	0.998	0.999	0.996	0.953	0.998
SC9	Precisión	85.2814	84.632	91.9913	89.3939	77.7056	85.9307
	F-Measure	0.852	0.847	0.92	0.893	0.775	0.858
	Área ROC	0.99	0.981	0.988	0.984	0.906	0.984
SC10	Precisión	89.0992	86.423	95.235	91.5796	85.4439	91.1227
	F-Measure	0.894	0.87	0.952	0.911	0.854	0.909
	Área ROC	0.994	0.987	0.997	0.976	0.956	0.993
SC11	Precisión	84.0495	78.0681	92.9873	88.1403	78.5693	89.9371
	F-Measure	0.84	0.779	0.929	0.878	0.787	0.867
	Área ROC	0.992	0.979	0.995	0.986	0.903	0.987

Tabla A-6: Resultados del desempeño de clasificación, F-Measure y Área-ROC con cada uno de los clasificadores para los 11 subconjuntos del conjunto complejo. Los resultados se basan en segmentación basada en Otsu y características Texturales

		BayesNet	NaviBayes	Perceptrón Multicapa	SMO	J48	Random Forest
SC1	Precisión	36.2624	31.8069	69.5545	48.3911	51.7327	57.9208
	F-Measure	0.348	0.293	0.691	0.467	0.513	0.573
	Área ROC	0.85	0.832	0.931	0.888	0.185	0.896
SC2	Precisión	50.5618	39.7004	77.5281	55.4307	62.5468	71.5356
	F-Measure	0.492	0.371	0.773	0.533	0.621	0.699
	Área ROC	0.822	0.796	0.95	0.844	0.815	0.916
SC3	Precisión	47.2275	36.1377	80.6883	60.2294	61.7591	68.6424
	F-Measure	0.457	0.332	0.804	0.563	0.617	0.684
	Área ROC	0.881	0.867	0.958	0.912	0.807	0.928
SC4	Precisión	42.0348	35.2075	66.5997	44.5114	53.5475	62.1151
	F-Measure	0.388	0.317	0.665	0.368	0.534	0.609
	Área ROC	0.878	0.853	0.933	0.881	0.789	0.911
SC5	Precisión	39.6352	28.1924	71.3101	55.058	55.7214	65.6716
	F-Measure	0.353	0.251	0.706	0.512	0.557	0.648
	Área ROC	0.867	0.799	0.944	0.856	0.795	0.916
SC6	Precisión	83.6478	75.4717	94.9686	89.3082	89.9371	93.7107
	F-Measure	0.837	0.752	0.949	0.891	0.898	0.937
	Área ROC	0.959	0.903	0.991	0.937	0.935	0.99
SC7	Precisión	61.4561	49.8929	84.3683	65.0964	75.803	79.015
	F-Measure	0.63	0.529	0.844	0.604	0.757	0.782
	Área ROC	0.87	0.837	0.972	0.88	0.87	0.949
SC8	Precisión	75.544	57.2539	85.2332	72.2798	75.9067	77.9793
	F-Measure	0.956	0.575	0.851	0.996	0.761	0.772
	Área ROC	0.923	0.912	0.966	0.92	0.885	0.958
SC9	Precisión	40.9091	37.4059	64.7186	51.2987	48.0519	55.5887
	F-Measure	0.399	0.353	0.645	0.484	0.478	0.553
	Área ROC	0.821	0.808	0.921	0.861	0.757	0.861
SC10	Precisión	44.2559	39.0992	77.5457	65.5352	67.624	73.6945
	F-Measure	0.471	0.423	0.776	0.587	0.66	0.714
	Área ROC	0.892	0.872	0.943	0.877	0.833	0.929
SC11	Precisión	33.6542	21.0382	61.3957	41.8701	48.8828	55.483
	F-Measure	0.321	0.188	0.603	0.327	0.483	0.534
	Área ROC	0.855	0.789	0.914	0.845	0.755	0.879

Tabla A-7: Resultados del desempeño de clasificación, F-Measure y Área-ROC con cada uno de los clasificadores para los 11 subconjuntos del conjunto complejo. Los resultados se basan en segmentación basada en Otsu y características Geométricas.

		BayesNet	NaviBayes	Perceptrón Multicapa	SMO	J48	Random Forest
SC1	Precisión	79.5037	74.6324	91.3003	84.0074	78.8603	89.1544
	F-Measure	0.794	0.749	0.912	0.839	0.787	0.891
	Área ROC	0.984	0.97	0.992	0.981	0.919	0.987
SC2	Precisión	83.8235	78.5294	87.3529	81.4706	83.2353	89.4118
	F-Measure	0.846	0.79	0.873	0.815	0.83	0.895
	Área ROC	0.985	0.963	0.977	0.953	0.919	0.991
SC3	Precisión	78.3634	72.9542	87.9334	74.3412	80.1664	90.5687
	F-Measure	0.782	0.728	0.876	0.736	0.8	0.905
	Área ROC	0.979	0.971	0.988	0.963	0.91	0.99
SC4	Precisión	75.0228	71.7867	90.7931	82.9535	81.7229	89.1067
	F-Measure	0.751	0.719	0.908	0.827	0.817	0.89
	Área ROC	0.979	0.97	0.99	0.98	0.924	0.992
SC5	Precisión	84.8889	84.3333	93.4444	89.1111	83.6667	92.4444
	F-Measure	0.853	0.849	0.934	0.888	0.837	0.924
	Área ROC	0.99	0.982	0.995	0.982	0.929	0.995
SC6	Precisión	92.4528	90.566	99.3711	99.3711	94.9686	97.4843
	F-Measure	0.925	0.904	0.994	0.994	0.95	0.975
	Área ROC	0.984	0.967	0.998	0.996	0.974	0.994
SC7	Precisión	98.3003	96.3173	98.8669	97.4504	96.7422	99.0085
	F-Measure	0.983	0.963	0.989	0.974	0.967	0.99
	Área ROC	1	0.996	1	0.993	0.989	1
SC8	Precisión	94.8718	93.6884	97.8304	95.4635	91.9132	96.8442
	F-Measure	0.948	0.937	0.978	0.954	0.919	0.968
	Área ROC	0.998	0.995	0.999	0.994	0.966	0.999
SC9	Precisión	84.4411	80.2115	92.7492	85.9517	83.9879	91.3897
	F-Measure	0.842	0.802	0.927	0.856	0.841	0.913
	Área ROC	0.987	0.975	0.99	0.982	0.941	0.99
SC10	Precisión	79.5854	74.9887	93.4205	83.6863	85.8945	92.1136
	F-Measure	0.806	0.768	0.938	0.814	0.859	0.919
	Área ROC	0.984	0.974	0.993	0.942	0.932	0.991
SC11	Precisión	74.2916	69.4487	87.8413	82.0453	79.6239	89.052
	F-Measure	0.744	0.689	0.876	0.815	0.796	0.89
	Área ROC	0.981	0.966	0.984	0.978	0.915	0.99

Tabla A-8: Resultados del desempeño de clasificación, F-Measure y Área-ROC con cada uno de los clasificadores para los 11 subconjuntos del conjunto complejo. Los resultados se basan en segmentación basada en PCA y características Cromáticas, Texturales y Geométricas.

		BayesNet	NaviBayes	Perceptrón Multicapa	SMO	J48	Random Forest
SC1	Precisión	91.909	80.1517	94.5638	96.2073	78.7611	85.5879
	F-Measure	0.919	0.809	0.945	0.962	0.788	0.853
	Área ROC	0.996	0.919	0.994	0.994	0.898	0.976
SC2	Precisión	87.7395	76.6284	88.1226	90.0383	74.3295	85.8238
	F-Measure	0.875	0.779	0.882	0.9	0.746	0.857
	Área ROC	0.987	0.906	0.991	0.97	0.859	0.983
SC3	Precisión	93.797	81.203	95.4887	96.9925	81.7669	89.0977
	F-Measure	0.939	0.812	0.954	0.97	0.815	0.889
	Área ROC	0.997	0.926	0.994	0.944	0.911	0.987
SC4	Precisión	90.9401	78.406	28.2016	96.2534	78.1335	84.4687
	F-Measure	0.909	0.789	0.251	0.962	0.781	0.84
	Área ROC	0.997	0.932	0.799	0.997	0.896	0.979
SC5	Precisión	95.7841	82.7993	95.7841	98.145	87.0152	91.7369
	F-Measure	0.958	0.831	0.958	0.981	0.87	0.913
	Área ROC	0.999	0.935	0.998	0.997	0.937	0.994
SC6	Precisión	98.7421	91.195	100	100	95.5975	97.4743
	F-Measure	0.987	0.915	1	1	0.956	0.974
	Área ROC	0.996	0.949	1	1	0.974	0.999
SC7	Precisión	98.5356	89.3305	97.0711	98.954	93.3054	97.4895
	F-Measure	0.985	0.894	0.971	0.989	0.933	0.976
	Área ROC	0.999	0.948	0.996	0.997	0.963	0.998
SC8	Precisión	96.3542	89.8438	97.3958	98.1771	92.1875	95.0521
	F-Measure	0.963	0.9	0.937	0.981	0.92	0.948
	Área ROC	0.998	0.958	0.994	0.997	0.958	0.996
SC9	Precisión	95.1965	75.3275	94.1965	93.8865	82.3144	87.9913
	F-Measure	0.952	0.753	0.941	0.938	0.826	0.878
	Área ROC	0.998	0.91	0.996	0.88	0.917	0.982
SC10	Precisión	90.0273	79.3033	45.9016	98.2923	84.0161	90.4372
	F-Measure	0.904	0.801	0.393	0.983	0.939	0.9
	Área ROC	0.994	0.923	0.895	0.996	0.902	0.991
SC11	Precisión	86.4827	71.2668	9.1648	93.8075	75.7254	83.581
	F-Measure	0.869	0.727	0.11	0.938	0.757	0.83
	Área ROC	0.993	0.906	0.648	0.994	0.879	0.978

Tabla A-9: Resultados del desempeño de clasificación, F-Measure y Área-ROC con cada uno de los clasificadores para los 11 subconjuntos del conjunto complejo. Los resultados se basan en segmentación basada en PCA y características Cromáticas y Texturales

		BayesNet	NaviBayes	Perceptrón Multicapa	SMO	J48	Random Forest
SC1	Precisión	87.3578	76.7383	90.7711	93.426	69.2794	80.2781
	F-Measure	0.873	0.777	0.906	0.934	0.692	0.789
	Área ROC	0.991	0.916	0.989	0.99	0.852	0.972
SC2	Precisión	82.0611	72.1374	84.3511	84.7328	67.9389	79.3893
	F-Measure	0.821	0.733	0.895	0.846	0.68	0.793
	Área ROC	0.969	0.887	0.985	0.949	0.816	0.957
SC3	Precisión	90.4135	78.7594	93.7969	95.8647	71.0526	87.218
	F-Measure	0.905	0.788	0.939	0.959	0.708	0.87
	Área ROC	0.994	0.923	0.997	0.991	0.856	0.978
SC4	Precisión	87.2616	75.0681	30.7901	94.0054	68.8011	80.7221
	F-Measure	0.873	0.754	0.299	0.94	0.686	0.799
	Área ROC	0.994	0.93	0.541	0.994	0.845	0.973
SC5	Precisión	92.2428	80.6071	93.5919	95.9528	81.4503	87.0152
	F-Measure	0.923	0.809	0.36	0.959	0.816	0.867
	Área ROC	0.996	0.934	0.995	0.993	0.909	0.985
SC6	Precisión	98.7421	88.6792	99.371	100	95.5975	97.4843
	F-Measure	0.987	0.889	0.994	1	0.956	0.975
	Área ROC	0.997	0.942	0.998	1	0.974	0.999
SC7	Precisión	96.8619	86.4017	96.0251	97.2803	87.2385	94.1423
	F-Measure	0.969	0.864	0.952	0.972	0.873	0.941
	Área ROC	0.999	0.938	0.991	0.994	0.922	0.996
SC8	Precisión	95.0521	88.8021	96.0937	97.1354	86.7188	91.6667
	F-Measure	0.95	0.889	0.972	0.971	0.867	0.911
	Área ROC	0.998	0.957	0.996	0.995	0.928	0.993
SC9	Precisión	87.7729	71.6157	89.7379	91.4847	67.6856	81.0044
	F-Measure	0.877	0.72	0.901	0.913	0.674	0.808
	Área ROC	0.992	0.903	0.991	0.979	0.833	0.97
SC10	Precisión	87.7486	77.2541	46.8579	96.9262	81.0792	86.6803
	F-Measure	0.872	0.783	0.454	0.968	0.81	0.86
	Área ROC	0.991	0.921	0.882	0.992	0.898	0.984
SC11	Precisión	80.5379	66.4897	8.351	92.569	69.1437	78.0962
	F-Measure	0.813	0.682	0.131	0.925	0.691	0.77
	Área ROC	0.987	0.901	0.251	0.991	0.841	0.968

Tabla A-10: Resultados del desempeño de clasificación, F-Measure y Área-ROC con cada uno de los clasificadores para los 11 subconjuntos del conjunto complejo. Los resultados se basan en segmentación basada en PCA y características Cromáticas y Geométricas.

		BayesNet	NaviBayes	Perceptrón Multicapa	SMO	J48	Random Forest
SC1	Precisión	92.3713	81.8015	96.2316	97.2426	80.9743	86.489
	F-Measure	0.923	0.823	0.62	0.972	0.809	0.862
	Área ROC	0.997	0.936	0.999	0.996	0.91	0.988
SC2	Precisión	87.0588	76.1765	90	90.5882	78.5294	87.6471
	F-Measure	0.869	0.768	0.9	0.905	0.784	0.871
	Área ROC	0.986	0.893	0.972	0.967	0.878	0.983
SC3	Precisión	92.5104	82.5243	96.3938	96.9487	83.6338	89.8752
	F-Measure	0.926	0.827	0.961	0.969	0.837	0.897
	Área ROC	0.997	0.937	0.999	0.996	0.917	0.989
SC4	Precisión	91.4234	80.2464	33.7135	97.2628	81.0675	88.458
	F-Measure	0.915	0.807	0.309	0.972	0.809	0.882
	Área ROC	0.997	0.946	0.692	0.998	0.908	0.987
SC5	Precisión	96.3333	83.3333	96.6666	98.1111	87.6667	92.6667
	F-Measure	0.964	0.867	0.966	0.981	0.877	0.923
	Área ROC	0.999	0.953	0.999	0.998	0.945	0.992
SC6	Precisión	98.7421	91.195	100	100	95.5975	97.4843
	F-Measure	0.987	0.915	1	1	0.956	0.975
	Área ROC	0.998	0.948	1	1	0.974	0.998
SC7	Precisión	98.8539	88.8252	96.991	98.5663	95.5587	97.5645
	F-Measure	0.989	0.888	0.969	0.986	0.956	0.975
	Área ROC	0.999	0.942	0.996	0.995	0.977	0.999
SC8	Precisión	96.8442	88.9546	97.2386	99.211	92.7022	95.858
	F-Measure	0.68	0.892	0.972	0.992	0.927	0.958
	Área ROC	0.96	0.964	0.997	0.999	0.965	0.997
SC9	Precisión	94.4109	76.435	93.6555	96.0725	82.3263	87.4622
	F-Measure	0.944	0.764	0.937	0.961	0.825	0.872
	Área ROC	0.998	0.925	0.997	0.993	0.919	0.983
SC10	Precisión	89.7706	80.5768	50.7886	97.2961	87.607	92.6093
	F-Measure	0.903	0.816	0.498	0.973	0.876	0.923
	Área ROC	0.992	0.934	0.825	0.993	0.925	0.993
SC11	Precisión	83.8152	68.6113	14.5069	94.889	76.8715	83.8152
	F-Measure	0.845	0.7	0.145	0.949	0.768	0.836
	Área ROC	0.991	0.91	0.642	0.995	0.888	0.979

Tabla A-11: Resultados del desempeño de clasificación, F-Measure y Área-ROC con cada uno de los clasificadores para los 11 subconjuntos del conjunto complejo. Los resultados se basan en segmentación basada en PCA y características Cromáticas

		BayesNet	NaviBayes	Perceptrón Multicapa	SMO	J48	Random Forest
SC1	Precisión	88.7052	77.4105	72.4105	94.2149	71.9927	84.2157
	F-Measure	0.886	0.782	0.733	0.942	0.719	0.836
	Área ROC	0.994	0.932	0.888	0.991	0.867	0.937
SC2	Precisión	81.5789	63.2982	84.5029	84.5029	67.2515	82.4561
	F-Measure	0.815	0.705	0.845	0.845	0.669	0.823
	Área ROC	0.969	0.883	0.954	0.954	0.811	0.968
SC3	Precisión	88.2271	78.5319	93.7673	94.7368	71.0526	85.0416
	F-Measure	0.883	0.787	0.936	0.947	0.712	0.849
	Área ROC	0.993	0.931	0.999	0.992	0.861	0.979
SC4	Precisión	86.2557	76.0731	74.4292	94.5662	74.4292	83.6986
	F-Measure	0.863	0.766	0.745	0.945	0.745	0.83
	Área ROC	0.995	0.939	0.874	0.994	0.874	0.98
SC5	Precisión	93.6777	84.5727	94.1176	95.5605	81.465	87.7913
	F-Measure	0.937	0.85	0.941	0.955	0.817	0.873
	Área ROC	0.997	0.949	0.992	0.994	0.908	0.99
SC6	Precisión	99.3711	88.0503	100	100	95.5775	98.1132
	F-Measure	0.994	0.882	1	1	0.956	0.981
	Área ROC	0.997	0.938	1	1	0.974	0.999
SC7	Precisión	96.5616	85.6734	95.8452	96.7049	87.5358	93.2665
	F-Measure	0.966	0.857	0.958	0.967	0.875	0.932
	Área ROC	0.997	0.93	0.988	0.989	0.925	0.993
SC8	Precisión	95.858	87.9684	95.8579	97.8304	83.8264	92.8979
	F-Measure	0.959	0.882	0.958	0.978	0.837	0.928
	Área ROC	0.996	0.962	0.991	0.995	0.922	0.994
SC9	Precisión	88.3861	71.9457	90.0452	90.6486	67.27	80.2413
	F-Measure	0.884	0.718	0.904	0.904	0.67	0.801
	Área ROC	0.99	0.906	0.994	0.983	0.838	0.965
SC10	Precisión	86.3001	78.639	48.5353	95.854	82.8301	89.4096
	F-Measure	0.869	0.798	0.481	0.958	0.827	0.889
	Área ROC	0.989	0.931	0.818	0.989	0.901	0.989
SC11	Precisión	77.62	62.984	11.7965	92.3335	68.5854	78.9365
	F-Measure	0.787	0.644	0.117	0.923	0.685	0.784
	Área ROC	0.982	0.904	0.546	0.991	0.844	0.97

Tabla A-12: Resultados del desempeño de clasificación, F-Measure y Área-ROC con cada uno de los clasificadores para los 11 subconjuntos del conjunto complejo. Los resultados se basan en segmentación basada en PCA y características Texturales y Geométricas.

		BayesNet	NaviBayes	Perceptrón Multicapa	SMO	J48	Random Forest
SC1	Precisión	86.4728	82.5537	93.8053	91.2769	77.6213	88.1163
	F-Measure	0.866	0.826	0.938	0.913	0.776	0.88
	Área ROC	0.992	0.983	0.997	0.992	0.906	0.99
SC2	Precisión	88.1226	81.1418	93.8697	90.4215	78.9272	89.6552
	F-Measure	0.882	0.831	0.938	0.903	0.79	0.896
	Área ROC	0.991	0.976	0.994	0.975	0.886	0.991
SC3	Precisión	85.1504	80.6391	93.0451	89.2857	80.2632	85.7143
	F-Measure	0.85	0.81	0.929	0.888	0.801	0.851
	Área ROC	0.989	0.978	0.996	0.988	0.903	0.985
SC4	Precisión	84.3324	82.2207	92.3706	88.7206	79.2916	84.8774
	F-Measure	0.843	0.923	0.923	0.887	0.792	0.846
	Área ROC	0.992	0.983	0.994	0.988	0.907	0.984
SC5	Precisión	90.2192	88.8702	94.6037	93.086	83.8111	88.027
	F-Measure	0.904	0.891	0.946	0.931	0.839	0.879
	Área ROC	0.96	0.989	0.998	0.992	0.923	0.994
SC6	Precisión	96.2264	91.8239	99.3711	99.3711	95.5975	96.2264
	F-Measure	0.962	0.918	0.994	0.994	0.953	0.962
	Área ROC	0.995	0.982	0.999	0.997	0.969	0.992
SC7	Precisión	99.3802	97.5207	98.5337	98.7603	95.0413	98.1405
	F-Measure	0.994	0.975	0.985	0.987	0.95	0.981
	Área ROC	1	0.997	1	0.997	0.982	0.999
SC8	Precisión	96.6146	97.1354	98.6979	98.9583	93.4896	95.3125
	F-Measure	0.966	0.971	0.987	0.989	0.935	0.952
	Área ROC	0.999	0.997	0.999	0.997	0.969	0.997
SC9	Precisión	87.5546	87.1179	92.1397	90.1747	80.786	87.1179
	F-Measure	0.876	0.871	0.919	0.901	0.808	0.872
	Área ROC	0.993	0.978	0.986	0.985	0.913	0.986
SC10	Precisión	90.5055	87.4317	95.6284	91.9399	84.4262	91.3934
	F-Measure	0.908	0.88	0.959	0.916	0.844	0.912
	Área ROC	0.995	0.986	0.998	0.979	0.924	0.992
SC11	Precisión	84.0395	78.6017	92.6554	87.8178	79.3432	87.2881
	F-Measure	0.838	0.784	0.925	0.876	0.794	0.87
	Área ROC	0.992	0.979	0.995	0.987	0.906	0.987

Tabla A-13: Resultados del desempeño de clasificación, F-Measure y Área-ROC con cada uno de los clasificadores para los 11 subconjuntos del conjunto complejo. Los resultados se basan en segmentación basada en PCA y características Texturales

		BayesNet	NaviBayes	Perceptrón Multicapa	SMO	J48	Random Forest
SC1	Precisión	36.2832	31.0999	67.7851	46.9027	49.9368	59.0392
	F-Measure	0.342	0.27	0.695	0.444	0.498	0.586
	Área ROC	0.851	0.831	0.926	0.889	0.768	0.914
SC2	Precisión	48.8733	41.6031	78.2443	53.0534	67.1756	64.5038
	F-Measure	0.472	0.416	0.78	0.505	0.662	0.635
	Área ROC	0.818	0.8	0.942	0.836	0.823	0.902
SC3	Precisión	47.1805	36.4662	77.0677	57.5188	62.218	70.8647
	F-Measure	0.469	0.349	0.765	0.54	0.62	0.707
	Área ROC	0.899	0.88	0.954	0.919	0.834	0.936
SC4	Precisión	41.7291	33.8325	67.3247	45.8816	54.9653	62.083
	F-Measure	0.383	0.309	0.679	0.383	0.546	0.607
	Área ROC	0.881	0.853	0.942	0.885	0.792	0.916
SC5	Precisión	40.3035	29.8482	75.0422	57.8415	58.6847	63.2368
	F-Measure	0.371	0.274	0.742	0.533	0.582	0.627
	Área ROC	0.87	0.81	0.945	0.868	0.802	0.911
SC6	Precisión	85.5346	76.1006	93.7107	89.3082	91.8239	91.195
	F-Measure	0.856	0.758	0.936	0.891	0.917	0.911
	Área ROC	0.959	0.905	0.995	0.939	0.946	0.978
SC7	Precisión	64.2562	51.6529	87.3967	65.0826	75.2066	79.3388
	F-Measure	0.656	0.544	0.871	0.607	0.748	0.788
	Área ROC	0.875	0.845	0.975	0.877	0.861	0.985
SC8	Precisión	66.4063	60.4167	87.2396	71.875	73.4375	78.6458
	F-Measure	0.667	0.612	0.872	0.688	0.734	0.781
	Área ROC	0.935	0.926	0.975	0.916	0.878	0.959
SC9	Precisión	40.393	38.209	68.7763	48.69	51.7467	55.8952
	F-Measure	0.387	0.352	0.681	0.47	0.517	0.554
	Área ROC	0.83	0.811	0.924	0.853	0.772	0.876
SC10	Precisión	39.5492	47.3361	78.8934	66.8716	65.9836	73.9071
	F-Measure	0.425	0.501	0.779	0.61	0.651	0.77
	Área ROC	0.873	0.899	0.945	0.882	0.823	0.932
SC11	Precisión	33.7218	21.6102	63.4181	42.161	48.7994	54.1314
	F-Measure	0.312	0.193	0.626	0.329	0.483	0.522
	Área ROC	0.856	0.794	0.917	0.851	0.753	0.883

Tabla A-14: Resultados del desempeño de clasificación, F-Measure y Área-ROC con cada uno de los clasificadores para los 11 subconjuntos del conjunto complejo. Los resultados se basan en segmentación basada en PCA y características Geométricas.

		BayesNet	NaviBayes	Perceptrón Multicapa	SMO	J48	Random Forest
SC1	Precisión	81.8934	77.3897	93.5662	87.4081	82.261	92.6471
	F-Measure	0.82	0.778	0.935	0.874	0.822	0.926
	Área ROC	0.988	0.975	0.994	0.986	0.927	0.994
SC2	Precisión	85.5882	77.3529	90	82.9412	81.4706	89.7059
	F-Measure	0.862	0.779	0.899	0.83	0.814	0.897
	Área ROC	0.968	0.96	0.985	0.953	0.912	0.994
SC3	Precisión	78.0886	74.7573	89.3204	78.9182	80.8599	88.7656
	F-Measure	0.779	0.75	0.892	0.782	0.808	0.885
	Área ROC	0.98	0.972	0.987	0.968	0.917	0.987
SC4	Precisión	74.7037	72.2881	90.7475	83.4093	81.1304	89.7903
	F-Measure	0.748	0.725	0.907	0.832	0.811	0.897
	Área ROC	0.979	0.969	0.989	0.979	0.922	0.99
SC5	Precisión	85.4444	85.7778	95.1111	90	86	91.2222
	F-Measure	0.858	0.863	0.951	0.897	0.861	0.911
	Área ROC	0.99	0.986	0.998	0.985	0.94	0.992
SC6	Precisión	93.0818	91.195	99.3711	98.7421	94.3396	96.8553
	F-Measure	0.932	0.911	0.994	0.987	0.943	0.968
	Área ROC	0.985	0.968	0.998	0.994	0.965	0.994
SC7	Precisión	97.8754	97.0255	98.7252	97.7337	97.5921	99.0085
	F-Measure	0.979	0.97	0.987	0.977	0.976	0.99
	Área ROC	1	0.995	0.999	0.994	0.991	1
SC8	Precisión	95.2663	94.4773	98.8166	97.0414	90.5325	97.2387
	F-Measure	0.953	0.945	0.988	0.97	0.905	0.972
	Área ROC	0.998	0.994	1	0.996	0.96	0.997
SC9	Precisión	84.5921	80.6647	93.8076	94.4109	82.6284	92.5982
	F-Measure	0.844	0.807	0.938	0.944	0.827	0.926
	Área ROC	0.987	0.973	0.993	0.989	0.938	0.994
SC10	Precisión	80.2163	76.521	93.105	84.1821	86.0748	92.2488
	F-Measure	0.812	0.784	0.931	0.82	0.859	0.92
	Área ROC	0.985	0.976	0.991	0.945	0.926	0.991
SC11	Precisión	74.8326	70.3503	88.3565	81.3241	80.0103	89.1293
	F-Measure	0.75	0.698	0.882	0.807	0.8	0.891
	Área ROC	0.982	0.966	0.983	0.978	0.915	0.991

Apéndice B

Resultados de clasificación con Algoritmo Genético

Las precisiones de clasificación mejoradas con el AG en cada combinación se presentan a continuación. En las columnas con el encabezado ‘TC’ muestran los datos de clasificación utilizando todas las características según sea el caso de la combinación. En las columnas con el encabezado ‘AG’ se muestran los resultados después de aplicar el Algoritmo Genético. Los valores marcados en negrita son los valores que tuvieron un incremento más significativo que se tuvo con el Algoritmo Genético.

Tabla B-1: Comparativa de resultados antes y después de aplicar el Algoritmo Genético en datos con segmentación **Otsu** y características **Cromáticas, Texturales y Geométricas**.

	Bayes Net		Navi Bayes		Perceptron Multicapa		SMO		J48		Random Forest	
	TC	AG	TC	AG	TC	AG	TC	AG	TC	AG	TC	AG
SC1	91.5842	92.3267	78.9604	79.8267	94.3069	94.3069	95.5446	95.9158	77.599	79.3316	85.8911	93.5644
SC2	88.3459	90.6015	77.4436	79.3233	89.0977	89.4736	89.4737	92.1052	81.203	83.0827	83.4586	88.3458
SC3	93.1166	95.0286	83.174	83.7476	95.7935	95.7935	95.9847	96.7495	77.8203	83.3652	89.1013	91.0133
SC4	91.5606	92.1634	78.0978	78.4996	9.2431	31.949	95.9812	95.9812	79.0355	80.2411	85.5325	86.872
SC5	94.859	96.0199	83.2504	86.733	97.3466	97.3466	98.01	98.3416	85.2405	86.5671	92.0398	92.5373
SC6	98.7421	99.371	91.195	96.2264	100	100	100	100	94.9686	96.8553	97.4843	99.371
SC7	98.4816	99.3492	88.5033	90.4553	97.18	97.3969	98.0477	98.9154	93.0586	96.0954	96.7462	97.8308
SC8	96.3731	97.6683	89.3782	92.487	96.8911	97.4093	97.1503	98.4455	88.601	92.2279	95.3368	96.373
SC9	93.9394	95.238	75.974	77.2727	93.9393	94.8051	94.5887	95.0216	78.5714	84.632	87.4459	90.4761
SC10	89.6867	91.1879	79.6345	80.7441	42.3028	48.3028	97.85	97.9765	83.8112	87.5979	91.0574	92.3629
SC11	86.6001	88.1846	72.2356	72.2356	9.0251	10.6097	94.9707	94.9707	75.4048	77.4371	83.4309	84.5332
T	88.8413	89.1197	73.4204	73.4204	6.2325	7.7746	92.3752	93.7245	76.2262	76.2475	81.3665	80.7025

Tabla B-2: Comparativa de resultados antes y después de aplicar el Algoritmo Genético en datos con segmentación **Otsu** y características **Cromáticas y Texturales**

	Bayes Net		Navi Bayes		Perceptron Multicapa		SMO		J48		Random Forest	
	TC	AG	TC	AG	TC	AG	TC	AG	TC	AG	TC	AG
SC1	87.2525	89.9752	76.3614	78.094	90.8415	91.0891	92.203	93.3168	69.5545	71.5346	81.1881	83.0445
SC2	80.5243	85.0187	71.5356	74.5318	82.0224	84.2696	85.0187	86.5168	62.1723	73.7827	80.8989	83.146
SC3	88.3365	91.2045	79.5411	80.8795	92.543	93.8814	94.4651	95.411	71.3193	75.1434	87.3805	87.3805
SC4	87.2739	88.7474	74.4139	75.6195	28.2652	32.7528	93.704	94.1058	68.7877	71.6008	81.9826	82.6523
SC5	91.5423	94.0298	81.592	84.9087	93.6981	93.864	95.3566	96.849	81.0945	84.4112	86.4013	89.8839
SC6	98.7421	99.371	88.6792	92.4528	100	100	100	100	94.9686	96.2264	96.2264	99.371
SC7	95.6616	98.4815	85.2495	87.4186	95.8785	96.3123	96.3124	97.3969	85.2495	91.3232	93.7093	95.4446
SC8	95.8549	97.1502	89.1192	91.9689	94.8186	96.1139	97.4093	98.1865	84.9741	89.6373	91.1917	94.8186
SC9	87.4459	89.8268	72.2944	73.1601	88.0952	88.961	89.8268	91.7748	67.7489	73.1601	78.1385	82.4675
SC10	87.0757	89.0992	78.1984	78.1984	50.1305	51.6971	96.7363	96.9973	79.5692	83.4204	88.5117	89.7519
SC11	80.0551	82.4319	67.9642	67.9642	3.1346	9.9207	92.525	92.525	68.5153	70.7199	80.0551	80.9507
T	81.3062	83.062	68.5653	68.5653	6.6595	6.6595	89.9357	90.6423	64.6467	65.4817	72.334	75.1392

Tabla B-3: Comparativa de resultados antes y después de aplicar el Algoritmo Genético en datos con segmentación **Otsu** y características **Cromáticas y Geométricas**

	Bayes Net		Navi Bayes		Perceptron Multicapa		SMO		J48		Random Forest	
	TC	AG	TC	AG	TC	AG	TC	AG	TC	AG	TC	AG
SC1	92.9032	94.1935	80.0922	80.9216	95.2073	95.8525	97.1429	97.235	78.894	80.9217	87.4654	88.2027
SC2	86.0947	89.6449	74.2604	75.7396	89.6449	90.5325	90.5325	91.7159	77.2189	80.7692	86.9822	89.3491
SC3	92.4581	93.4357	82.1229	82.8212	96.229	96.648	97.905	97.905	77.2346	82.4022	90.2235	91.3407
SC4	91.1427	92.0146	79.6696	79.8072	30.335	35.5667	97.1088	97.1088	83.2951	83.2951	87.9302	89.9495
SC5	96.3211	96.4325	86.1761	87.068	96.8784	96.9899	97.9933	98.3277	88.7402	89.1861	92.1962	93.7569
SC6	98.7421	100	91.195	95.5974	99.371	100	100	100	94.9686	96.8553	98.1132	99.371
SC7	98.5591	99.1354	90.3458	90.634	97.9827	97.9827	98.7032	98.8472	95.6772	96.9741	98.2709	98.7032
SC8	97.8131	98.6083	89.8608	91.0536	98.8071	99.0059	99.006	99.6023	93.2406	94.6322	95.6262	98.4155
SC9	94.6809	95.8966	74.6201	79.3313	93.465	94.5288	96.0486	96.6565	83.5466	84.6504	87.69	90.8814
SC10	89.7065	90.9706	81.7156	82.3025	46.0496	55.3498	97.3815	97.562	86.6817	89.7065	93.0474	93.4989
SC11	83.8643	85.6255	68.7128	68.7128	11.0593	11.2924	94.5869	94.5869	76.7159	78.1144	83.942	85.1075
T	87.8337	89.0521	73.7861	73.7861	4.7482	8.5826	91.2381	94.2841	77.065	77.6742	80.8457	83.4976

Tabla B-4: Comparativa de resultados antes y después de aplicar el Algoritmo Genético en datos con segmentación **Otsu** y características **Cromáticas**,

	Bayes Net		Navi Bayes		Perceptron Multicapa		SMO		J48		Random Forest	
	TC	AG	TC	AG	TC	AG	TC	AG	TC	AG	TC	AG
SC1	89.4301	89.705	77.3897	77.3897	15.3493	58.0882	93.75	94.117	70.9559	74.1728	84.1912	91.636
SC2	81.5789	82.7485	68.7135	69.0058	33.3333	85.9649	84.5029	84.795	69.5905	75.73	52.0877	84.5029
SC3	88.3171	89.29	80.1113	80.1113	16.968	93.324	95.1321	95.1321	69.541	76.6342	87.7872	93.185
SC4	86.2557	87.534	76.0731	76.347	31.8721	33.4703	94.5662	94.611	74.4292	77.853	83.6986	84.7032
SC5	93.1111	94	85.2222	85.5556	94.4444	94.7777	95.6667	95.888	81.1111	85.333	88.2222	91.222
SC6	98.7421	99.371	88.0503	90.5066	100	100	100	100	94.9686	96.226	95.5975	98.742
SC7	97.1347	97.1347	85.6734	86.246	95.8452	96.8481	96.7049	97.707	87.2493	90.83	93.9828	94.985
SC8	95.858	96.252	87.3767	88.757	95.4635	96.2524	98.2249	98.2249	83.0375	87.573	92.5049	93.885
SC9	87.6133	88.519	71.997	72.8097	90.3323	90.7854	91.6918	91.842	67.9758	75.075	78.0967	84.139
SC10	86.21	87.877	78.6841	79.495	45.0202	49.1212	96.0342	96.0342	82.5146	85.353	90.0406	94.0964
SC11	77.5168	78.187	63.0872	63.0872	11.5126	13.526	92.0496	92.0496	67.9143	70.134	74.8849	90.1652
T	79.8716	81.19	68.4436	68.4436	6.1151	7.1135	87.8409	89.32	66.7855	66.7855	72.1697	73.596

Tabla B-5: Comparativa de resultados antes y después de aplicar el Algoritmo Genético en datos con segmentación Otsu y características Texturales y Geométricas.

	Bayes Net		Navi Bayes		Perceptron Multicapa		SMO		J48		Random Forest	
	TC	AG	TC	AG	TC	AG	TC	AG	TC	AG	TC	AG
SC1	82.9208	85.643	80.8168	83.415	92.0792	92.0792	90.2228	90.2228	75.495	78.7129	86.1386	90.2228
SC2	89.8496	91.353	83.4586	85.714	92.8571	92.8571	92.1053	92.1053	78.1955	84.586	86.4662	90.601
SC3	84.3212	87.954	82.0268	86.042	91.7782	91.7782	88.5277	88.5277	77.0554	80.688	85.4685	88.9101
SC4	83.858	86.0683	81.8486	84.3269	92.4983	92.4983	88.1447	88.1447	80.71	81.446	85.2646	91.8955
SC5	89.5222	92.371	87.3964	90.049	93.2007	94.5273	92.3715	92.3715	81.592	85.074	91.2106	92.868
SC6	96.2264	99.371	91.195	98.1132	99.3711	100	99.3711	100	96.2264	98.113	97.4843	99.371
SC7	98.9293	99.571	96.788	97.858	98.2869	98.9293	96.788	98.501	92.7195	95.717	97.8587	99.571
SC8	96.114	98.704	94.3005	96.891	98.1865	98.7046	97.9275	98.186	88.0829	93.234	96.3731	97.927
SC9	85.2814	89.393	84.632	87.662	91.9913	92.2077	89.3939	90.476	77.7056	82.251	85.9307	89.393
SC10	89.0992	91.187	86.423	89.0339	95.235	95.235	91.5796	91.5796	85.4439	86.814	91.1227	93.6684
SC11	84.0495	87.1777	78.0681	80.4056	92.9873	92.9873	88.1403	88.1403	78.5693	79.718	89.9371	91.9904
T	85.9037	86.802	81.0481	83.893	91.9786	92.1069	88.1925	88.1925	78.8877	78.8877	88.7273	94.0321

Tabla B-6: Comparativa de resultados antes y después de aplicar el Algoritmo Genético en datos con segmentación Otsu y características Texturales

	Bayes Net		Navi Bayes		Perceptron Multicapa		SMO		J48		Random Forest	
	TC	AG	TC	AG	TC	AG	TC	AG	TC	AG	TC	AG
SC1	36.2624	42.45	31.8069	38.49	69.5545	70.544	48.3911	48.3911	51.7327	54.95	57.9208	62.6238
SC2	50.5618	56.928	39.7004	53.183	77.5281	77.902	55.4307	55.4307	62.5468	67.041	71.5356	72.284
SC3	47.2275	57.17	36.1377	49.904	80.6883	80.879	60.2294	60.2294	61.7591	64.244	68.6424	73.805
SC4	42.0348	45.046	35.2075	37.349	66.5997	68.674	44.5114	44.5114	53.5475	57.161	62.1151	63.654
SC5	39.6352	53.565	28.1924	38.805	71.3101	74.129	55.058	58.228	55.7214	61.359	65.6716	66.5
SC6	83.6478	89.937	75.4717	78.616	94.9686	98.742	89.3082	91.823	89.9371	93.081	93.7107	95.597
SC7	61.4561	64.6681	49.8929	64.8821	84.3683	86.081	65.0964	65.0964	75.803	77.944	79.015	80.728
SC8	75.544	75.544	57.2539	63.7306	85.2332	86.787	72.2798	72.2798	75.9067	79.274	77.9793	82.383
SC9	40.9091	45.454	37.4059	45.887	64.7186	66.017	51.2987	51.2987	48.0519	52.38	55.5887	57.142
SC10	44.2559	57.6371	39.0992	53.2637	77.5457	77.872	65.5352	65.5352	67.624	69.778	73.6945	75.4569
SC11	33.6542	38.191	21.0382	26.091	61.3957	63.114	41.8701	41.8701	48.8828	49.57	55.483	60.0206
T	26.5825	29.747	22.4979	26.368	55.2823	56.822	28.8922	28.8922	42.2797	42.2797	48.9307	50.299

Tabla B-7: Comparativa de resultados antes y después de aplicar el Algoritmo Genético en datos con segmentación Otsu y características Geométricas.

	Bayes Net		Navi Bayes		Perceptron Multicapa		SMO		J48		Random Forest	
	TC	AG	TC	AG	TC	AG	TC	AG	TC	AG	TC	AG
SC1	79.5037	81.709	74.6324	77.2978	91.3003	91.3003	84.0074	84.0074	78.8603	80.698	89.1544	92.1875
SC2	83.8235	88.823	78.5294	82.647	87.3529	92.647	81.4706	82.647	83.2353	85.882	89.4118	92.9412
SC3	78.3634	80.9986	72.9542	76.144	87.9334	88.21	74.3412	75.312	80.1664	80.721	90.5687	92.3717
SC4	75.0228	78.4868	71.7867	76.9371	90.7931	90.7931	82.9535	82.9535	81.7229	82.224	89.1067	93.7101
SC5	84.8889	89.222	84.3333	88.444	93.4444	94.222	89.1111	89.333	83.6667	86.444	92.4444	94.6667
SC6	92.4528	97.484	90.566	96.226	99.3711	99.3711	99.3711	99.371	94.9686	96.226	97.4843	99.371
SC7	98.3003	99.433	96.3173	97.592	98.8669	99.291	97.4504	98.158	96.7422	98.583	99.0085	99.575
SC8	94.8718	97.435	93.6884	96.252	97.8304	99.013	95.4635	95.857	91.9132	94.082	96.8442	98.027
SC9	84.4411	87.7644	80.2115	84.29	92.7492	93.806	85.9517	86.2537	83.9879	86.706	91.3897	94.864
SC10	79.5854	85.1735	74.9887	83.8666	93.4205	93.4205	83.6863	83.6863	85.8945	87.246	92.1136	93.6458
SC11	74.2916	78.851	69.4487	74.085	87.8413	87.8413	82.0453	82.0453	79.6239	79.829	89.052	89.155
T	79.6972	82.724	73.7845	76.26	88.7444	88.7444	77.3998	77.3998	81.2289	81.2289	90.7391	91.291

Tabla B-8: Comparativa de resultados antes y después de aplicar el Algoritmo Genético en datos con segmentación PCA y características Cromáticas, Texturales y Geométricas.

	Bayes Net		Navi Bayes		Perceptron Multicapa		SMO		J48		Random Forest	
	TC	AG	TC	AG	TC	AG	TC	AG	TC	AG	TC	AG
SC1	91.909	93.2996	80.1517	80.9102	94.5638	95.1959	96.2073	96.5865	78.7611	80.1517	85.5879	85.9671
SC2	87.7395	90.4214	76.6284	78.1609	88.1226	88.5057	90.0383	92.3371	74.3295	82.3754	85.8238	89.6551
SC3	93.797	95.3007	81.203	83.4586	95.4887	95.6766	96.9925	97.3684	81.7669	85.7142	89.0977	90.0375
SC4	90.9401	92.1662	78.406	79.7002	28.2016	30.4495	96.2534	96.3896	78.1335	79.019	84.4687	86.5803
SC5	95.7841	96.4586	82.7993	85.4974	95.7841	97.1332	98.145	98.8195	87.0152	88.7015	91.7369	93.2546
SC6	98.7421	99.371	91.195	95.5974	100	100	100	100	95.5975	97.4842	97.4743	98.7421
SC7	98.5356	99.3723	89.3305	91.6317	97.0711	97.6987	98.954	99.5815	93.3054	97.4895	97.4895	98.1171
SC8	96.3542	97.1354	89.8438	92.1875	97.3958	97.9166	98.1771	99.2187	92.1875	94.2708	95.0521	95.8333
SC9	95.1965	96.9432	75.3275	78.6026	94.1965	95.1965	93.8865	95.6331	82.3144	84.4978	87.9913	90.1746
SC10	90.0273	91.1202	79.3033	81.2841	45.9016	49.795	98.2923	98.4972	84.0161	86.2021	90.4372	91.53
SC11	86.4827	87.3319	71.2668	71.4083	9.1648	9.8372	93.8075	94.7983	75.7254	77.3885	83.581	84.2533
T	88.4886	89.9186	73.2877	73.2877	5.0727	6.9135	93.6429	94.4349	75.9846	76.6481	79.3022	80.77

Tabla B-9: Comparativa de resultados antes y después de aplicar el Algoritmo Genético en datos con segmentación PCA y características Cromáticas y Texturales

	Bayes Net		Navi Bayes		Perceptron Multicapa		SMO		J48		Random Forest	
	TC	AG	TC	AG	TC	AG	TC	AG	TC	AG	TC	AG
SC1	87.3578	89.5069	76.7383	77.6232	90.7711	92.0353	93.426	94.1845	69.2794	71.8078	80.2781	83.4386
SC2	82.0611	85.4961	72.1374	76.3358	84.3511	84.7328	84.7328	87.0229	67.9389	76.3358	79.3893	82.8244
SC3	90.4135	92.4812	78.7594	81.3909	93.7969	94.7368	95.8647	96.0526	71.0526	77.2556	87.218	87.218
SC4	87.2616	88.7602	75.0681	75.2724	30.7901	31.1307	94.0054	94.6866	68.8011	72.5476	80.7221	83.2425
SC5	92.2428	94.435	80.6071	84.1483	93.5919	94.435	95.9528	96.7959	81.4503	84.1483	87.0152	90.8937
SC6	98.7421	99.371	88.6792	92.4528	99.371	100	100	100	95.5975	96.2264	97.4843	100
SC7	96.8619	98.1171	86.4017	87.8661	96.0251	96.4435	97.2803	98.1171	87.2385	91.6317	94.1423	95.3974
SC8	95.0521	96.875	88.8021	91.6666	96.0937	96.875	97.1354	98.177	86.7188	90.8854	91.6667	94.2708
SC9	87.7729	90.8296	71.6157	73.7991	89.7379	91.2663	91.4847	92.358	67.6856	73.1441	81.0044	82.9694
SC10	87.7486	89.4125	77.2541	78.2786	46.8579	49.3852	96.9262	96.9262	81.0792	83.6065	86.6803	89.4808
SC11	80.5379	81.8117	66.4897	66.4897	8.351	13.4465	92.569	92.569	69.1437	70.7714	78.0962	80.0778
T	81.0788	82.8125	68.1507	68.1507	4.1755	5.9314	88.9554	90.625	64.2123	64.8972	71.8108	74.1438

Tabla B-10: Comparativa de resultados antes y después de aplicar el Algoritmo Genético en datos con segmentación PCA y características Cromáticas y Geométricas.

	Bayes Net		Navi Bayes		Perceptron Multicapa		SMO		J48		Random Forest	
	TC	AG	TC	AG	TC	AG	TC	AG	TC	AG	TC	AG
SC1	92.3713	93.3823	81.8015	82.261	96.2316	96.5073	97.2426	97.5183	80.9743	83.272	86.489	88.511
SC2	87.0588	88.8235	76.1765	76.4705	90	90	90.5882	91.1764	78.5294	82.0588	87.6471	89.4117
SC3	92.5104	93.7586	82.5243	83.079	96.3938	97.0873	96.9487	97.3647	83.6338	85.7142	89.8752	90.5686
SC4	91.4234	92.3813	80.2464	80.3832	33.7135	35.5383	97.2628	97.354	81.0675	82.5273	88.458	88.9598
SC5	96.3333	96.8888	83.3333	88.6666	96.6666	96.8888	98.1111	98.6666	87.6667	90	92.6667	94.222
SC6	98.7421	99.371	91.195	95.5974	100	100	100	100	95.5975	96.8553	97.4843	99.371
SC7	98.8539	99.1404	88.8252	91.2607	96.991	97.7077	98.5663	98.5673	95.5587	97.7077	97.5645	98.2808
SC8	96.8442	98.2248	88.9546	90.927	97.2386	97.8303	99.211	99.6055	92.7022	93.8856	95.858	96.2524
SC9	94.4109	95.1661	76.435	78.2477	93.6555	95.0151	96.0725	96.3746	82.3263	85.4984	87.4622	91.6918
SC10	89.7706	90.7616	80.5768	82.2893	50.7886	52.456	97.2961	97.6115	87.607	89.0941	92.6093	92.6543
SC11	83.8152	85.3639	68.6113	68.6113	14.5069	15.3588	94.889	94.889	76.8715	77.2586	83.8152	85.3381
T	87.7853	88.5342	44.0068	72.7175	5.0641	6.883	94.0263	94.0263	77.1041	77.9957	80.4387	82.6319

Tabla B-11: Comparativa de resultados antes y después de aplicar el Algoritmo Genético en datos con segmentación PCA y características Cromáticas

	Bayes Net		Navi Bayes		Perceptron Multicapa		SMO		J48		Random Forest	
	TC	AG	TC	AG	TC	AG	TC	AG	TC	AG	TC	AG
SC1	88.7052	90.266	77.4105	77.4105	72.4517	84.2975	94.2149	94.2149	71.9927	74.471	84.2157	93.2966
SC2	81.5789	82.7485	63.2982	69.298	84.5029	84.7953	84.5029	86.549	67.2515	73.684	82.4561	90.3509
SC3	88.2271	88.781	78.5319	78.5319	93.7673	94.3213	94.7368	94.7368	71.0526	73.684	85.0416	92.7978
SC4	86.2557	86.861	76.0731	76.414	74.4292	74.4292	94.5662	94.799	74.4292	75.7299	83.6986	92.3814
SC5	93.6777	94.006	84.5727	84.683	94.1176	94.4506	95.5605	96.004	81.465	84.794	87.7913	93.3407
SC6	99.3711	99.3711	88.0503	91.194	100	100	100	100	95.5775	95.597	98.1132	99.3711
SC7	96.5616	97.134	85.6734	86.103	95.8452	96.8481	96.7049	96.848	87.5358	91.1175	93.2665	97.4212
SC8	95.858	96.055	87.9684	89.349	95.8579	96.2524	97.8304	98.224	83.8264	87.9684	92.8979	95.6607
SC9	88.3861	88.8386	71.9457	72.85	90.0452	90.7993	90.6486	91.402	67.27	74.8115	80.2413	92.7602
SC10	86.3001	87.787	78.639	79.089	48.5353	49.3916	95.854	95.989	82.8301	84.858	89.4096	94.0964
SC11	77.62	78.781	62.984	62.984	11.7965	15.9525	92.3335	92.3335	68.5854	69.8245	78.9365	90.8105
T	79.373	80.922	69.1486	69.1486	4.9875	2.2116	87.5133	89.77	66.637	66.637	73.4592	89.2412

Tabla B-12: Comparativa de resultados antes y después de aplicar el Algoritmo Genético en datos con segmentación PCA y características Texturales y Geométricas.

	Bayes Net		Navi Bayes		Perceptron Multicapa		SMO		J48		Random Forest	
	TC	AG	TC	AG	TC	AG	TC	AG	TC	AG	TC	AG
SC1	86.4728	88.748	82.5537	85.082	93.8053	93.8053	91.2769	91.2769	77.6213	80.783	88.1163	92.1618
SC2	88.1226	90.804	81.1418	87.356	93.8697	95.4022	90.4215	90.4215	78.9272	83.141	89.6552	90.421
SC3	85.1504	87.593	80.6391	84.0226	93.0451	93.7969	89.2857	89.2857	80.2632	82.142	85.7143	91.1654
SC4	84.3324	85.4223	82.2207	84.1962	92.3706	92.7111	88.7206	88.7206	79.2916	80.04	84.8774	92.2343
SC5	90.2192	93.086	88.8702	91.062	94.6037	94.9409	93.086	93.423	83.8111	85.328	88.027	94.2604
SC6	96.2264	98.7421	91.8239	96.226	99.3711	100	99.3711	100	95.5975	97.484	96.2264	98.7421
SC7	99.3802	99.586	97.5207	98.553	98.5337	98.9669	98.7603	98.7603	95.0413	97.727	98.1405	99.5868
SC8	96.6146	98.697	97.1354	98.437	98.6979	99.4791	98.9583	99.218	93.4896	95.312	95.3125	97.656
SC9	87.5546	90.174	87.1179	89.082	92.1397	93.6681	90.1747	90.6113	80.786	84.934	87.1179	91.9214
SC10	90.5055	92.144	87.4317	89.6175	95.6284	95.9699	91.9399	91.9399	84.4262	86.748	91.3934	93.8525
SC11	84.0395	86.8644	78.6017	80.79	92.6554	92.6554	87.8178	87.8178	79.3432	80.755	87.2881	91.702
T	86.3403	87.43	81.5306	85.292	93.9504	93.9504	89.2048	89.2048	79.7349	79.7349	88.9483	94.8696

Tabla B-13: Comparativa de resultados antes y después de aplicar el Algoritmo Genético en datos con segmentación PCA y características Texturales

	Bayes Net		Navi Bayes		Perceptron Multicapa		SMO		J48		Random Forest	
	TC	AG	TC	AG	TC	AG	TC	AG	TC	AG	TC	AG
SC1	36.2832	40.075	31.0999	37.8	67.7851	71.934	46.9027	46.9027	49.9368	53.223	59.0392	62.4526
SC2	48.8733	55.725	41.6031	54.1985	78.2443	78.2443	53.0534	54.198	67.1756	69.083	64.5038	72.519
SC3	47.1805	56.39	36.4662	46.052	77.0677	78.759	57.5188	57.5188	62.218	64.097	70.8647	71.992
SC4	41.7291	45.426	33.8325	39.006	67.3247	68.277	45.8816	45.8816	54.9653	56.228	62.083	63.648
SC5	40.3035	52.951	29.8482	40.64	75.0422	75.0422	57.8415	57.8415	58.6847	63.406	63.2368	67.622
SC6	85.5346	91.194	76.1006	82.389	93.7107	97.484	89.3082	91.194	91.8239	94.968	91.195	95.597
SC7	64.2562	66.735	51.6529	66.5289	87.3967	87.809	65.0826	65.0826	75.2066	78.099	79.3388	83.2645
SC8	66.4063	70.833	60.4167	66.145	87.2396	87.2396	71.875	71.875	73.4375	77.864	78.6458	83.333
SC9	40.393	44.978	38.209	45.851	68.7763	68.7763	48.69	48.69	51.7467	53.711	55.8952	60.043
SC10	39.5492	56.8306	47.3361	53.347	78.8934	78.8934	66.8716	66.8716	65.9836	68.647	73.9071	75.7514
SC11	33.7218	39.159	21.6102	28.036	63.4181	64.371	42.161	42.161	48.7994	49.858	54.1314	56.497
T	28.2386	30.654	22.6165	27.233	56.5413	56.5413	29.5853	29.5853	40.3005	41.983	49.9786	50.662

Tabla B-14: Comparativa de resultados antes y después de aplicar el Algoritmo Genético en datos con segmentación PCA y características Geométricas.

	Bayes Net		Navi Bayes		Perceptron Multicapa		SMO		J48		Random Forest	
	TC	AG	TC	AG	TC	AG	TC	AG	TC	AG	TC	AG
SC1	81.8934	83.731	77.3897	80.514	93.5662	93.5662	87.4081	87.4081	82.261	83.547	92.6471	95.4963
SC2	85.5882	87.941	77.3529	82.647	90	92.647	82.9412	83.5294	81.4706	87.0588	89.7059	93.5294
SC3	78.0886	81.4147	74.7573	77.808	89.3204	89.3204	78.9182	79.7503	80.8599	81.414	88.7656	91.5395
SC4	74.7037	77.028	72.2881	77.256	90.7475	90.7475	83.4093	83.4093	81.1304	81.631	89.7903	90.428
SC5	85.4444	88.888	85.7778	88.222	95.1111	95.1111	90	90.222	86	88.333	91.2222	96
SC6	93.0818	97.484	91.195	94.339	99.3711	99.3711	98.7421	99.371	94.3396	95.597	96.8553	99.371
SC7	97.8754	98.725	97.0255	98.3	98.7252	99.433	97.7337	98.583	97.5921	98.441	99.0085	99.575
SC8	95.2663	96.844	94.4773	97.238	98.8166	99.211	97.0414	97.633	90.5325	93.885	97.2387	98.027
SC9	84.5921	87.915	80.6647	84.441	93.8076	93.8076	94.4109	94.4109	82.6284	85.649	92.5982	93.353
SC10	80.2163	84.182	76.521	84.362	93.105	93.285	84.1821	84.1821	86.0748	87.652	92.2488	93.15
SC11	74.8326	78.954	70.3503	74.806	88.3565	88.3565	81.3241	81.3241	80.0103	80.783	89.1293	89.82
T	79.9858	81	73.0413	75.961	89.1026	89.1026	79.0954	79.0954	79.7899	80.306	91.4708	91.684

Apéndice C

Programas de apoyo

C.1. Programa para el AG básico

El algoritmo genético que permitió la reducción de características fue realizado en el lenguaje Java. Se utilizó como IDE (*Integrated Development Environment*) el programa NetBeans 8.0.1 junto con el API (*Application Programming Interface*) de Weka para obtener las precisiones en cada clasificación. La interfaz de dicho programa se muestra en la Figura C-1.

A continuación, se describen las funciones de cada apartado del programa:

Carpeta destino: Es la ruta de la carpeta que contiene todos los archivos que se les aplicara el genético.

Guardar resultados en: Contiene la ruta de la carpeta en donde se guardaran los datos finales.

Tamaño de población: Indica el número de población con que se iniciara el genético.

Numero de características iniciales: Obtiene el número de características por cada archivo para definir el tamaño de las cadenas binarias.

Razón de paro: Indica el número de veces que el genético se ejecuta cuando no encuentre individuos con mejor aptitud.

Aptitudes iniciales: Contiene la cadena binaria con mejor aptitud con que inicio el genético. (Cada cadena pertenece a un archivo distinto).

Aptitudes finales: Contiene la cadena binaria que obtuvo la mejor aptitud cuando termino el AG.

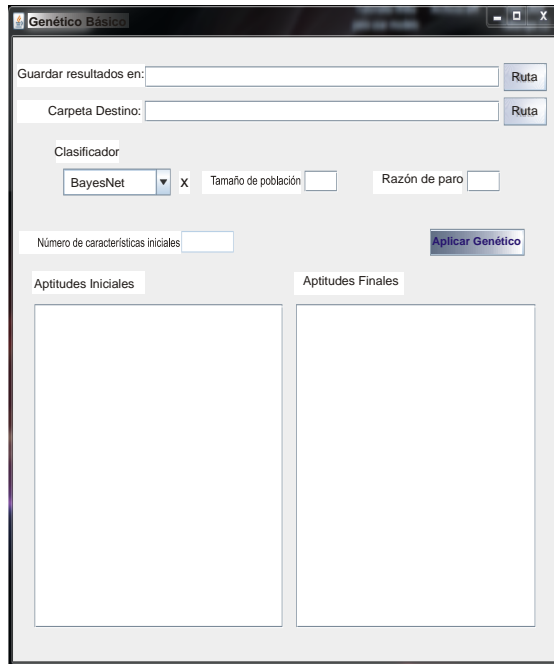


Figura C-1: Interfaz del AG propuesto programado en Java.

Clasificador: Despliega una lista con los 6 clasificadores a evaluar.

C.1.1. Manejo del programa

Al iniciar el programa todos los campos se encuentran vacíos y el programa no permite comenzar el AG si no se ha llenado todo correctamente.

Una ventaja de este programa, es que puede aplicarse el AG a todos los archivos que se encuentran en una carpeta, sin necesidad de estar cargando archivo por archivo. Así que antes de utilizar el programa, se deben guardar todos los archivos a los que se les aplicara el AG en una carpeta. Además, se debe tener otra carpeta vacía que servirá como destino para almacenar los resultados finales. En el apartado 'Carpeta destino' se elige la ruta de la carpeta que contiene los archivos. En el apartado "Guardar resultado en:" se elige la ruta de la carpeta vacía designada.

En el cuadro desplegable se elige uno de los 6 clasificadores que será el que devuelva las precisiones de la población. En el apartado 'Tamaño de población' se coloca el número de individuos que conformara la población. . Es sabido, que entre más población haya, más posibilidad existe de encontrar un mejor resultado. Sin embargo, para clasificadores como el Perceptron Multi-

capa o SMO que consumen más tiempo en devolver resultados puede ser impredecible cuando terminara el AG. Por lo tanto se recomienda manejar poblaciones pequeñas. En el apartado ‘Razón de paro’, se coloca el número de veces que el AG continuara si no encuentra mejora en las aptitudes. Es importante considerar un valor pequeño para la razón de paro, ya que cuando el AG no genera mejora es porque la mayoría de su población ha convergido de forma similar, careciendo de diversidad. Así que el colocar un valor alto en la razón de paro, seria utilizar tiempo del computador inútil.

Una vez llenado todos los campos correctamente, se puede dar clic en el botón ‘Aplicar Genético’. El programa iniciara colocando el número de características encontradas en el archivo en el apartado ‘Número de características iniciales’. Cuando finaliza el programa, coloca las cadenas con que inicio cada archivo en el recuadro ‘Aptitudes iniciales’. También coloca la cadena con mejor aptitud en el recuadro ‘Aptitudes Finales’. La Figura C-2 muestra la interfaz después de haber finalizado el AG.

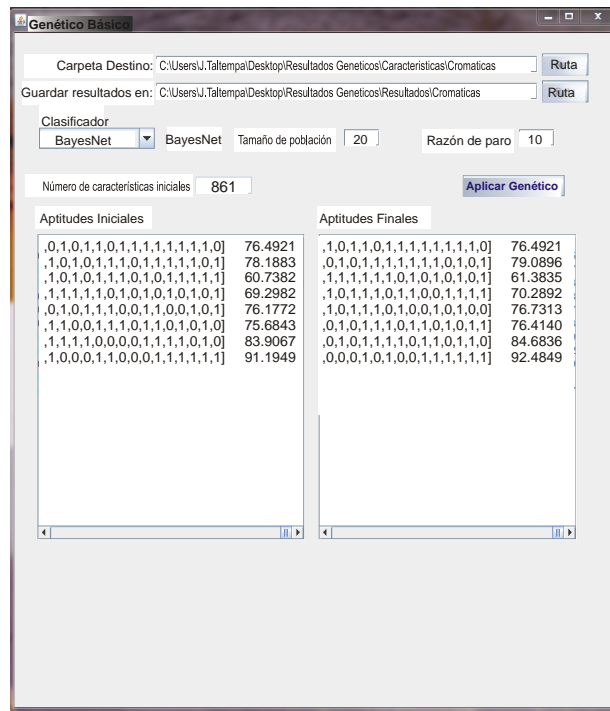


Figura C-2: Ejemplo de datos resultantes después de finalizar el AG.

Las cadenas mostradas en el recuadro ‘Aptitudes iniciales’ solo se muestran en la interfaz

del programa pero no son almacenadas en ningún lado. Las cadenas mostradas en el recuadro ‘Aptitudes finales’ son almacenadas por el programa en un archivo de texto en la carpeta que se eligió como destino. El programa almacena un archivo de texto por cada archivo de características. Para identificar que archivo de texto corresponde con que archivo de características, el programa nombra al archivo de texto con el mismo nombre del archivo añadiendo al inicio el tipo de clasificador con el que se trabajó. Por ejemplo si el archivo tiene como nombre ‘OTSU_CXX_CG1.arff’ y se trabajó con el clasificador Bayes Net, el nombre del archivo de texto será ‘BayesNet_OTSU_CXX_CG1.txt’.

Apéndice D

Artículos Publicados

Los artículos publicados que parten de esta tesis y llevan por nombre ‘Mejorando la Clasificación en Conjuntos de Datos No-balanceados Utilizando un Algoritmo Genético para Selección de Atributos’ y ‘Análisis Comparativo de las técnicas utilizadas en un Sistema de Reconocimiento de Hojas de Planta’ se presentan a continuación.

> De: ees.riai.0.33e0ee.cda2b956@eesmail.elsevier.com
 [ees.riai.0.33e0ee.cda2b956@eesmail.elsevier.com] En nombre de RIAI [secretaria@revista-riai.org]
 > Enviado el: viernes, 11 de septiembre de 2015 09:03 a.m.
 > Para: Jesus Taltempa; talte_203@hotmail.com
 > Asunto: RIAI: Confirmación pendiente / Pending OK
 >
 > Revista Iberoamericana de Automática e Informática Industrial
 > Título: Análisis Comparativo de las técnicas utilizadas en un Sistema de Reconocimiento de Hojas de Planta Comparative analysis of the techniques used in a recognition system of plant leaves
 > Autores: Ing.; Jesus Taltempa, Dr.; Jair Cervantes, Ph.D.; Marcos Cano, Ing.; Laura Jalili, Ing.; José Sergio Ruiz, Dr.; Adrian Trueba.
 >
 > Estimado/a Jesus:
 >
 > El PDF de su artículo "Análisis Comparativo de las técnicas utilizadas en un Sistema de Reconocimiento de Hojas de Planta Comparative analysis of the techniques used in a recognition system of plant leaves" ya se ha generado y está listo para su aprobación. Por favor, revise el pdf antes de aceptarlo para confirmar que no contiene errores. Si ya ha aceptado el PDF de su artículo, ignore este mensaje.
 >
 > Para aprobar el PDF, por favor, entre en su cuenta de Elsevier Editorial System como Autor:
 >
 > <http://ees.elsevier.com/riai/>
 > Your username is: talte_203@hotmail.com
 >
 > Entre en la carpeta "Submissions Waiting for Author's Approval" para revisar y aceptar el PDF de su artículo. Si lo desea, puede clicar en "Action Links" para desplegar los links de la columna "Action".
 >
 > También tendrá que confirmar que ha leído y que acepta la declaración "Ethics in Publishing" antes de completar el proceso de envío. Una vez termine todos estos pasos, recibirá un email de confirmación de la Oficina Editorial. Para más información o si tiene alguna duda con el proceso de envío, consulte:
 > http://help.elsevier.com/app/answers/detail/a_id/1411/p/7923/.
 >
 > Por favor, asegúrese de que todos los componentes de su artículo aparecen correctamente en el PDF ya que no será posible modificarlo una vez haya confirmado que es correcto. Si tiene algún problema con el PDF o para completar estos pasos, consulte http://help.elsevier.com/app/answers/detail/a_id/1411/p/7923/.
 >
 > Recibirá un email con el número de referencia de su artículo cuando este se asigne a uno de los editores de la revista.
 >
 > Gracias por su tiempo y su paciencia.
 > Reciba un cordial saludo,
 >
 > Oficina Editorial
 > Revista Iberoamericana de Automática e Informática Industrial
 >
 > *****

> Para más información sobre el programa, visite nuestra página de soporte
http://help.elsevier.com/app/answers/detail/a_id/732/p/7923 (listado de soluciones disponibles en español. Para buscar más soluciones en inglés, introduzca su búsqueda en el panel superior).

> En esta página encontrará soluciones a diversos temas, tutoriales interactivos y las respuestas a las preguntas más frecuentes sobre el funcionamiento del EES. También encontrará los datos de contacto de nuestro servicio de asistencia telefónica 24/7, en caso que necesite ayuda de uno de nuestros agentes.

> _____

>

> Dear Jesus,

>

> The PDF for your submission, "Análisis Comparativo de las técnicas utilizadas en un Sistema de Reconocimiento de Hojas de Planta Comparative analysis of the techniques used in a recognition system of plant leaves" has now been built and is ready for your approval. Please view the submission before approving it, to be certain that it is free of any errors. If you have already approved the PDF of your submission, this e-mail can be ignored.

>

> To approve the PDF please login to the Elsevier Editorial System as an Author:

>

> <http://ees.elsevier.com/riai/>

> Your username is: talte_203@hotmail.com

>

> Then click on the folder 'Submissions Waiting for Author's Approval' to view and approve the PDF of your submission. You may need to click on 'Action Links' to expand your Action Links menu.

>

> You will also need to confirm that you have read and agree with the Elsevier Ethics in Publishing statement before the submission process can be completed. Once all of the above steps are done, you will receive an e-mail confirming receipt of your submission from the Editorial Office. For further information or if you have trouble completing these steps please go to:
http://help.elsevier.com/app/answers/detail/a_id/88/p/7923.

>

> Please note that you are required to ensure everything appears appropriately in PDF and no change can be made after approving a submission. If you have any trouble with the generated PDF or completing these steps please go to: http://help.elsevier.com/app/answers/detail/a_id/88/p/7923.

>

> Your submission will be given a reference number once an Editor has been assigned to handle it.

>

> Thank you for your time and patience.

> Kind regards,

> Editorial Office

> Revista Iberoamericana de Automática e Informática Industrial

>

> *****

> For further assistance, please visit our customer support site at
<http://help.elsevier.com/app/answers/list/p/7923>. Here you can search for solutions on a range of topics, find answers to frequently asked questions and learn more about EES via interactive tutorials. You will also find our 24/7 support contact details should you need any further assistance from one of our customer support representatives.

>

Análisis Comparativo de las técnicas utilizadas en un Sistema de Reconocimiento de Hojas de Planta

Jesús Taltempa^{1,*}, Jair Cervantes¹, Marcos A. Cano¹, Laura D. Jalili¹, José S. Ruiz Castilla¹, Adrian Trueba¹

Resumen

El desarrollo de sistemas de identificación de hojas de plantas es un reto actual que constituye numerosas aplicaciones que van desde alimentación, medicina, industria y medio ambiente. En la literatura actual, se han propuesto varias técnicas con el objetivo de identificar plantas en diversos campos de aplicación. Sin embargo, las técnicas actuales están restringidas al reconocimiento e identificación de tipos de plantas limitados, utilizando descriptores de características específicos. En este artículo, se realiza un análisis comparativo de diversos métodos de extracción de características (texturales, cromáticas y geométricas) y clasificación sobre conjuntos de plantas muy similares y disimilares entre sí. Doce conjuntos de hojas con características de forma similares son estudiados utilizando varios clasificadores. Se analiza el desempeño de diferentes combinaciones de características en cada conjunto. Los resultados obtenidos muestran que para incrementar el desempeño de los clasificadores estudiados, es necesaria una combinación de las diferentes técnicas de extracción de características, esta necesidad es mayor cuando se trabaja con conjuntos de hojas con características muy similares. Además, se muestra el mejor desempeño de un clasificador con otro. *Copyright © 2015 CEA. Todos los derechos reservados.*

Palabras Clave: Clasificación, Descriptores, SVM, Conjuntos de Datos

1. Introducción

La implementación de algoritmos de visión en la actualidad abarca casi cualquier campo de aplicación que uno pudiera imaginar. La agronomía parece no ser la excepción. El desarrollo de sistemas de visión que detecten automáticamente e identifiquen las plantas es un reto actual que tiene numerosas aplicaciones. Estas pueden ir desde simple identificación, detección de plagas, detección de enfermedades, identificación de plantas por personal de aduanas, identificación de la planta para cuidado y protección de pesticidas, etc.

La identificación de plantas no es un reto fácil debido a que existen muchas características foliares que los botánicos emplean para identificar las plantas, además de la enorme cantidad de diferentes plantas, muchas de las cuales poseen y/o comparan una o varias propiedades como: forma, tamaño, textura, color, aún cuando pertenecen a plantas diferentes. Aunado a ello existen otros factores que puede influir en algunas características de la planta como: la cromaticidad, color y textura que se ven afectadas principalmente por nivel de madurez de la hoja, humedad, enfermedades, etc.

El desarrollo de algoritmos de identificación automática de hojas ha sido direccionado por varios investigadores en años recientes. Sin embargo, en reconocimiento de patrones los métodos de identificación pueden restringirse a descriptores geométricos, texturales, cromáticos y de venación.

En este artículo, estudiamos la influencia de los diferentes tipos de características (geométricas, texturales y cromáticas) en la precisión de clasificación. Además se analiza el desempeño de varios clasificadores sobre conjuntos de imágenes con características geométricas muy similares y disimilares entre sí.

En la metodología propuesta empleamos técnicas de extracción de características geométricas invariantes a escalado, traslación y rotación, técnicas de extracción de características texturales y cromáticas. En la fase de clasificación utilizamos un clasificador Bayesiano, el algoritmo Backpropagation y Máquinas de Vectores de Soporte (SVM). Los resultados obtenidos muestran la influencia de los tipos de características en la precisión de clasificación.

2. Estado del Arte

Las plantas juegan un importante papel para la vida y el desarrollo humano, ya que no solo son de interés en investigaciones de botánica, sino también en otras ramas, tales como la agricultura (29) (1) (12) (27), ecología vegetal (29) (30) (7),

*Todos los autores pertenecen a la Universidad Autónoma del Estado de México, Prolongación de Av. Zumpango s/n, Fraccionamiento El Tejocote, Texcoco, México, 52346.

Correos electrónicos:
talte203@hotmail.com(JesúsTaltempa), jcervantesc@uaemex.mx(JairCervantes),

medicamentos basado en plantas (6) (12) (30) (8), conservación natural y también en muchas situaciones de interés público. En el mundo, existen aproximadamente una variedad de 310 000 a 420 000 especies de plantas (6), sin tomar en cuenta que aún existen muchas que no han sido clasificadas. Por esta razón, identificar una planta a partir de imágenes de hojas no es una tarea trivial. Las técnicas de reconocimiento de patrones empleadas actualmente involucran técnicas de medición de características morfológicas y de textura de los objetos contenidos en la imagen y el desarrollo de sistemas capaces de reconocer el objeto a partir de aquellas características de medida. Es bien sabido que la mejor forma de extraer características válidas es basándose en la imagen de la hoja de la planta. En la literatura actual se ha mostrado que la forma externa de la hoja provee rica información para clasificar. Así varios estudios, se han enfocado a la extracción de características y métodos de reconocimiento de patrones para la hoja, utilizando cuatro importantes características para clasificación, que son: La forma (6) (11) (19) (27) (8), textura (22) (1) (12), el color (29) (3), y la venación de la hoja (7) (15) (16).

La forma de la hoja es una de las características más importantes de la hoja de la planta y los dos enfoques básicos para este tipo de análisis son los basados en contorno y basados en la región. El enfoque basado en la región suele utilizar descriptores de momentos, que incluyen momentos geométricos, momentos de Zernike y momentos de Legendre. El enfoque basado en el contorno usualmente obtiene el contorno con métodos basados en la curvatura de la hoja (29). Otros estudios han utilizado una combinación de características geométricas y texturales, permitiéndoles incluso utilizar hojas secas, mojadas o deformes (12). Los descriptores utilizados por los distintos autores van desde descriptores básicos como perímetro, área, circularidad y elipsoidad (22), hasta descriptores invariantes como momentos de Hu y descriptores de Fourier para reconocimiento de contorno de la hoja (19). Recientemente se han propuesto sistemas para extraer características que describen variaciones del borde de la hoja, utilizando descriptores invariantes a traslación, rotación y tamaño (8).

La textura de la hoja puede ser definida como las características que la hoja posee en su superficie y que manifiesta como variaciones de escala de grises en la imagen, a partir de donde se extraen las matrices de coocurrencia que servirán para obtener los descriptores texturales (17) (18).

Otros estudios usan el color como característica de comparación de imágenes, ya que una simple similitud de color entre dos imágenes puede ser medida comparando sus histogramas de color (29). En (3) utilizan como alternativa el espacio de color $L^* a^* b^*$, que muestra colores más consistentes y presenta más o menos el mismo eje para toda la hoja a diferencia del espacio RGB. Aunque los enfoques de clasificación como la forma, textura y color son válidos, no se ha documentado la influencia de cada tipo de características en el desempeño de los algoritmos de clasificación,

Los primeros estudios de reconocimiento de plantas utilizaron la cromaticidad de la planta como un descriptor importante para comparar imágenes. Descriptores muy simples de cromaticidad pueden obtener el color promedio en la región previa-

mente segmentada de la hoja, gradiente promedio en el borde o la similitud de color entre dos imágenes que se puede medir mediante la comparación de sus histogramas de color. Descriptores de color más complejos utilizan momentos de invariancia comúnmente utilizados para obtener características geométricas pero incorporándoles la información de la variables de color de la hoja (26) (23) (25). Sin embargo, un problema recurrente en las hojas de las plantas es que la cromaticidad de estas no es estática, esta es variable con respecto al tiempo y comúnmente con respecto a otros factores más. Uno de los enfoques más empleados en la actualidad consiste en analizar la forma de la hoja extrayendo características geométricas como tamaño, elongación, elipsoidad, área, longitud, diámetro, rectangularidad, esfericidad, excentricidad, etc. (4) (21) (28) (13) (14). Algunos autores han agregado a estos descriptores geométricos básicos, momentos de Hu y momentos de Fourier mejorando el desempeño de los clasificadores (20) (24) (9). Otros autores consideran además de cromaticidad y forma, la textura de la hoja (19) (5) o utilizan combinaciones de descriptores para mejorar el desempeño de clasificación (8) (12) (29) (2).

Existe gran cantidad de información en la literatura sobre identificación de plantas y/o arboles a partir de hojas. Sin embargo, en la mayoría de los sistemas actuales es fácil identificar tres problemas:

- No existe un estudio completo sobre identificación complejo de plantas, es decir, un estudio que muestre la eficiencia de un algoritmo sobre conjuntos de imágenes muy similares entre sí.
- Los trabajos actuales utilizan características texturales, cromáticas y geométricas. Sin embargo, no existen trabajos que realicen un análisis sobre la influencia de las características en el desempeño del clasificador.

En esta investigación se estudia la influencia de los tipos de descriptores en la precisión de clasificación en dos tipos generales de imágenes; imágenes muy disimilares entre si e imágenes con propiedades muy similares entre sí.

3. Método Propuesto

La metodología del sistema propuesto es mostrada en la Figura 1. Los pasos son los habituales en cualquier sistema de reconocimiento a partir de características. En los experimentos realizados se obtuvieron características geométricas, cromáticas y texturales, no se incluyeron características de venación debido a que el métodos tiene algunos fallos cuando las imágenes de hojas no contienen una venación prominente. Debido a las condiciones de espacio se definen brevemente los pasos que son comunes en los sistemas de clasificación y que son llevados a cabo también por el método propuesto.

Primero, las imágenes son preprocesadas y segmentadas. Regularmente en preprocesamiento se emplea una mascara Gaussiana para obtener una buena segmentación. En las simulaciones realizadas se aplicó este tipo de mascarar. Sin embargo, debido a que las imágenes fueron tomadas en un ambiente contro-

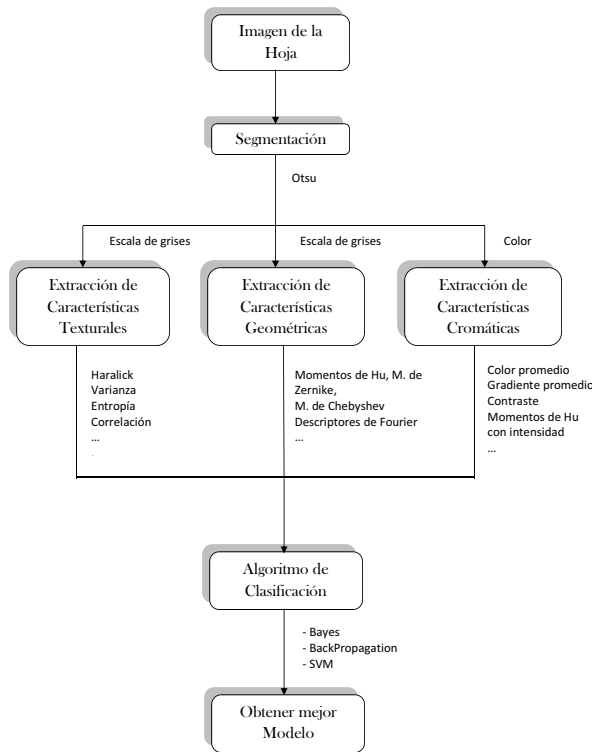


Figura 1: Diagrama de metodología propuesta

lado, se notó que la fase de preprocesamiento no era necesaria para este conjunto de datos.

En general, la segmentación autónoma es una de las tareas más difíciles en el procesamiento de imágenes. En imágenes de hojas a menudo están rodeadas de zonas verdes en el fondo. Sin embargo, las imágenes utilizadas son hojas en ambientes totalmente controlados (imágenes de únicamente la hoja con fondo blanco). Se realizaron pruebas de segmentación con los algoritmos de segmentación frontera adaptativa, Otsu y segmentación utilizando una fase de análisis de componentes principales (PCA) y no se obtuvieron diferencias entre estos por la naturaleza del conjunto de datos, por ello finalmente se utilizó el algoritmo de Otsu para realizar la segmentación.

3.1. Técnicas de segmentación

La región de la hoja en cada imagen fue segmentada empleando los siguientes pasos 1) Cálculo de alto contraste en escala de grises a partir de combinación lineal óptima de los componentes de color en RGB; 2) Estimar una frontera óptima empleando momentos acumulativos de orden cero y de primer orden (método de Otsu). 3) Operaciones morfológicas para rellenar posibles espacios vacíos en la imagen segmentada. Todo esto con el objetivo de obtener una buena segmentación aún cuando existan cambios en las condiciones globales de brillo. Al segmentar la imagen, el sistema propuesto puede utilizar únicamente la región de la hoja, determinar sus bordes y calcular las propiedades mediante la extracción de características.

3.2. Extractores de Características

Una vez segmentada la región se extraen sus características. La extracción de características nos permite representar la imagen mediante un conjunto de valores numéricos con gran poder discriminativo, eliminando características redundantes y reduciendo la dimensionalidad de la imagen. Las características obtenidas son capaces de asociar rangos muy similares a imágenes similares, asociar rangos diferentes a imágenes diferentes, además de ser invariantes a escalado, rotación y traslación, permitiendo al clasificador reconocer objetos a pesar de tener diferente tamaño, posición y orientación. Todas estas características juegan un rol importante en el desempeño del algoritmo y permiten al clasificador discriminar de una forma apropiada entre distintas clases.

3.2.1. Características geométricas

Las características geométricas son los rasgos visuales más importantes y utilizados para clasificar plantas. Las características geométricas más elementales describen las propiedades básicas de la región a reconocer, estas son; área de la región, redondez de la hoja, longitud del borde de la hoja, elongación definida por la longitud y ancho de la hoja, las coordenadas x e y del centro de gravedad, densidad, definida por la longitud de los bordes de la hoja y el área de esta. Sin embargo, un sistema eficiente de clasificación de hojas debe permitir reconocer las hojas independientemente de su orientación, localización y tamaño, i.e. debe ser invariante a escalado, rotación y traslado.

Los momentos son comúnmente utilizados en reconocimiento de imágenes, estos permiten reconocer imágenes independientemente de su rotación, traslación o inversión. Los momentos invariantes fueron inicialmente introducidos por Hu (10). Los dos momentos de Hu de orden $(p + q)$ de una función de intensidad $f(x, y)$ son definidos como:

Los momentos de orden $(p + q)$ son definidos como:

$$m_{pq} = \sum_x \sum_y x^p y^q \rho(x, y). \quad (1)$$

donde $\rho(x, y)$ es definida por la región segmentada. Los momentos de orden pequeño describen la forma de la región. Por ejemplo m_{00} describe el área de la región segmentada, mientras que m_{01} y m_{10} definen las coordenadas x e y del centro de gravedad. Sin embargo, los momentos m_{02} , m_{03} , m_{11} , m_{12} , m_{20} , m_{21} y m_{30} son invariantes a traslación, rotación e inversión. Los momentos centrales son invariantes a desplazamiento y pueden ser calculados mediante

$$\mu_{pq} = \sum_{i, j \in R} (i - \bar{i})^p (j - \bar{j})^q \quad (2)$$

donde p, q pertenecen a la región segmentada y el centro de gravedad de la región es definido por:

$$\bar{i} = \frac{m_{10}}{m_{00}}, \bar{j} = \frac{m_{01}}{m_{00}} \quad (3)$$

Los momentos de Hu pueden ser obtenidos de la siguiente manera:

$$\begin{aligned}
 \phi_1 &= \eta_{20} + \eta_{02} \\
 \phi_2 &= (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2 \\
 \phi_3 &= (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - \eta_{03})^2 \\
 \phi_4 &= (\eta_{30} - 3\eta_{12})^2 + (\eta_{21} + \eta_{03})^2 \\
 \phi_5 &= (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] + \\
 &\quad (3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] \\
 \phi_6 &= (\eta_{20} - \eta_{02})[(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] + \\
 &\quad 4(\eta_{11}(\eta_{30} + \eta_{12})(\eta_{21} + \eta_{03})) \\
 \phi_7 &= (3\eta_{21} - \eta_{03})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] - \\
 &\quad (\eta_{30} - 3\eta_{12})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2]
 \end{aligned} \tag{4}$$

donde $\eta_{pq} = \frac{\mu_{rs}}{\mu_{00}^t}$, $t = \frac{p+q}{2} + 1$.

Otras características empleadas fueron descriptores de elipse, convexidad de región, momentos de Flusser y orientación, en total 57 características geométricas fueron extraídas de cada imagen.

3.2.2. Características texturales

Los algoritmos de extracción de características texturales buscan patrones repetitivos básicos con estructuras periódicas o aleatorias en imágenes. Estas estructuras dan lugar a una propiedad que puede ser rugosidad, aspereza, granulación, fineza, suavidad, etc. Debido a que una textura repite un patrón a lo largo de una superficie, las texturas son invariantes a desplazamientos, ello explica porqué la percepción visual de una textura es independiente de la posición. En este artículo, se utilizaron características texturales de Haralick. Estos extractores toman en cuenta la distribución de valores de intensidad en la región obteniendo la media y rango de las siguientes variables: media, mediana, varianza, suavidad, sesgo, curtosis, correlación, energía o entropía, contraste, homogeneidad, y correlación. Los descriptores de Haralick utilizados son descritos a continuación:

$$f_1 = \sum_i \sum_j [p(i, j)^2] \tag{5}$$

$$f_2 = \sum_{n=0}^{N_g-1} n^2 \left\{ \begin{array}{l} \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} p(i, j) \\ |i - j| = n \end{array} \right\} \tag{6}$$

$$f_3 = \frac{\sum_{i=1}^{N_g} \sum_{j=1}^{N_g} [ijp(i, j) - \mu_x \mu_y]}{\sigma_x \sigma_y} \tag{7}$$

$$f_4 = \sum_i \sum_j (i - \mu_x)^2 p(i, j) \tag{8}$$

$$f_5 = \sum_i \sum_j \frac{1}{1 + (i - j)^2} p(i, j) \tag{9}$$

$$f_6 = \sum_{i=2}^{2N_g} iP_{x+y}(i) \tag{10}$$

$$f_7 = \sum_{i=2}^{2N_g} (i - f_8)^2 P_{x+y}(i) \tag{11}$$

$$f_8 = - \sum_{i=2}^{2N_g} P_{x+y}(i) \log\{P_{x+y}(i)\} \tag{12}$$

$$f_9 = - \sum_i \sum_j p(i, j) \log\{p(i, j)\} \tag{13}$$

$$f_{10} = \sum_{i=0}^{N_g-1} (i - f_8)^2 P_{x-y}(i) \tag{14}$$

$$f_{11} = - \sum_{i=0}^{N_g-1} P_{x-y}(i) \log\{P_{x-y}(i)\} \tag{15}$$

$$f_{12} = \frac{HXY - HXY1}{\max\{HX, HY\}} \tag{16}$$

$$f_{13} = (1 - e^{[-2(HXY2 - HXY)]})^{\frac{1}{2}} \tag{17}$$

$$f_{14} = (\text{Segundo eigenvalor mas grande de } Q)^{\frac{1}{2}} \tag{18}$$

donde $p(i, j)$ = define el valor de la matriz de coocurrencia en la posición (i, j) . N_g = Niveles de grises trabajados en la región segmentada de la imagen. $p_x(i)$ = i -ésima entrada en una matriz de probabilidad original de coocurrencia que es obtenida mediante la suma de filas en $p(i, j)$. $p_y(j)$ = j -ésima entrada en una matriz de probabilidad original de coocurrencia que es obtenida mediante la suma de columnas en $p(i, j)$. R = Número total de elementos en la matriz de coocurrencia. μ_x, μ_y = Media de P_x y P_y . σ_x, σ_y = Desviación estandar de P_x y P_y . Hx, HY = Entropía de P_x y P_y . En total se obtuvieron 14 descriptores texturales de cada imagen.

3.2.3. Características cromáticas

Las características cromáticas proveen información de la intensidad del color de una región segmentada. éstas características pueden ser calculadas por cada canal de intensidad, por ejemplo, rojo, verde, azul, escala de grises, tono (Hue), saturación (Saturation) e intensidad (Value), etc. Las características que se emplearon fueron; características de intensidad estandar, éstas describen la media, desviación estandar de la intensidad, primera y segunda derivada derivada en la región segmentada, Momentos de Hu con información de intensidad, características de Gabor basadas en funciones de gabor en 2D. En los experimentos realizados se obtuvieron 122 características por cada canal. Ya que los experimentos fueron realizados en RGB solo se utilizaron 366 características cromáticas.

3.3. Clasificadores

3.3.1. Máquinas de Vectores Soporte (SVM)

En los experimentos realizados se compararon los resultados con tres clasificadores, se utilizó un clasificador Bayesiano, el algoritmo Backpropagation y Máquinas de Vectores de Soporte (SVM). En este apartado solo se describe a las SVM por razones de espacio. Formalmente las SVM puede ser definidas de la siguiente manera:

Asumiendo que un conjunto de datos de entrenamiento X es dado como:

$$(\mathbf{x}_1, \mathbf{y}_1), (\mathbf{x}_2, \mathbf{y}_2), \dots, (\mathbf{x}_n, \mathbf{y}_n) \quad (19)$$

i.e. $X = \{x_i, y_i\}_{i=1}^n$ donde $x_i \in \mathbb{R}^d$ y $y_i \in \{+1, -1\}$. Entrenar una SVM permite resolver un problema de programación cuadrática como sigue:

$$\begin{aligned} & \underset{\alpha_i}{\text{máx}} -\frac{1}{2} \sum_{i,j=1}^l \alpha_i y_i \alpha_j y_j \mathbf{K} \langle x_i \cdot x_j \rangle + \sum_{i=1}^l \alpha_i \\ & \text{sujeto a: } \sum_{i=1}^l \alpha_i y_i = 0, \quad C \geq \alpha_i \geq 0, i = 1, 2, \dots, l \end{aligned} \quad (20)$$

donde $C > 0$, $\alpha_i = [\alpha_1, \alpha_2, \dots, \alpha_l]^T$, $\alpha_i \geq 0, i = 1, 2, \dots, l$, son coeficientes que corresponden a x_i, x_j con α_i diferentes a cero que son llamados Vectores Soporte(SV). La función \mathbf{K} es un función, que debe satisfacer las condiciones de Mercer.

Sea S el conjunto de SV obtenidos después del entrenamiento, entonces el hiperplano óptimo es dado por:

$$\sum_{i \in S} (\alpha_i y_i) K(\mathbf{x}_i, \mathbf{x}_j) + b = 0 \quad (21)$$

y la función de decisión óptima es definida como

$$f(x) = \text{sign} \left(\sum_{i \in S} (\alpha_i y_i) K(\mathbf{x}_i, \mathbf{x}_j) + b \right) \quad (22)$$

donde $\mathbf{x} = [x_1, x_2, \dots, x_l]$ son los datos de entrada, α_i y y_i son los multiplicadores de Lagrange. Un nuevo objeto x puede ser clasificado empleando (22). El Vector \mathbf{x}_i es dado en la forma de producto punto. Existe un multiplicador de Lagrange α para cada punto de entrenamiento. Cuando el máximo margen del hiperplano es encontrado, solamente los puntos más cercanos al hiperplano satisfacen $\alpha > 0$. Estos puntos son los $S V$.

4. Resultados Experimentales

En esta sección se muestran la técnica de selección de parámetros, normalización de datos y los resultados experimentales obtenidos con el sistema propuesto.

4.1. Conjunto de datos

En los experimentos realizados se utilizó el conjunto de datos ICL, que es una colección de hojas de la Universidad de Hefei. El conjunto de datos contiene 16849 imágenes de hojas de 220 especies.



Figura 2: Conjunto de hojas empleado

Con el objetivo de realizar una análisis comparativo de los clasificadores y las características extraídas separamos los conjuntos en dos tipos, triviales (cuyas formas pueden ser fácilmente distinguidas y diferenciadas) y complejas (cuyas formas son muy similares entre sí). A partir del conjunto de datos original se obtuvo un conjunto trivial y 11 conjuntos complejos.

Es decir al conjunto de hojas trivial que contiene hojas muy distintas entre sí, se le puede atribuir que dada la gran diferencia en características, el clasificador pueda distinguir entre una y otra hoja muy fácilmente. Por otro lado, en el conjunto de hojas complejo que contiene hojas muy similares entre sí, podría ser un reto para un clasificador distinguir entre una hoja y otras dado que sus características son muy similares. Sin embargo, es cierto que cada clasificador utiliza técnicas muy distintas que pueden tener variaciones en sus resultados. El proceso de separación de conjuntos, se realizó de forma manual de acuerdo a las semejanzas y diferencian notables a simple vista. Finalmente, el total de especies asociadas al conjunto trivial fue de 90 de las 220 con las que se contaba. La Figura 2 muestra un ejemplo de hoja de las 90 familias asociadas al conjunto trivial.

Para el conjunto de hojas complejas, se fueron formando subconjuntos dependiendo de las hojas que tenían cierta similitud. Finalmente se formaron 11 subconjuntos distintos con 169 especies. El subconjunto con menos especies fue de 3 y el subconjunto con más especies asociadas fue de 37.

Las Figuras 3 y 4 muestran cada subconjunto creado con algunos ejemplos de las familias de hojas que se tomaron para formar cada grupo, además se describe la razón de similitud que se tomó en cuenta para asignar cada hoja y se muestran 3 hojas

Subconjunto	Familias Asociadas	Ejemplos	Razon de Similitud
1	19		Orbicular
2	8		Lineal
3	14		Lanceolada
4	27		Elíptica
5	12		Aovada
6	3		Lacerada con forma de pentágono

Figura 3: Conjuntos de hojas utilizados

como ejemplo aunque, el total de cada subconjunto se describe en Familias Asociadas.

4.2. Normalización de datos

Todas las características extraídas fueron normalizadas con la relación:

$$f_{ij} = \frac{T_{ij} - \mu_j}{\sigma_j}$$

donde $i = 1, \dots, m$ y $j = 1, \dots, n$, μ_j y $\sigma_j T_{ij} = \sigma_j$ representan la media y desviación estándar de la j -ésima característica, T_{ij} representa la j -ésima característica del i -ésimo vector, m es el número de imágenes y n el número de características. Las características normalizadas tienen media cero y desviación estándar igual a 1.

4.3. Selección de parámetros

La selección de parámetros es un paso muy importante, ya que una buena selección de parámetros tiene un efecto considerable en el desempeño del clasificador.

En todos los clasificadores utilizados se obtuvieron los parámetros óptimos mediante validación cruzada y búsqueda de malla.

4.4. Resultados

En los experimentos realizados, todos los conjuntos de datos fueron normalizados y se utilizó validación cruzada con $k = 10$. La Tabla 1 muestra los resultados obtenidos con características geométricas, texturales y cromáticas, cada una como características individuales. CH_i define el conjunto de datos de hojas utilizado. Para cada clasificador utilizado, se reportan las precisiones obtenidas con cada conjunto individual de características. La métrica utilizada para evaluar el desempeño del

Subconjunto	Familias Asociadas	Ejemplos	Razon de Similitud
7	7		Lineal dentada
8	9		Espatulada
9	13		Aovada con cuspe
10	20		Elongada
11	37		Obovada

Figura 4: Conjuntos de hojas utilizados

clasificador fue precisión y esta se obtiene de los aciertos del clasificador entre el total del conjunto de datos.

En los resultados obtenidos, no es posible inferir que la similitud entre hojas afecte significativamente a los clasificadores, los desempeños de los clasificadores utilizados con imágenes muy similares entre sí y disimilares, no son contrastantes. Sin embargo, es posible apreciar que las características texturales son poco discriminativas para la mayoría de los conjuntos de datos, excepto para el conjunto CH₆. Una posible razón es que el tamaño del conjunto de datos es muy pequeño (solo tres clases).

Tabla 1. Desempeño con características cromáticas, texturales y geométricas.

	Cromáticas			Texturales			Geométricas		
	Bayes	BP	SVM	Bayes	BP	SMO	Bayes	BP	SVM
CH.1	88.7	94.214	94.913	36.283	47.785	56.902	81.893	93.566	95.408
CH.2	81.57	84.502	88.288	38.873	68.244	73.053	85.588	90.194	92.941
CH.3	88.22	94.736	95.035	37.180	74.067	77.518	78.088	89.320	91.918
CH.4	86.25	94.566	95.628	36.729	67.324	75.881	74.703	90.747	93.409
CH.5	93.67	95.560	95.727	28.303	71.042	72.841	85.444	95.111	97.289
CH.6	99.37	100	100	75.534	91.710	93.308	93.081	99.371	99.742
CH.7	96.56	96.704	97.457	48.256	82.396	85.082	97.875	98.752	98.733
CH.8	95.85	97.830	98.561	56.406	84.239	87.875	95.266	98.316	98.941
CH.9	88.38	90.648	90.872	38.393	65.776	69.69	84.592	93.807	94.410
CH.10	86.31	95.854	96.365	39.549	78.893	82.871	80.216	93.105	94.182
CH.11	77.62	92.333	93.912	33.721	63.418	68.161	74.832	88.356	91.324
CH.12	79.37	95.722	97.754	58.238	61.558	65.585	79.985	86.933	89.095

La Tabla 2 muestra los resultados obtenidos con combinaciones de características. En los resultados es posible apreciar una mejora en la precisión de clasificación en comparación con los desempeños obtenidos con características no combinadas. Utilizar únicamente características texturales no produce resultados satisfactorios. Sin embargo, cuando son combinadas las características texturales con geométricas o cromáticas, estas obtienen muy buenos desempeños.

Tabla 2. Desempeño con características cromáticas-texturales, cromáticas-geométricas y texturales-geométricas.

	Cromáticas-Texturales			Cromáticas-Geométricas			Texturales-Geométricas		
	Bayes	BP	SVM	Bayes	BP	SVM	Bayes	BP	SVM
CH_1	87.357	93.426	94.264	92.371	97.242	97.593	86.472	93.805	95.276
CH_2	82.061	84.732	86.253	87.058	90.588	92.476	88.122	91.869	91.421
CH_3	90.413	95.864	96.421	92.51	96.948	97.93	85.150	93.045	94.285
CH_4	87.261	94.005	95.218	91.423	97.262	95.825	84.332	92.370	94.720
CH_5	92.242	95.952	95.301	96.333	95.111	96.015	90.219	94.603	95.086
CH_6	98.742	100	100	98.742	100	100	96.226	99.371	99.371
CH_7	96.861	95.48	97.289	95.853	96.566	97.717	96.380	96.533	98.760
CH_8	95.052	96.135	96.211	96.844	96.211	97.823	96.614	96.697	96.958
CH_9	87.772	91.484	92.108	94.41	95.072	95.238	87.554	92.139	90.174
CH_10	87.748	95.926	95.332	89.77	95.296	96.163	90.505	92.628	91.939
CH_11	80.537	92.569	93.169	83.815	93.889	94.925	84.039	92.655	92.817
CH_12	81.0788	95.342	95.865	87.785	95.233	96.721	86.340	89.204	93.204

La Tabla 3 muestra los resultados obtenidos con todas las características (cromáticas, texturales y geométricas). Sin embargo, se puede notar que utilizar todas las características no precisamente infiere en una buena clasificación, ya que los resultados fueron mejores al utilizar solo características geométricas con cromáticas.

Tabla 3. Desempeño con características geométricas, texturales y cromáticas.

	Cromáticas-Texturales-Geométricas		
	Bayes	BP	SVM
CH_1	91.909	96.207	98.524
CH_2	87.739	90.038	94.941
CH_3	93.797	96.992	98.882
CH_4	90.940	96.253	97.183
CH_5	95.784	98.145	99.153
CH_6	98.742	100	100
CH_7	98.535	98.954	99.062
CH_8	96.354	98.177	99.828
CH_9	95.196	93.886	97.893
CH_10	90.027	98.292	98.531
CH_11	86.482	93.807	95.917
CH_12	88.488	96.19	98.526

5. Conclusiones

En este artículo presentamos un análisis comparativo de la influencia de las características en el desempeño de varios clasificadores. El sistema propuesto extrae un conjunto de características basadas en propiedades geométricas, texturales y cromáticas. En los resultados obtenidos se aprecia que una combinación de las diferentes características es necesaria para obtener una buena precisión de clasificación. Aunque no siempre es necesario incluir la combinación de todas las características, ya que algunas de estas afectan la precisión de clasificación. Por lo tanto se debe de hacer una buena combinación de estas. En este trabajo se demostró que para los conjuntos utilizados la mejor combinación son las características geométricas con las características cromáticas.

English Summary

Comparative Analysis of the Techniques Used in a Recognition System of Plant Leaves.

Abstract

The development of vision systems for identifying plants by leaves is an important challenge which has numerous applications ranging from food, medicine, industry and environment. Recently, several techniques have been proposed in the literature in order to identify plants in various fields of application. However, current techniques are restricted to the recognition and identification of plants using specific descriptors. In this paper, is accomplished a comparative analysis using different methods of feature extraction (textural, chromatic and geometric) and different methods of classification. The experiments are executed on very similar plants. Twelve sets of leaves with similar shape characteristics are studied using several classifiers. The performance of different combinations of classifiers-descriptors are analyzed in detail for each set. The results show that a combination of different feature extraction techniques is necessary in order to improve the performance. This combination of descriptors is more necessary when the leaves have similar characteristics.

Keywords:

Classification Descriptors SVM Data Sets

Agradecimientos

Este estudio fué financiado por la Secretaría de Investigación de la Universidad Autónoma del Estado de México con el proyecto de investigación 3771/2014/CIB.

Referencias

- [1] H. Muhammad Asraf, M.T. Nooritawati, M.S.B. Shah Rizam, A Comparative Study in Kernel-Based Support Vector Machine of Oil Palm Leaves Nutrient Disease, *Procedia Engineering*, Volume 41, 2012, Pages 1353-1359, ISSN 1877-7058.
- [2] Andreas BrandstÄdt, Van Bang Le, Structure and linear time recognition of 3-leaf powers, *Information Processing Letters*, Volume 98, Issue 4, 31 May 2006, Pages 133-138.
- [3] Guillaume Cerutti, Laure Tougne, Julien Mille, Antoine Vacavant, Didier Coquin, Understanding leaves in natural images – A model-based approach for tree species identification, *Computer Vision and Image Understanding*, Volume 117, Issue 10, October 2013, Pages 1482-1501, ISSN 1077-3142.
- [4] Chaki, J., Parekh, R. (2012). "Designing an automated system for plant leaf recognition" *International Journal of Advances in Engineering Technology*, 2(1), 149-158.
- [5] James S. Cope, David Corney, Jonathan Y. Clark, Paolo Remagnino, Paul Wilkin, Plant species identification using digital morphometrics: A review, *Expert Systems with Applications*, Volume 39, Issue 8, 15 June 2012, Pages 7562-7573.
- [6] Ji-Xiang Du, Xiao-Feng Wang, Guo-Jun Zhang, Leaf shape based plant species recognition, *Applied Mathematics and Computation*, Volume 185, Issue 2, 15 February 2007, Pages 883-893, ISSN 0096-3003.
- [7] Ji-xiang Du, Chuan-Min Zhai, Qing-Ping Wang, Recognition of plant leaf image based on fractal dimension features, *Neurocomputing*, Volume 116, 20 September 2013, Pages 150-156, ISSN 0925-2312.
- [8] Chih-Ying Gwo, Chia-Hung Wei, Yue Li, Rotary matching of edge features for leaf recognition, *Computers and Electronics in Agriculture*, Volume 91, February 2013, Pages 124-134, ISSN 0168-1699.
- [9] D.J. Hearn "Shape analysis for the automated identification of plants from images of leaves" *Taxon*, 58 (2009), pp. 934-954

- [10] M.K. Hu, Visual pattern recognition by moment invariants, *IRE Trans. Inform. Theory*, 8 (1962), pp. 179-187
- [11] Rui Hu, John Collomosse, A performance evaluation of gradient field HOG descriptor for sketch based image retrieval, *Computer Vision and Image Understanding*, Volume 117, Issue 7, July 2013, Pages 790-806, ISSN 1077-3142,
- [12] Z. Husin, A.Y.M. Shakaff, A.H.A. Aziz, R.S.M. Farook, M.N. Jaafar, U. Hashim, A. Harun, Embedded portable device for herb leaves recognition using image processing techniques and neural network algorithm, *Computers and Electronics in Agriculture*, Volume 89, November 2012, Pages 18-29, ISSN 0168-1699.
- [13] Kadir, A., Nugroho, L. E., Susanto, A., Santosa, P. I. (2012a). "Experiments of distance measurements in a foliage plant retrieval system", *International Journal of Signal Processing, Image Processing and Pattern Recognition*, 5, 256-263.
- [14] Kaur, G., Kaur, G. (2012). "Classification of biological species based on leaf architecture", *International Journal of Engineering Research and Development*, 1, 35-42.
- [15] Mónica G. Larese, Rafael Namías, Roque M. Craviotto, Miriam R. Arango, Carina Gallo, Pablo M. Granitto, Automatic classification of legumes using leaf vein image features, *Pattern Recognition*, Volume 47, Issue 1, January 2014, Pages 158-168, ISSN 0031-3203.
- [16] Mónica G. Larese, Ariel E. Bayá, Roque M. Craviotto, Miriam R. Arango, Carina Gallo, Pablo M. Granitto, Multiscale recognition of legume varieties based on leaf venation images, *Expert Systems with Applications*, Volume 41, Issue 10, August 2014, Pages 4638-4647, ISSN 0957-4174.
- [17] W. Ma, B. Manjunath, Texture features and learning similarity, in: *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)*, 1996, pp. 425-430.
- [18] B. Manjunath, W. Ma, Texture features for browsing and retrieval of image data, *IEEE Trans. Pattern Anal. Mach. Intell.* 18 (1996) 837-842.
- [19] Petr Novotný, Tomáš Suk, Leaf recognition of woody species in Central Europe, *Biosystems Engineering*, Volume 115, Issue 4, August 2013, Pages 444-452, ISSN 1537-5110.
- [20] N. Kumar, P.N. Belhumeur, A. Biswas, et al., Leafsnap: a computer vision system for automatic plant species identification, in: *Proc. ECCV* 2012, 2012, pp. 502-516
- [21] J.S. Park, T.-Y. Kim, "Shape-based image retrieval using invariant features", in: K. Aizawa, Y. Nakamura, S. Satoh, (Eds.), *Advances in Multimedia Information Processing-PCM 2004*, Berlin/Heidelberg Lecture Notes in Computer Science, 2004, pp. 146-153
- [22] Guillermo Sampallo. Reconocimiento de tipos de hojas. *Inteligencia Artificial. Revista Iberoamericana de Inteligencia Artificial*, vol. 7, núm. 21, 2003, pp. 55-62, Asociación Española para la Inteligencia Artificial España.
- [23] A.W.M. Smeulders, M. Worring, S. Santini, A. Gupta, R. Jain, Content-based image retrieval at the end of the early years, *IEEE Trans. Pattern Anal. Mach. Intell.* 22 (2000) 1349-1380.
- [24] C.S. Sumathi, A.V.S. Kumar, "Edge and texture fusion for plant leaf classification", *International Journal of Computer Science and Telecommunications*, 3 (2012), pp. 6-9
- [25] M. Tico, T. Haverinen, P. Kuosmanen, A method of color histogram creation for image retrieval, in: *Proceedings of the Nordic Signal Processing Symposium (NORSIG-2000)*, Kolmarden, Sweden, 2000, pp. 157-160.
- [26] C.C. Venters, D.M. Cooper, A Review of Content-based Image Retrieval Systems, Technical Report, Manchester Visualization Centre, Manchester Computing, University of Manchester, Manchester, UK, 2000.
- [27] Chunlei Xia, Jang-Myung Lee, Yan Li, Yoo-Han Song, Bu-Keun Chung, Tae-Soo Chon, Plant leaf detection using modified active shape models, *Biosystems Engineering*, Volume 116, Issue 1, September 2013, Pages 23-35, ISSN 1537-5110.
- [28] Ji-Xiang Du, Xiao-Feng Wang, Guo-Jun Zhang, Leaf shape based plant species recognition, *Applied Mathematics and Computation*, Volume 185, Issue 2, 15 February 2007, Pages 883-893, ISSN 0096-3003
- [29] Shanwen Zhang, Ying-Ke Lei, Modified locally linear discriminant embedding for plant leaf recognition, *Neurocomputing*, Volume 74, Issues 14–15, July 2011, Pages 2284-2290, ISSN 0925-2312.
- [30] Shanwen Zhang, Yingke Lei, Tianbao Dong, Xiao-Ping Zhang, Label propagation based supervised locality projection analysis for plant leaf classification, *Pattern Recognition*, Volume 46, Issue 7, July 2013, Pages 1891-1897, ISSN 0031-3203.

Mejorando la Clasificación en Conjuntos de Datos No-balanceados Utilizando un Algoritmo Genético para Selección de Atributos

Jesus Taltempa*
Posgrado e Investigación
UAEMEX-Textcoco
talte_203@hotmail.com

Irene Aguilar
Posgrado e Investigación
UAEMEX-Textcoco
ireneico@gmail.com

Alfonso Zarco
Posgrado e Investigación
UAEMEX-Textcoco
azarcox@hotmail.com

Joel Ayala
Posgrado e Investigación
UAEMEX-Textcoco
joelayala@yahoo.com.mx

Laura D. Jalili
Posgrado e Investigación
UAEMEX-Textcoco
jalili.anderi@yahoo.com.mx

Emmanuel Calderon
Posgrado e Investigación
UAEMEX-Textcoco
Emmanuel_Cal01@hotmail.com

ABSTRACT

The performance of classification algorithms in unbalanced data-sets is a difficult issue. Classical classification algorithms favor the majority class due to the representation of the minority class, noise and inability to expand the limits of the minority class. In recent years, sampling techniques to imbalanced data-sets have gained popularity to improve the performance of the classifiers in the identification of the minority class. However, sometimes the inherent noise in the dataset dimensionality prevents that the performance is improved significantly. In this research, a genetic algorithm is proposed to extract the most discriminating features and some popular techniques as under-sampling, over-sampling and SMOTE are used in conjunction with the proposed algorithm. The proposed technique reduces the noise introduced in the dataset dimensionality, improves the performance in the minority class and improves the total performance by reducing the bias and variance in the error. The obtained results using 14 datasets are compared. Experimental results show that the proposed technique obtains a significant performance improvement in all metrics tested.

Keywords

Classification, Data-sets, Imbalanced

Resumen

El desempeño de algoritmos de clasificación en conjuntos de datos no balanceados es una tarea difícil. Los algoritmos

*Todos los autores pertenecen a la Universidad Autónoma del Estado de México, Prolongación de Av. Zumpango s/n, Fraccionamiento El Tejocote, Textcoco, México, 52346.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

CICOM 2015 Cartagena, Colombia

Copyright 20XX ACM X-XXXXX-XX-X/XX/XX ...\$15.00.

comunes favorecen a la clase mayoritaria debido a la representación de la clase minoritaria, conjunto de datos con ruido e imposibilidad de expandir los límites de la clase minoritaria. En los últimos años, las técnicas de muestreo para conjuntos de datos no balanceados han ganado popularidad al mejorar el rendimiento de los clasificadores al mejorar el desempeño en la identificación de la clase minoritaria. Sin embargo, en algunas ocasiones el ruido inherente en la dimensionalidad del conjunto de datos evita que el desempeño sea mejorado significativamente. En este trabajo de investigación, se propone un algoritmo genético para extraer las características más discriminativas y se utilizan las técnicas de bajo-ejemplificado, sobre-ejemplificado y SMOTE en conjunto con el algoritmo. La técnica propuesta disminuye el ruido introducido en la dimensionalidad del conjunto de datos, mejora el desempeño en la predicción de la clase minoritaria, y el rendimiento de clasificación en general al reducir el sesgo y la varianza en el error. Se comparan los resultados utilizando 14 conjuntos de datos. Los resultados experimentales muestran que la técnica propuesta obtiene una mejora significativa del rendimiento en todas las métricas probadas.

Palabras clave:

Clasificación, Conjuntos de datos, Desbalance

Categories and Subject Descriptors

I.5.4 [Reconocimiento de Patrones]: Aplicaciones

General Terms

Algoritmos

1. INTRODUCCIÓN

En los últimos años, el aumento en el poder de procesamiento de los equipos de cómputo y capacidad de almacenamiento han permitido el almacenamiento de enormes cantidades de información. La búsqueda de información útil a partir de grandes cantidades de datos es una tarea difícil para los científicos y profesionales de la comunidad de aprendizaje automático. Entre estos desafíos, un problema muy importante en aprendizaje automático es el desarrollo

de algoritmos eficientes de clasificación en conjuntos de datos no-balanceados.

En conjuntos de datos no-balanceados binarios, una clase domina otra clase por un gran número de casos. La clase que contiene sustancialmente un gran número de casos se le conoce como clase mayoritaria y la clase con menos cantidad de casos se conoce como clase minoritaria. El desbalance en conjuntos de datos se presenta en una gran cantidad de casos de la vida real, por ejemplo, en detección de operaciones fraudulentas en un banco, detección de derrames de petróleo en imágenes de radar, búsqueda de células cancerosas entre las células no cancerosas, la búsqueda de documentos de interés en clasificación de texto, etc. [1], [2].

Los algoritmos de aprendizaje comunes generalmente obtienen mejores desempeños al identificar objetos que pertenecen a la clase mayoritaria. Este fenómeno se basa en una observación. Las funciones obtenidas para predecir la clase mayoritaria tienen tasas de error menores en comparación con los desempeños obtenidos para predecir la clase minoritaria. La principal razón de ello es debido a que los algoritmos de entrenamiento comunes aprenden una hipótesis sesgada, degradando el desempeño sobre la clase minoritaria.

Algunas razones importantes del bajo desempeño además del imbalance podría deberse a: pequeños subconjuntos disjuntos en el conjunto de datos, superposición o traslape en las clases, así como una mala selección de atributos. Estos problemas se consideran característicos de ruido en los datos. Los datos con ruido degradan el rendimiento de algoritmos de aprendizaje al sesgar los límites de decisión o sobreajuste del modelo mediante la incorporación de puntos de datos incorrectos.

Esta investigación está motivada por estas desventajas. El algoritmo propuesto utiliza un algoritmo genético para obtener los atributos que mejor ayudan al desempeño del clasificador. El algoritmo propuesto reduce el ruido introducido al clasificador al hacer una búsqueda de combinaciones de atributos. La técnica propuesta explota el poder de búsqueda de los algoritmos genéticos, esta búsqueda es guiada utilizando diferentes métricas de desempeño. Los resultados experimentales obtenidos muestran que el algoritmo propuesto puede ayudar a mejorar el desempeño de los clasificadores al ser combinado con otras técnicas para conjuntos de datos no-balanceados

2. ESTADO DEL ARTE

Debido a la naturaleza ubicua de los conjuntos de datos no-balanceados en muchos campos de aplicación sensibles, en los últimos años se han propuesto muchos enfoques en el estado del arte. Las técnicas de re-muestreo de datos son enfoques ampliamente utilizados en el aprendizaje de conjuntos no-balanceados [3] [4] [5] [6] [7] [8]. Las técnicas de re-muestreo de datos se pueden dividir en sub-muestreo y sobre-muestreo.

Las mayoría de las técnicas básicas de remuestreo son sub-muestreo y sobre-muestreo aleatorio. En sub-muestreo, las instancias de la clase de la mayoría se descartan al azar para conseguir una relación de clases balanceada. El principal inconveniente de esta técnica es la pérdida de información valiosa. Por otro lado, el sobre-muestreo añade duplica las instancias de la clase minoritaria hasta que ésta ambas posean el mismo tamaño. Sobre-muestreo es el método más simple para agregar instancias de la clase minoritaria. Algunos autores argumentan que esta técnica tiene un impacto

negativo en el desempeño de los clasificadores, sin embargo se han propuesto algunas técnicas para evitarlo. Sobre-muestreo sintético (SMOTE) es un enfoque que genera de forma inteligente las instancias de la clase minoritaria. La técnica minimiza satisfactoriamente los efectos negativos de sobremuestreo azar. Algunos autores han desarrollado técnicas de aprendizaje que utilizan diferentes costos o pesos para equilibrar el desbalance en las clases [9] [10] and [11]. A cada instancia en la clase minoritaria se le asigna un alto peso, mientras que a cada instancia en la clase mayoritaria se le asigna un bajo peso, ésta combinación hace más densa a la clase minoritaria. En [12], los autores utilizan un criterio de penalización para producir un efecto similar. Sin embargo, el principal inconveniente es que no todos los algoritmos de aprendizaje son lo suficientemente flexibles para incorporar pesos.

Por otro lado, los algoritmos genéticos (AG) son heurísticas bastante utilizadas en los últimos años. Los AG fueron originalmente introducidos en la literatura por John Holland en los 70's. Los AG son algoritmos de optimización muy populares basados en la mecánica de la selección natural y la evolución. En particular, son procedimientos de búsqueda heurística que modifican el valor de los individuos comúnmente codificados como cadenas binarias. éstos han sido aplicados profusamente en muchos campos como bioinformática [13] [14], predicción [15], finanzas [16], control de procesos bio-químicos [17], manufactura [18], vehículos autónomos [19], robótica [20], etc.

Concretamente en problemas de clasificación los genéticos han sido utilizados para obtener parámetros óptimos (tuning) [21], reducir la dimensionalidad del conjunto de datos [22], mejorar la precisión de clasificación [23] y hasta para generar conjuntos de reglas compactas a partir de datos de entrada [24].

Aunque los métodos de muestreo parecen dominar las investigaciones actuales en aprendizaje no-balanceado. Los enfoques basados en AG también han sido utilizados en aprendizaje no-balanceado. Algunos autores emplean genéticos para balancear los conjuntos de datos [25] [2]. En [26] los autores utilizan un AG para bajo-muestrear los conjuntos de datos, el algoritmo es utilizado para reducir el tamaño del conjunto de datos ayudando a una SVM a mejorar el tiempo de entrenamiento sin afectar su desempeño. En [27] los autores utilizan una SVM fuzzy para mejorar el desempeño del clasificador. El algoritmo permite reducir el efecto de datos con ruido y valores atípicos asignando diferentes valores con funciones de membresía fuzzy a las instancias basados en su importancia. En [28], se propone un sistema de clasificación para detectar las reglas más importantes y las reglas que perturban el comportamiento del clasificador afectando su desempeño. El sistema emplea reglas difusas jerárquicas y un AG para seleccionar los mejores datos y mejorar el desempeño en conjuntos no balanceados. Otros autores [1] utilizan AG para generar instancias artificiales agregando información nueva y mejorando el desempeño de los clasificadores al evolucionar los datos artificiales.

Aunque las técnicas basadas en AG mejoran el desempeño de los clasificadores no existen en la literatura técnicas que ayuden a seleccionar los mejores atributos para conjuntos de datos no-balanceados.

3. PRELIMINARES

En esta Sección se definen las técnicas básicas utilizadas

en el desarrollo del método propuesto.

3.1 Métricas de desempeño

Las métricas de desempeño clásicas no son adecuadas al trabajar en conjuntos de datos no balanceados. Utilizar únicamente la métrica de precisión puede llevarnos a conclusiones erróneas, ya que la clase minoritaria tiene un pequeño impacto en la precisión general en comparación con la clase mayoritaria. Por ejemplo, en un conjunto de datos con radio de imbalance de 99 a 1, un clasificador puede obtener 99% de precisión, aún cuando el clasificador clasifique incorrectamente a todos los datos de la clase minoritaria. Esta precisión es considerada muy buena en la mayoría de los casos. Sin embargo, para el caso de datos no-balanceados esta precisión no siempre es adecuada.

Con el propósito de evaluar adecuadamente y validar el desempeño un clasificador en conjuntos de datos no balanceados, es necesario utilizar diferentes métricas de desempeño. En este artículo empleamos F-measure y G-mean para evaluar el desempeño obtenido. G-mean es una métrica que mide cuan balanceadas se encuentran ambas clases y puede ser calculada mediante la ecuación:

$$G\text{-mean} = \sqrt{S_n^{true} + S_n^{false}} \quad (1)$$

donde S_n^{true} representa la sensibilidad del clasificador. La sensibilidad es la proporción de ejemplos positivos (ejemplos de clase +1) que han sido correctamente clasificados y es calculada como:

$$S_n^{true} = \frac{T_P}{T_P + F_N} \quad (2)$$

mientras que S_n^{false} representa la especificidad del clasificador. La especificidad define la proporción de ejemplos negativos (ejemplos de clase -1) que han sido correctamente pronosticados y es calculada como:

$$S_n^{false} = \frac{T_N}{T_N + F_P} \quad (3)$$

en las ecuaciones T_P representa el número de objetos de clase positiva (+1) que han sido pronosticados como clase +1, T_N representa el número de objetos de clase positiva (+1) que han sido pronosticados por el clasificador como clase -1, F_P representa el número de objetos de clase negativa (-1) que han sido pronosticados como clase +1, F_N representa el número de objetos de clase positiva (+1) que han sido pronosticados por el clasificador como clase -1.

Una medida que combina precisión y recall es la métrica F-measure:

$$F = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} \quad (4)$$

Aunado a estas métricas de desempeño, ROC (receiver operating characteristic) y las curvas ROC son ampliamente utilizados para analizar el desempeño de clasificadores binarios. La métrica ROC muestra la tasa de relación entre verdaderos positivos y falsos positivos [29]. Una curva ROC puede ser generada utilizando las etiquetas de los datos de entrada y las salidas del clasificador. La ventaja más importante de las curvas ROC es que permite una comparación visual de los clasificadores. Los beneficios numéricos y visuales asociados con este método le permiten una gran flexibilidad en el análisis de desempeño.

3.2 Clasificadores

La clasificación, consiste en detectar o reconocer un patrón en términos de propiedades o rasgos. El reconocimiento de patrones es una de las tareas más importantes. Sin embargo, también es una de las tareas más complejas. En la literatura actual, a pesar de que los clasificadores actuales se desempeñan muy bien sobre conjuntos de datos sin desbalance, éstos tienen bastantes problemas al trabajar en conjuntos de datos desbalanceados.

En esta Sección se define una SVM, que es el algoritmo con que se trabajo en esta investigación.

3.2.1 Clasificación basada en SVM

Las Maquinas de Vectores de Soporte (SVM por sus siglas en ingles, *Support Vector Machines*), son sistemas de aprendizaje que utilizan funciones lineales en espacios característicos. Una SVM mapea los puntos de entrada a un espacio de características altamente dimensional encontrando un hiperplano en esa dimensión que separe el conjunto de datos y maximice el margen m entre las clases. Las SVM encuentran el hiperplano óptimo utilizando el producto punto con funciones en el espacio de características que son llamados *Kernels*. La solución del hiperplano óptimo puede ser escrita como la combinación de unos pocos puntos de entrada que son llamados vectores de soporte. Básicamente una SVM se puede definir como sigue: Sea $z = \phi(x)$, el vector correspondiente en el espacio de características con un mapeo ϕ de \mathcal{R} a un espacio de características Z . Se desea minimizar la siguiente función:

$$\min \left\{ \frac{1}{2} w \cdot w + C \sum \xi_i \right\}$$

donde C es una constante y puede ser definida como un parámetro de regularización, mientras que ξ es una variable de holgura en las restricciones.

Cuando la dimensión del espacio de características para separar un conjunto de datos es muy grande y no se tiene ningún conocimiento de ϕ . Existe una propiedad efectiva de la SVM donde solo se necesita una función llamada *Kernel* (K), que calcula el producto punto de los puntos de entrada en el espacio de características Z , esto es:

$$Z_i Z_j = \phi(x_i) \phi(x_j) = K(x_i, x_j)$$

Esto permite realizar una separación de los datos en el espacio de características.

4. METODOLOGÍA PROPUESTA

El desempeño de los clasificadores en conjuntos con desbalance es un problema habitual. Actualmente se han implementado diversos algoritmos para enfrentar este problema, sin embargo no se ha investigado si la dimensionalidad del conjunto de datos afecta el desempeño de los clasificadores en conjuntos con desbalance. En esta investigación se implementa un algoritmo genético con el objetivo de optimizar la dimensionalidad del conjunto de datos con desbalance bajo una métrica de desempeño.

La dimensionalidad del conjunto de datos es un factor importante en el desempeño de los clasificadores. Algunos atributos pueden en ocasiones afectar el desempeño del clasificador en lugar de ayudarlo. El uso adecuado de atributos ayuda a mejorar el desempeño de un clasificador. Este problema ha sido tratado por varios autores [?] [8], este problema

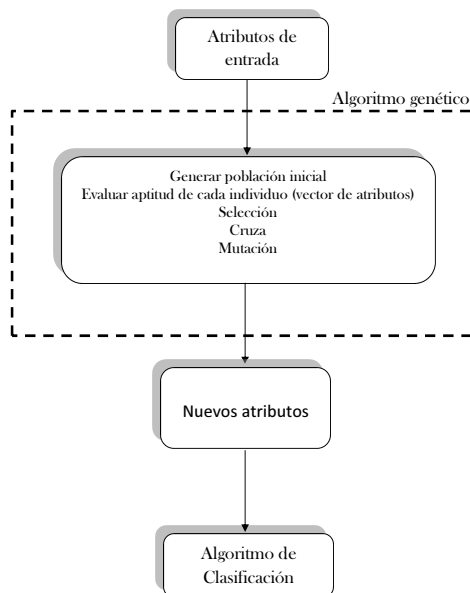


Figura 1: Diagrama de flujo del método propuesto

es habitual en reconocimiento de patrones y es comunmente llamado curso de la dimensionalidad. Un factor importante al momento de reducir características, es eliminar aquellas que no son importantes para el clasificador o encontrar la combinación de atributos que optimiza el desempeño del clasificador.

La selección de características o reducción de dimensionalidad es regularmente planteado como un problema de optimización. En los últimos años, los algoritmos genéticos (AGs) han sido extensamente utilizados para resolver problemas de optimización. Los AGs parten de la premisa de emplear la evolución natural como un procedimiento de optimización. Se caracterizan por representar las soluciones al problema que abordan en forma de cadenas binarias. Esas representaciones binarias les aporta características muy importantes de eficiencia. Sin embargo, es necesario disponer de un método para pasar esa representación binaria al espacio de búsqueda natural de cada problema.

Formalmente, dado un conjunto de patrones n -dimensional, la tarea del Algoritmo Genético es encontrar un conjunto de atributos en un espacio k -dimensional que maximice un criterio de optimización, donde $k \ll n$. Los patrones obtenidos son evaluados basados en dos condiciones la dimensionalidad del conjunto de datos y la separación entre clases o la precisión de clasificación. La Figura 1 muestra la metodología propuesta para resolver el problema de clasificación en conjuntos de datos con desbalance.

El esquema general de un algoritmo genético propuesto es el siguiente:

La estructura se describe con más detalle a continuación:

4.1 Población inicial

Para ejecutar un AG, se requiere de una población de individuos. Cada individuo, es un candidato a ser la solución del problema tratado que permite llegar a la solución.

Cada individuo de la población se representa con una cadena binaria. Los individuos de la población inicial sue-

Entrada:	Conjunto de datos X y clase t.
Salida:	Mejores atributos
1:	Crear población inicial
2:	Calcular aptitud de población inicial
3:	WHILE condición de paro no se cumpla Do
4:	Selección de individuos para la reproducción
5:	Cruza de individuos
6:	Mutación de individuos
7:	Calcular aptitud de nueva generación END

len ser cadenas de 0's y 1's generados de forma completamente aleatoria, esto es importante porque permite dotar al algoritmo genético de población con suficientemente variedad, para poder explorar todas las zonas del espacio de búsqueda. Las cadenas se denominan *genótipo* del individuo y que es análoga al *cromosoma* en el sistema biológico. Cada genótipo representa un punto x en el espacio de búsqueda del problema. A cada punto x se le denomina *fenótipo*. Se usa el termino *gen* para referirse a la codificación de una determinada característica del individuo. Cada *gen* puede tomar distintos valores que son llamados *alelos*. Para referirse a una determinada posición de la cadena binaria se usa el término locus.

En nuestro caso, el tamaño del cromosoma depende del número original de características que tiene el conjunto de datos. De acuerdo al número de características de cada conjunto de datos, se formó el tamaño de cada cadena binaria que se necesitó para implementar el algoritmo genético. La relación que existe entre cada cadena cromosómica con el conjunto de atributos, es que el 1 se toma como característica empleada y el 0 como ausencia de ese atributo. En la Figura 2 se muestra un ejemplo que muestra el genótipo y fenótipo las cadenas utilizadas.

4.2 Selección de individuos

La idea básica de selección, es utilizar una distribución de probabilidad de selección de una cadena, donde la probabilidad es directamente proporcional a la función de aptitud del individuo. La aptitud de cada individuo se toma de la precisión que se obtiene al clasificar el conjunto correspondiente con dicha cadena de atributos. Es decir, el proceso de selección debe favorecer la cantidad de copias de los individuos más adaptados. En este trabajo solo se utilizó la técnica de selección de ruleta.

4.3 Cruza

El proceso de cruza provee un mecanismo para heredar rasgos a su descendencia donde intervienen ambos padres, este es un método de fusión sobre la información genética de dos individuos. El mecanismo de cruza utilizado en los experimentos fue cruza de dos puntos.

4.4 Mutación

La mutación es un proceso donde el material genético puede ser alterado en forma aleatoria, debidamente a un error en la reproducción o la deformación de genes. A diferencia de la genética humana, la probabilidad en un algoritmo genético es mayor y evita que el algoritmo quede atrapado en mínimos locales. La forma más sencilla de mutación consiste en cambiar el valor de una de las posiciones de la cadena. Si el valor es cero pasa a uno, y si es uno pasa a cero.

<p>Cadena cromosómica completa</p> <p>1 1 1 1 1 1 1 1 1 1</p> <p>$\phi_1 \phi_2 \phi_3 \phi_4 \phi_5 \phi_6 \phi_7 \phi_8 \phi_9 \phi_{10}$</p> <p>0.4 0.2 0.71 0.83 0.91 0.31 0.25 0.01 0.3 0.17</p>	<p>Cadena cromosómica aleatoria</p> <p>1 1 0 1 1 1 0 1 0 0</p> <p>$\phi_1 \phi_2 \phi_4 \phi_5 \phi_6 \phi_8$</p> <p>0.4 0.2 0.83 0.91 0.31 0.01</p>
--	---

Figura 2: Generación de individuos

4.5 Condición de paro

Es necesario especificar las condiciones en las que el algoritmo deja de evolucionar y se presenta la mejor solución encontrada. La condición de paro más sencilla, se presenta al detectar que la mayor parte de la población ha convergido a una forma similar, careciendo de la suficiente diversidad para que tenga sentido continuar con la evolución. En el método propuesto, el algoritmo finaliza cuando después de 10 iteraciones no hubo mejora. La cadena con mejor precisión se almacena en un archivo de texto junto con la precisión obtenida.

4.6 Elitismo

De toda la población, se toma el individuo con mejor resultado de clasificación en la generación y este pasa intacto a la siguiente generación. Se continua con un método de selección y se realiza el método de cruce con los individuos seleccionados para la nueva generación. Se realiza el mismo procedimiento para la siguiente generación, y si ahora se obtiene un mejor resultado que el anterior, esté reemplaza al individuo que tenía la mejor precisión. Si no hubo cadena que mejore la aptitud, la cadena anterior vuelve a permanecer intacta en la nueva generación.

5. RESULTADOS EXPERIMENTALES

En esta sección se muestran la técnica de selección de parámetros, normalización de datos y los resultados experimentales obtenidos con el sistema propuesto.

5.1 Selección de parámetros

Seleccionar un conjunto de parámetros para el clasificador empleado es de gran importancia, ya que una buena selección de parámetros tiene un efecto considerable en el desempeño del clasificador. En todos los experimentos realizados empleamos funciones radiales base como kernel, que es definida como:

$$K(x_i - x_j) = \exp(-\gamma \|x_i - x_j\|^2), \gamma > 0$$

para obtener los parámetros óptimos se utilizó validación cruzada y búsqueda de malla. En los experimentos todos los conjuntos de datos fueron normalizados y se utilizó validación cruzada con $k = 10$. En todos los experimentos se realizaron 10 corridas y el promedio de estas es el que se reporta. Las métricas utilizadas para evaluar la SVM son F-measure y G-mean que fueron descritas en la Sección II.

5.2 Conjunto de datos

En esta investigación se utilizaron 18 conjuntos de datos, que son usualmente utilizados para probar algoritmos que utilizan conjuntos de datos con desbalance. El conjunto de datos puede ser obtenido del repositorio de KEEL data-set (Disponible en <http://sci2s.ugr.es/keel/datasets.php>).

La Tabla 1 muestra los conjuntos de datos y propiedades de ellos utilizados en los experimentos. Para cada conjunto

de datos, la Tabla muestra los ejemplos en la clase minoritaria (mc), número de ejemplos en clase mayoritaria (Mc), número de atributos (f) y el radio de desbalance. En los experimentos, todos los conjuntos de datos fueron normalizados utilizando la relación:

$$f_{ij} = \frac{T_{ij} - \mu_j}{\sigma_j}$$

donde $i = 1, \dots, m$ y $j = 1, \dots, n$, μ_j y $\sigma_j T_{ij} = \sigma_j$ representan la media y desviación estándar de la j -ésima característica, T_{ij} representa la j -ésima característica del i -ésimo vector, m es el número de imágenes y n el número de características. Las características normalizadas tienen media cero y desviación estándar igual a 1. La normalización de los datos permite al clasificador discriminar mejor cuando se tienen atributos con rangos muy grandes.

6. RESULTADOS EXPERIMENTALES

En esta Sección, se muestran las mejoras llevadas a cabo en una SVM utilizando el método propuesto. La utilidad de la metodología propuesta es comprobada al ser comparados los resultados con las implementaciones clásicas utilizadas en conjuntos de datos con desbalance.

Tabla 1: Conjuntos de datos con desbalance utilizados

Dataset	mc (+1)	Mc(-1)	Atributos (f)	Desbalance
liver_disorders	145	200	6	1:01.379
glass1	76	138	9	1:01.816
glass0	70	144	9	1:02.057
vehicle2	218	628	18	1:02.881
vehicle3	212	634	18	1:02.991
ecoli1	77	259	7	1:03.364
ecoli3	35	301	7	1:08.600
new-thyroid1	35	180	5	1:5.1428
new-thyroid2	35	180	5	1:5.1428
yeast4	51	1433	8	1:28.098
yeast6	35	1449	8	1:41.400
glass2	17	197	9	1:11.588
German	300	700	20	1:2.3333
Haberman	81	225	3	1:2.7777
Abalone	42	689	7	1:16.4047
Letter	789	19211	16	1:24.3485
Pima	268	500	8	1:1.8657
Shuttle	1706	2175	9	1:1.2749

El método propuesto se implemento en Matlab. En los experimentos llevados a cabo se utilizó validación cruzada con $k = 10$. Los resultados de los experimentos sobre conjuntos con desbalance son mostrados en la Tabla 2. En la Tabla, la primera columna indica el conjunto de datos, las otras columnas reportan las métricas AUC y G-mean obtenidas. En la Tabla σ representa la desviación estandar del método propuesto al ser evaluado en 10 corridas. Las precisiones obtenidas con las diferentes cadenas cromosómicas de cada corrida son promediadas y el resultados es el que se reporta en este trabajo, de la misma manera, la desviación estandar de las variaciones de desempeño de esas corridas es la que se reporta en este trabajo de investigación.

Tabla 2: Resultados de técnicas de bajo-ejemplificado, sobre-ejemplificado, SMOTE y Técnica propuesta

Dataset	Under-sampling		Over-sampling		SMOTE		PM		σ
	AUC	G	AUC	G	AUC	G	AUC	G	
liver_disorders	0.786	0.737	0.754	0.691	0.837	0.792	0.841	0.802	0.027
glass1	0.765	0.624	0.741	0.673	0.746	0.636	0.780	0.697	0.490
glass0	0.805	0.761	0.801	0.768	0.765	0.725	0.804	0.727	0.192
vehicle2	0.944	0.939	0.945	0.898	0.953	0.945	0.957	0.943	0.091
vehicle3	0.593	0.675	0.635	0.678	0.658	0.706	0.712	0.708	0.132
ecoli1	0.852	0.871	0.806	0.877	0.886	0.877	0.917	0.894	0.110
ecoli3	0.809	0.787	0.798	0.780	0.741	0.817	0.825	0.814	0.112
new-thyroid1	0.989	0.981	0.983	0.964	0.977	0.959	0.981	0.970	0.091
new-thyroid2	0.978	0.963	0.917	0.973	0.972	0.969	0.978	0.965	0.052
yeast4	0.793	0.781	0.786	0.729	0.791	0.761	0.815	0.794	0.137
yeast6	0.845	0.817	0.841	0.816	0.837	0.812	0.857	0.819	0.059
German	0.753	0.728	0.735	0.641	0.785	0.710	0.799	0.719	0.216
Haberman	0.683	0.632	0.652	0.600	0.689	0.634	0.729	0.692	0.038
Abalone	0.835	0.776	0.821	0.781	0.845	0.783	0.852	0.805	0.120
Letter	0.996	0.952	0.954	0.842	0.998	0.993	0.990	0.969	0.091
pima	0.696	0.725	0.647	0.718	0.714	0.735	0.735	0.736	0.203
glass2	0.607	0.639	0.624	0.652	0.674	0.725	0.713	0.739	0.079
shuttle	0.950	0.871	0.921	0.853	0.950	0.877	0.948	0.878	0.025

La Tabla 2 muestra que en la mayoría de los conjuntos de datos el desempeño del clasificador es mejorado al utilizar el método propuesto. Una razón de ello podría ser que algunos atributos introduzcan ruido al clasificador o que este ruido afecte más al clasificador en conjuntos de datos con desbalance.

Los resultados experimentales muestran que en algunos casos, optimizar el número de atributos de un conjunto de datos con desbalance puede ayudar más que utilizar los métodos clásicos. En todos los conjuntos de datos el método propuesto mejora significativamente el desempeño. Estos resultados permiten resaltar la utilidad del método propuesto.

La dimensionalidad de los conjuntos de datos fue reducida como lo muestra la Tabla 3. Se puede observar que en algunos subconjuntos el tamaño de la dimensión reducida es diferente aún cuando son conjuntos similares (glass1, glass2 y glass0, ecoli1 y ecoli3, new-thyroid1 y new-thyroid2).

Tabla 3: Resultados de técnicas de bajo-ejeuestra

Dataset	Features (f)	R
liver_disorders	6	4
glass1	9	6
glass0	9	5
vehicle2	18	14
vehicle3	18	12
ecoli1	7	5
ecoli3	7	4
new-thyroid1	5	4
new-thyroid2	5	4
yeast4	8	6
yeast6	8	6
glass2	9	6
German	20	16
Haberman	3	2
Abalone	7	5
Letter	16	12
Pima	8	6
Shuttle	9	5

7. CONCLUSIONES

Los métodos de clasificación actuales obtienen buenos resultados cuando estos son utilizados en conjuntos de datos balanceados. Sin embargo, para el caso de conjuntos de datos con desbalance, la mayoría de los clasificadores no obtienen resultados aceptables debido a que las fronteras de decisión son calculadas sin importar la diferencia de datos en las clases. En el estado de arte actual, una gran cantidad de algoritmos han sido desarrollados con el objetivo de enfrentar esta desventaja. Los algoritmos desarrollados han mejorando el desempeño de los clasificadores utilizando varias métricas de desempeño. Sin embargo, en estos algoritmos no ha sido tomada en cuenta la influencia de los atributos sobre el desempeño del clasificador.

En este artículo, se propone un método para mejorar el desempeño de clasificadores sobre conjuntos de datos no balanceados. El método propuesto ayuda a mejorar el desempeño del clasificador al eliminar atributos que describen a cada patrón que introducen ruido al clasificador. El método propuesto utiliza un algoritmo genético para eliminar atributos utilizando como aptitud una métrica de desempeño especial. El método propuesto realiza una búsqueda de los mejores atributos y/o combinación de mejores atributos, eliminando aquellos atributos que afectan su desempeño. Los experimentos obtenidos muestran que el método propuesto genera notables resultados al eliminar atributos que no aportan información debido al desbalance en el conjunto. La principal ventaja del método propuesto es su facilidad de implementación y su fácil uso sobre conjuntos de datos de pequeño y mediano tamaño.

8. REFERENCIAS

- [1] J. Cervantes, X. Li and W. Yu, "Imbalanced data classification via support vector machines and genetic algorithms", *Connection Science* 26 (4), 335-348, 2014.
- [2] S. Zou, Y. Huang, Y. Wang, J. Wang, and C. Zhou, "SVM learning from imbalanced data by ga sampling for protein domain prediction," in *Proceedings of the 2008 The 9th International Conference for Young Computer Scientists*, ser. ICYCS 08. Washington, DC, USA: IEEE Computer Society, 2008, pp. 982-987.
- [3] R. Batuwita and V. Palade, "Class Imbalance Learning Methods for Support Vector Machines", In *Imbalanced Learning: Foundations, Algorithms and Applications*, Haibo He and Yunqian Ma Ma (Eds.), Wiley, 2013
- [4] M.A.H. Farquard, Indranil Bose, Preprocessing unbalanced data using support vector machine, *Decision Support Systems*, Vol 53 (1), pp. 226-233, 2012.
- [5] José Hernández Santiago, Jair Cervantes, Asdrúbal López Chau, Farid García-Lamont: "Enhancing the Performance of SVM on Skewed Data Sets by Exciting Support Vectors". *IBERAMIA 2012*: 101-110
- [6] Jair Cervantes, Xiaoou Li, Wen Yu, "Using Genetic Algorithm to Improve Classification Accuracy on Imbalanced Data". *SMC 2013*: pp. 2659-2664, 2013
- [7] Xiong, N., "A hybrid approach to input selection for complex processes," *Systems, Man and Cybernetics, Part A: Systems and Humans*, IEEE Transactions on , vol.32, no.4, pp.532,536, Jul 2002
- [8] Chih-Fong Tsai, William Eberle, and Chi-Yuan Chu. 2013. Genetic algorithms in feature and instance

- selection. *Know.-Based Syst.* 39 (February 2013), 240-247.
- [9] H. M. Nguyen, E. W. Cooper, and K. Kamei, "Borderline over-sampling for imbalanced data classification," *Int. J. Knowl. Eng. Soft Data Paradigm.*, vol. 3, no. 1, pp. 4-21, Apr. 2011.
- [10] S. Hu, Y. Liang, L. Ma, Y. He, MSMOTE: improving classification performance when training data is imbalanced, in: *Proceeding 2nd International Workshop Computer Science Engineering*, vol. 2, pp. 13-17, 2009.
- [11] Guo, H. & Viktor, H. L. Learning from Imbalanced Data Sets with Boosting and Data Generation: The DataBoost-IM Approach SIGKDD Explor. Newsl., ACM, 6, pp. 30-39, 2004
- [12] K. Veropoulos, C. Campbell, and N. Cristianini, "Controlling the sensitivity of support vector machines," in *Proceedings of the International Joint Conference on AI*, pp. 55-60, 1999.
- [13] Jair Cervantes, De-Shuang Huang, Xiaou Li, Wen Yu, "A New Approach to Detect Splice-Sites Based on Support Vector Machines and a Genetic Algorithm". CIARP (2) 2013, pp. 342-349, 2013.
- [14] Jair Cervantes, De-Shuang Huang, Farid García-Lamont, Asdrúbal López Chau, "A Hybrid Algorithm to Improve the Accuracy of Support Vector Machines on Skewed Data-Sets". ICIC (1) 2014: 782-788, 2014.
- [15] Ch. Mizas, G.Ch. Sirakoulis, V. Mardiris, I. Karafyllidis, N. Glykos, R. Sandaltzopoulos, Reconstruction of DNA sequences using genetic algorithms and cellular automata: Towards mutation prediction?, *Biosystems*, Volume 92, Issue 1, pp 61-68, April 2008.
- [16] F. Mokhtab Rafiei, S.M. Manzari, S. Bostanian, Financial health prediction models using artificial neural networks, genetic algorithm and multivariate discriminant analysis: Iranian evidence, *Expert Systems with Applications*, Volume 38, Issue 8, pp 10210-10217, August 2011.
- [17] Antonio C. Caputo, Pacifico M. Pelagagge, Mario Palumbo, Economic optimization of industrial safety measures using genetic algorithms, *Journal of Loss Prevention in the Process Industries*, Vol. 24, Issue 5, pp 541-551, September 2011.
- [18] Rui Zhang, Pei-Chann Chang, Cheng Wu, A hybrid genetic algorithm for the job shop scheduling problem with practical considerations for manufacturing costs: Investigations motivated by vehicle production, *International Journal of Production Economics*, Available online 8 November 2012, ISSN 0925-5273.
- [19] Jeremy Breen, P. de Souza, G.P. Timms, R. Ollington, Onboard assessment of XRF spectra using genetic algorithms for decision making on an autonomous underwater vehicle, *Nuclear Instruments and Methods in Physics Research Section B: Beam Interactions with Materials and Atoms*, Vol. 269, Issue 12, pp 1341-1345, 15 June 2011.
- [20] Raşit Köker, A genetic algorithm approach to a neural-network-based inverse kinematics solution of robotic manipulators based on error minimization, *Information Sciences*, Vol 222, pp 528-543, 10 February 2013.
- [21] K. Asan Mohideen, G. Saravanakumar, K. Valarmathi, D. Devaraj, T.K. Radhakrishnan, Real-coded Genetic Algorithm for system identification and tuning of a modified Model Reference Adaptive Controller for a hybrid tank system, *Applied Mathematical Modelling*, Vol 37, Issue 6, pp 3829-3847, 15 March 2013.
- [22] Amelia Zafra, Mykola Pechenizkiy, Sebastián Ventura, HyDR-MI: A hybrid algorithm to reduce dimensionality in multiple instance learning, *Information Sciences*, Volume 222, Pages 282-301, 10 February 2013.
- [23] Jui-Sheng Chou, Min-Yuan Cheng, Yu-Wei Wu, Improving classification accuracy of project dispute resolution using hybrid artificial intelligence and support vector machine models, *Expert Systems with Applications*, Volume 40, Issue 6, pp 2263-2274, May 2013.
- [24] Bikash Kanti Sarkar, Shib Sankar Sana, Kripasindhu Chaudhuri, Selecting informative rules with parallel genetic algorithm in classification problem, *Applied Mathematics and Computation*, Volume 218, Issue 7, Pages 3247-3264, 1 December 2011.
- [25] Yakoub Bazi and Farid Melgani, Semisupervised PSO-SVM Regression for Biophysical Parameter Estimation, *IEEE Transactions on Geoscience and Remote Sensing*, Vol. 45, No. 6, June 2007.
- [26] Choi, Jong Myong, "A Selective Sampling Method for Imbalanced Data Learning on Support Vector Machines" (2010). PhD Dissertation. Iowa State University.
- [27] Batuwita, R.; Palade, V., "FSVM-CIL: Fuzzy Support Vector Machines for Class Imbalance Learning," *Fuzzy Systems*, *IEEE Transactions on* , vol.18, no.3, pp.558-571, June 2010
- [28] A. Fernandez, M. Jesus, and F. Herrera, "Improving the performance of fuzzy rule based classification systems for highly imbalanced datasets using an evolutionary adaptive inference system," in *Bio-Inspired Systems: Computational and Ambient Intelligence*. Berlin, Germany: Springer-Verlag, pp. 294-301, 2009.
- [29] T. Fawcett, "An introduction to roc analysis," *Pattern Recognition Letters.*, vol. 27, no. 8, pp. 861-874, Jun. 2006.
- [30] Mario Koppen. The Curse of Dimensionality. 5th Online World Conference on Soft Computing in Industrial Applications (WSC5)., September 4-18 2000.