



**UNIVERSIDAD AUTÓNOMA DEL ESTADO DE MÉXICO**  
**CENTRO UNIVERSITARIO UAEM TEXCOCO**

---

---

“IDENTIFICACIÓN AUTOMÁTICA DE HOJAS UTILIZANDO UN CLASIFICADOR  
BAYESIANO”

## TESIS

QUE PARA OBTENER EL TÍTULO DE  
LICENCIADO EN INGENIERIA EN COMPUTACIÓN

PRESENTA:

LIZBET ALEJANDRA DELGADO VALVERDE

DIRECTOR:

JAIR CERVANTES CANALES

REVISORES:

Dr. EN C. FARID GARCÍA LAMONT  
Dr. EN C. JOSÉ SERGIO RUIZ CASTILLA

TEXCOCO, ESTADO DE MÉXICO AGOSTO 2017

Universidad Autónoma del Estado de México

Centro Universitario UAEM Texcoco

Ingeniería en Computación

“IDENTIFICACIÓN DE HOJAS UTILIZANDO UN CLASIFICADOR BAYESIANO”

LizbetAlejandra Delgado Valverde

Agosto 2017

## Contenido

•	Introducción .....	6
1.1	Planteamiento del problema .....	7
1.2	Justificación.....	7
1.3	Objetivos y alcances del proyecto .....	8
1.3.1	Objetivo General .....	8
1.3.2	Objetivos Específicos .....	8
1.4	Metodología científica.....	8
2	Técnicas de Segmentación .....	9
2.1	Frontera Global .....	10
2.2	Segmentación adaptativa.....	11
2.3	Segmentación por clustering .....	12
2.4	Segmentación Otsu.....	13
2.5	Técnicas de dilatación y erosión .....	14
2.6	Técnicas para eliminar ruido en imágenes binarias.....	14
3	Técnicas de Extracción de características .....	16
3.1	Altura y ancho .....	16
3.2	Área .....	16
3.3	Perímetro .....	17
3.4	Rectangularidad.....	17
3.5	Inercia.....	17
3.6	Solidez.....	18
3.7	Excentricidad.....	18
3.8	Compacidad.....	18
3.9	Número de Euler .....	199
3.10	Proyección.....	20
3.11	Centro de gravedad o centroide.....	20
3.12	Excentricidad: Método de ejes principales.....	20

3.13	Varianza de elipse .....	222
3.14	Momentos frontera .....	22
3.15	Momentos invariantes de Hu.....	233
3.15.1	Momentos ordinarios.....	23
3.15.2	Momentos centralizados.....	244
3.15.3	Momentos normalizados .....	244
3.16	Momentos R .....	255
3.17	Momentos Flusser .....	255
3.18	Descriptores de Fourier .....	266
4	Técnicas de clasificación.....	277
4.1	Clasificador Bayesiano.....	277
4.2	Redes neuronales artificiales .....	28
4.3	SVM.....	31
4.4	Árboles de decisión .....	35
5	Metodología .....	388
5.1	Conjunto de datos utilizados .....	388
5.2	Pre procesamiento .....	399
5.3	Segmentación .....	40
5.4	Extracción de características .....	411
5.5	Clasificación e identificación .....	422
6	Resultados Experimentales .....	433
7	Discusión y conclusiones .....	466
	Glosario.....	477
	Referencias.....	488
8	Bibliografía .....	<b>¡Error! Marcador no definido.</b>

## Resumen

La identificación automática de plantas es un reto actual muy importante. El desarrollo de sistemas que identifiquen de forma automática las plantas impactará en amplias y distintas áreas como son: la alimentación, medicina, industria, medio ambiente, aduanas. En la literatura actual, varias técnicas han sido propuestas con el objetivo de identificar plantas en diversos campos de aplicación. En este proyecto, se desarrollará un sistema de reconocimiento que permita identificar plantas a partir de imágenes en una base de datos con imágenes con múltiples clases de plantas. Debido a la cantidad de imágenes se requieren descriptores de diferentes tipos que ayuden a aumentar el margen de discriminación entre plantas y mejorar así el desempeño del clasificador. Sin embargo, no solo es importante mejorar el desempeño de los clasificadores, una parte fundamental de los clasificadores es el tiempo de respuesta. Muchos campos de aplicación actuales requieren que la respuesta sea casi inmediata.

El presente trabajo propone una solución para la identificación de plantas a partir de sus hojas utilizando un clasificador Bayesiano. Los algoritmos existentes en el estado del arte para la identificación de plantas a partir de hojas utilizan diferentes técnicas de aprendizaje máquina que van desde redes neuronales, Máquinas de Soporte Vectorial, Árboles de decisión hasta clasificadores bayesianos. En este caso, se utilizó un clasificador bayesiano por su velocidad de respuesta y precisión. La presente tesis muestra un análisis comparativo de los tiempos de respuesta del clasificador Bayesiano en comparación con algunos clasificadores del estado del arte actual.

**Palabras clave:** Reconocimiento de hojas, procesamiento digital de imágenes, extracción de características.

## • Introducción

En los últimos años, los algoritmos de visión han sido implementados en diversos campos de aplicación. 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 las plantas para cuidado y protección de pesticidas, etc.

Existen muchas características foliares que los botánicos emplean para identificar las plantas. Sin embargo, en reconocimiento de patrones los métodos de identificación pueden restringirse a descriptores geométricos, texturales, cromáticos y de venación. Uno de los enfoques más comunes empleados consiste en analizar la forma de la hoja extrayendo características como tamaño, elongación, elipsidad, (Ji-Xiang Du X.-F. W.-J., 2007)(Chih-Ying Gwo, 2013), (Rui Hu, 2013), (Petr Novotný, 2013), otros autores consideran cromaticidad, textura de la hoja(Shanwen Zhang, 2011),(Guillaume Cerutti, 2013),una combinación de las anteriores (N. Kumar, 2012),(Jair Cervantes, 2017)e incluso la forma de la venación de las hojas (Ji-Xiang Du C.-M. Z.-P., 2013), (Mónica G. Larese R. N., 2014), (Mónica G. Larese A. E., 2014).



Figura 1. Conjunto de hojas utilizado en los experimentos.

Como se dijo anteriormente, existe una gran cantidad de información en la literatura sobre identificación de plantas y/o arboles a partir de hojas. Sin embargo, es fácil identificar un problema actual en los algoritmos de reconocimiento de hojas. Los estudios muestran algoritmos con diferentes técnicas de clasificación que en ocasiones son muy complejos y en otras no es posible utilizarlos en algunos dispositivos debido a su complejidad computacional o se requieren dispositivos con capacidades altas para implementarlos.

La motivación de esta tesis es evaluar el tiempo de reconocimiento de los clasificadores Bayesianos y compararlo con las técnicas más utilizadas en los algoritmos actuales. En los experimentos realizados se presenta un análisis comparativo de tiempos con: Máquinas de Vectores de Soporte (SVM) por sus siglas en inglés, son una técnica de clasificación binaria que en los últimos años han obtenido excelentes resultados en problemas de clasificación. Redes neuronales que en los últimos años han sido extensamente utilizadas con resultados prometedores en múltiples áreas de investigación. Árboles de decisión que es una técnica muy empleada, pero ha perdido uso últimamente debido al auge de las redes neuronales y SVM. Para la selección de las características utilizadas se basa esta investigación en algunos trabajos previos (Ji-Xiang Du X.-F. W.-J., 2007),(Rui Hu, 2013), (Petr Novotný, 2013), (Jair Cervantes, 2017), eliminando muchas características con bajo poder discriminativo o que introducen ruido al clasificador.

## **1.1 Planteamiento del problema**

El tiempo de procesamiento y reconocimiento automático de hojas es afectado debido a varios factores como pre-procesamiento, segmentación, extracción de características y técnica de clasificación. Los procesos más complejos computacionalmente hablando son extracción de características y clasificación, sin embargo, extracción de características es un proceso que no se puede evitar cuando se han identificado las características o descriptores discriminativos que se deben obtener. Por otro lado, en el caso de las técnicas de clasificación es posible no utilizar alguna técnica que sea muy compleja y utilizar una con menor complejidad computacional y que obtenga casi el mismo desempeño. No obstante, no existe un estudio que muestre los tiempos de reconocimiento que conllevan las diferentes etapas de un sistema de reconocimiento. En esta tesis se evalúa el desempeño (tiempo y precisión) de varios clasificadores actuales en comparación con clasificadores bayesianos.

## **1.2 Justificación**

Anteriormente se ha mencionado que se han realizado múltiples estudios para reconocimiento automático de plantas y que es un tema actual y recurrente en múltiples campos de investigación; por esta razón se ha trabajado por varios años desarrollando y mejorando los procesos y algoritmos que contribuyan a efectuar esta tarea de una manera más eficiente. El resultado de la investigación aportará conocimiento sobre la capacidad

discriminativa y tiempos de procesamiento de los clasificadores Bayesianos en comparación con otros clasificadores actuales en la identificación de plantas a partir de características.

### **1.3 Objetivos y alcances del proyecto**

#### **1.3.1 Objetivo General**

El objetivo general de la tesis es obtener un análisis comparativo de los tiempos de reconocimiento que ocupan varios sistemas para identificar hojas utilizando clasificación Bayesiana en comparación con otras técnicas de clasificación.

#### **1.3.2 Objetivos Específicos**

- Desarrollar un sistema que realice el proceso completo de identificación de plantas utilizando clasificación Bayesiana.
- Obtener un análisis comparativo de varios clasificadores con respecto al tiempo de respuesta.

### **1.4 Metodología científica**

Para llevar a buen término la investigación se llevarán a cabo los siguientes pasos:

- Estudiar las técnicas más recientes de segmentación y extracción de características que ayuden a mejorar la precisión de clasificación.
- Obtener las imágenes de hojas de plantas bajo las condiciones requeridas para conformar la base de datos con la que se trabajará.
- Probar los patrones obtenidos con los algoritmos de clasificación mencionados
- Determinar los parámetros óptimos de funcionamiento para los algoritmos.
- Realizar las comparaciones de los algoritmos mencionados.



## 2 Técnicas 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 relevantes. El objetivo de este Capítulo es darle al lector una introducción a los métodos de segmentación utilizados en el sistema propuesto.

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 todos los tipos de imágenes. En esta Sección se describen las técnicas clásicas más utilizadas en segmentación de imágenes.

Los métodos de valor umbral pertenecen a los métodos más antiguos de tratamiento de imágenes digitales. Los métodos de valor umbral son especialmente sencillos, permitiendo implementarlos rápidamente y obtener resultados con relativamente poco esfuerzo, esto ha contribuido a su aceptación. No obstante, la calidad de la segmentación suele ser mala en comparación con otros métodos más sofisticados y más costosos computacionalmente hablando.

Los métodos del valor umbral son un grupo de algoritmos cuya finalidad es segmentar gráficos, es decir separar los objetos de una imagen que nos interesen del resto. Con la ayuda de los métodos de valor umbral en las situaciones más sencillas se puede decidir qué píxeles conforman los objetos que buscamos y qué píxeles son sólo el entorno de estos objetos. Este método es especialmente útil para separar el texto de un documento del fondo de la imagen (papel amarillento, con manchas y arruguitas, por ejemplo).

Esto es especialmente útil si queremos digitalizar libros antiguos, en los que el contraste entre el texto (que ya ha perdido parte de sus pigmentos) y el papel (oscurecido y manoseado) no es demasiado elevado. Como con todos los métodos de segmentación se trata de asignar cada píxel a un cierto grupo, llamado comúnmente "segmento". La imagen está compuesta por valores numéricos (uno o más valores de color para cada píxel). La pertenencia de un píxel a un cierto segmento se decide mediante la comparación de su nivel de gris (u otro valor unidimensional) con un cierto valor umbral. El nivel de gris de un píxel equivale a su nivel de luminosidad; el resto de la información sobre el color no se tiene en cuenta. Dado que esta comparación de valores se realiza individualmente para cada píxel, al método del valor umbral se le considera un método de segmentación orientado a píxeles

## 2.1 Frontera Global

Se define como frontera global a una técnica de umbralización o *thresholding*. Esta es una técnica de segmentación rápida cuyo coste computacional es relativamente bajo. La obtención del umbral está considerada bajo diversos criterios, ya que puede ser manejado por el mismo usuario, donde se establece un punto de umbralización o se obtiene de forma automática. Para obtener el umbral de manera automática es posible hacerlo de dos formas, encontrando el umbral con la suma del pixel con la intensidad más alta con el pixel de intensidad más baja y dividirlo entre dos, como se muestra en las ecuaciones (2.1) y (2.2).

$$th = \frac{\max(I(x,y)) - \min(I(x,y))}{2} \quad (2.1)$$

$$th = \sum_i \sum_j \frac{f(i,j)}{M \times N} \quad (2.2)$$

El otro modo es mediante la normalización del histograma de una imagen  $I$  de tamaño  $M \times N$ , esta técnica es sencilla porque al obtener el histograma normalizado, que como bien se sabe es la sumatoria de las probabilidades de existencia de cada pixel, no considera información espacial sino solamente la distribución de grises en la imagen, la ecuación (2.2) muestra cómo obtener el histograma normalizado. En pocas palabras, la umbralización permite convertir una imagen de niveles de gris en una imagen binaria, esto por medio de la ecuación (2.2), donde si cumple la condición de *thresholding* umbral, la salida será 1, en caso contrario 0.

$$G(x,y) = \begin{cases} 0 & \text{si } 0 \leq f(x,y) \leq th \\ 1 & \text{si } th < f(x,y) \leq 255 \end{cases} \quad (2.3)$$

donde  $th$  representa el valor del umbral asignado.

El uso del histograma es importante debido a que es mucho más fácil encontrar el punto de umbralización más adecuado, esto permite visualizar diversas características de la misma imagen evitando de esta forma la pérdida de información que podría llegar a ser importante.

## 2.2 Segmentación adaptativa

La frontera adaptativa al igual que la frontera global, están basadas en un umbral para la obtención de aquellos pixeles necesarios para la obtención de alguna característica en común, ambos devuelven una imagen binarizada, pero la diferencia que existe entre ambos es el algoritmo para la obtención del umbral. El algoritmo aquí presente muestra cómo obtener el umbral por este método.

1. Se comienza seleccionando una frontera inicial  $th(i)$ , donde  $i=1$ . La frontera inicial puede ser cualquiera, en nuestro caso se dispuso  $unth(1) = 127$ . Este umbral puede ser de igual forma obtenido con la ecuación (2.4).
2. Se agrupan los pixeles basándonos en  $th$  creando así dos grupos. Donde el primer grupo es para todos aquellos pixeles con intensidades mayores o iguales a  $th$  y el segundo para los pixeles con intensidades menores a  $th$ . La ecuación (2.4) y la ecuación (2.5) muestran el agrupamiento de pixeles.

$$G_1 = \{0, \dots, th\} \quad (2.4)$$

$$G_2 = \{th + 1, th + 2, \dots, L\} \quad (2.5)$$

donde  $\Lambda$  = Intensidad máxima presente en la imagen.

3. Se obtiene la media de cada grupo basados en si son mayores a  $th$  o menores al mismo. La ecuación (2.6) y la ecuación (2.7) muestran cómo obtener la media de cada grupo.

$$\mu_1 = \sum_{i=0}^{th} \frac{G_1(i)}{th} \quad (2.6)$$

$$\mu_2 = \sum_{i=th+1}^L \frac{G_2(i)}{L} \quad (2.7)$$

4. Una vez obtenida la media por ambos grupos, se actualiza la frontera  $th(i)$  aplicando la ecuación (2.8).

$$th(i) = \frac{\mu_1 + \mu_2}{2} \quad (2.8)$$

5. Finalmente, este proceso se realiza hasta que  $th(i)$  no se llegue a modificar con respecto al resultado de dos iteraciones anteriores a la misma, en caso contrario se siguen ejecutando todos los pasos. Una vez obtenido el umbral se aplica la misma condición mostrada en la ecuación (2.8).

### 2.3 Segmentación por clustering

Dentro de las técnicas de segmentación, k-mean es una de las más utilizadas por su baja complejidad y robustez en segmentación binaria. K-means es un método de agrupamiento, que tiene como objetivo la partición de un conjunto de  $n$  elementos en  $k$  grupos en el que cada elemento pertenece al grupo más cercano a la media calculada. Es generalmente un método utilizado en minería de datos para calcular la distribución de un conjunto de elementos que se tenga, en base a la distancia de cada elemento al número de centroides que se tengan. La figura 2 muestra un ejemplo de la aplicación del algoritmo K-means aplicado a agrupamiento de tres tipos de flores iris (setosa, virginica y versicolor). El algoritmo k-means es utilizado para agrupar los tipos de flor en base a dos características o componentes (longitud de sépalo y ancho de sépalo). La Figura muestra los resultados de la agrupación y los centros de clusters.

En el caso de las imágenes el algoritmo trabaja para segmentar el fondo de la parte de la imagen que ocupamos, para ello utilizará una componente (si la imagen está en escala de grises) o tres componentes (si la imagen es a color -RGB-)

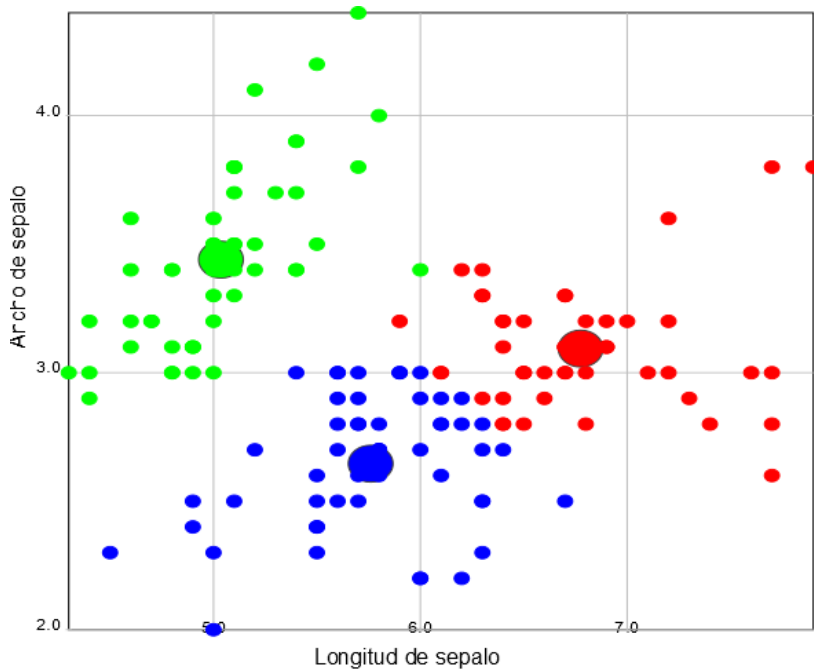


Figura 2. Agrupamiento de tres tipos de flores iris.

El algoritmo a utilizar, está basado en los pasos aquí mostrados.

1. Se eligen  $k$ -puntos como centroides, de los que existan en nuestra imagen para comenzar a agrupar los datos que se tienen.

2. Se calcula la distancia de todos los elementos, con respecto al número de centroides que se tenga, generalmente esta distancia es calculada haciendo uso de distancias euclidianas y agregando el punto al centroide con la distancia más corta.
3. Una vez obtenido el grupo de puntos por cada centroide, se calcula la media por cada punto y se obtiene la posición del nuevo centroide.
4. Este proceso se realiza hasta que no existan cambios en los centros.

## 2.4 Segmentación 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  $N$  píxeles que representan  $L$  niveles de gris, las probabilidades acumuladas hasta  $T$  y desde  $T$  hasta  $L$  resulta ser:

$$w_1(t) = \sum_{z=1}^T P(z) \quad (2.9)$$

$$w_2(t) = \sum_{z=T+1}^L P(z) \quad (2.10)$$

Donde  $P(z)$  es la distribución del histograma. Las medias y varianzas asociadas están por:

$$\mu_1(t) = \sum_{z=1}^T \frac{zP(z)}{w_1(t)} \quad (2.11)$$

$$\mu_2(t) = \sum_{z=1}^T \frac{zP(z)}{w_2(t)} \quad (2.12)$$

$$\sigma_1^2(t) = \sum_{z=1}^T (z - \mu_1(t))^2 \frac{P(z)}{w_1(t)} \quad (2.13)$$

$$\sigma_2^2(t) = \sum_{z=T+1}^L (z - \mu_2(t))^2 \frac{P(z)}{w_2(t)} \quad (2.14)$$

Finalmente, la varianza ponderada se define como:

$$\sigma_w^2(t) = w_1(t)\sigma_1^2(t) + w_2(t)\sigma_2^2(t) \quad (2.15)$$

Se elige el umbral  $T$  correspondiente al nivel de intensidad que proporcione la mínimavarianza ponderada definida en (2.9). De modo que para el umbral  $T$  con la condición mayor /menor o igual se elige la imagen deseada.

## 2.5 Técnicas de dilatación y erosión

La imagen binaria  $A$  aun puede contener pixels que son parte del fondo, para eliminarlos sin alterar los caracteres, la imagen binaria es erosionada. Esto es:

$$A_e = A \ominus B = \{(x', y') | (x' + i, y' + j) \in P_A, (i, j) \in P_B\} \quad (2.16)$$

Donde  $A$  es la imagen original,  $A_e$  representa la imagen erosionada con el operador de erosión y  $\ominus$  denota el operador de erosión empleando la mascara  $B$ .

Por otro lado, la imagen  $A$  puede ser dilatada para restaurar posibles caracteres *rotos* o incompletos, esto es posible realizarlo mediante el siguiente operador:

$$A_d = A \oplus B = \{(x', y') = (x + i, y + j) | (x', y') \in P_A, (i, j) \in P_B\} \quad (2.17)$$

Donde  $A_d$  es la imagen dilatada y  $\oplus$  denota el operador de dilatación utilizando la mascara  $B$ .

## 2.6 Técnicas para eliminar ruido en imágenes binarias

Aunado a las técnicas de erosión y dilatación, existen técnicas para eliminar el ruido en las imágenes. Las siguientes ecuaciones permiten realizar este trabajo.

1. Quitar ruido (puntos negros) de zonas blancas

$$P_0 = P + (b \cdot g \cdot (d + e) + (d \cdot e \cdot (b + g))) \quad (2.18)$$

2. Quitar ruido (puntos blancos) de zonas negras

$$P_0 = P + (((a + b + d) \cdot (e + g + h)) + ((b + c + e) \cdot (d + f + g))) \quad (2.19)$$

3. Reconstruir esquinas de figura blanca

a. Superior derecha

$$P_0 = \bar{P} \cdot d \cdot f \cdot g \cdot \overline{(a + b + c + e + h)} + P \quad (2.20)$$

b. Inferior derecha

$$P_0 = \bar{P} \cdot a \cdot b \cdot d \cdot \overline{(e + c + f + g + h)} + P \quad (2.21)$$

c. Superior izquierda

$$P_0 = \bar{P} \cdot e \cdot g \cdot h \cdot \overline{(a + b + c + d + f)} + P \quad (2.22)$$

d. Inferior izquierda

$$P_0 = \bar{P} \cdot b \cdot c \cdot e \cdot \overline{(a + d + f + g + h)} + P \quad (2.23)$$

4. Reconstruir esquinas de figura negra

a. Superior derecha

$$P_0 = P \cdot (d + f + g) + \overline{(a \cdot b \cdot c \cdot e \cdot h \cdot P)} \quad (2.24)$$

b. Inferior derecha

$$P_0 = P \cdot (a + b + d) + \overline{(c \cdot e \cdot f \cdot g \cdot h \cdot P)} \quad (2.25)$$

c. Superior izquierda

$$P_0 = P \cdot (e + g + h) + \overline{(a \cdot b \cdot c \cdot d \cdot f \cdot P)} \quad (2.26)$$

d. Inferior izquierda

$$P_0 = P \cdot (b + c + e) + \overline{(a \cdot d \cdot f \cdot g \cdot h \cdot P)} \quad (2.27)$$

### 3 Técnicas de Extracción de características

En la actualidad, las imágenes son utilizadas en múltiples campos de aplicación. Uno de los campos de aplicación más activos en los últimos años es el de visión artificial. La visión artificial utiliza métodos para procesar analizar y comprender imágenes de una forma parecida a como lo hace el ser humano. Para ello, utiliza métodos de procesamiento que permiten extraer la información importante de la imagen. En este Capítulo se muestran las técnicas de extracción de características comúnmente utilizadas.

#### 3.1 Alturay ancho

Una vez obtenida la imagen segmentada, obtenemos estos descriptores calculando la altura y ancho de la región segmentada. Estos dos descriptores son muy convenientes para objetos de poca complejidad. Pueden ser estimados mediante las siguientes expresiones:

$$\text{Ancho} = w = i_{\max} - i_{\min} \quad (3.1)$$

$$\text{Altura} = h = j_{\max} - j_{\min} \quad (3.2)$$

Donde  $i$  y  $j$  son, respectivamente, el eje mayor y menor del objeto. A partir de estos descriptores se pueden definir un rasgo combinado conocido en la literatura como elongación o excentricidad.

#### 3.2 Área

En una imagen digital, el área de un objeto está dada por el número de píxeles que representan al mismo, por lo tanto, el cálculo del área se realiza contando el número de píxeles.

Si una imagen está representada por un contorno codificado en código de cadena, el cálculo del área se realiza por un algoritmo que trabaja en forma similar a la integración numérica.



### 3.3 Perímetro

Es un parámetro geométrico que al igual que el área, puede ser calculado a partir del código de cadena. Para realizar este cálculo, es necesario contar la longitud del código y tomar en consideración que los pasos en direcciones diagonales deben ser multiplicados por un factor igual a raíz cuadrada de dos.

El perímetro  $p$  de un contorno codificado en código de cadena se expresa como:

$$p = n_e + n_o\sqrt{2} \quad (3.3)$$

Donde  $n_e$  representa el número de pasos pares del código y  $n_o$  el número de pasos impares del código.

### 3.4 Rectangularidad

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

$$R = \frac{A_S}{A_R} \quad (3.4)$$

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

### 3.5 Inercia

Cuanto mayor es el contraste de la imagen, mayor es el valor de Inercia al ponderar las diferencias  $(i - j)$  con  $(i - j)^2$ . La sumatoria de todas estas diferencias nos da la inercia de la región segmentada.

$$\sum_i \sum_j (i - j)^2 c_{ij} \quad (3.5)$$

### 3.6 Solidez

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

$$\text{Solidez} = \frac{A_s}{H} \quad (3.6)$$

donde  $A_s$  es el área de la región de la imagen y  $H$  es el área del caso convexo de la imagen.

### 3.7 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:

$$E = \frac{\lambda_2}{\lambda_1} \quad (3.7)$$

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

$$E = \frac{W-H}{H} \quad (3.8)$$

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

$$W = \max X(t) - \min X(t) \quad (3.9)$$

$$H = \max Y(t) - \min Y(t) \quad (3.10)$$

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

### 3.8 Compacidad

La compacidad de un objeto segmentado permite calcular cuan poroso es un objeto en su interior, cuanto mayor compacidad tenga la región segmentada, menor porosidad tendrá el objeto. La Figura 3 muestra varios ejemplos con regiones segmentadas cuya área es idéntica pero la compacidad es diferente.

La compacidad es calculada mediante la siguiente ecuación:

$$\text{Compacidad} = \frac{P^2}{A} \quad (3.11)$$

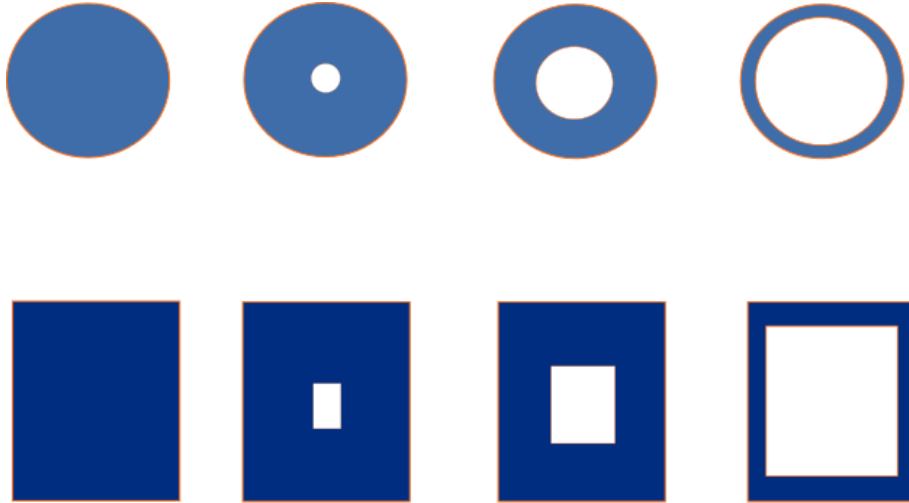


Figura 3. Regiones Segmentadas con la misma área

### 3.9 Número de Euler

El número de Euler describe la relación entre el número de partes convexas y el número de huecos en una forma. En la figura 3 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 convexas y a  $N$  el número de huecos. Entonces el número de Euler viene de la siguiente expresión:

$$Eul = S - N$$

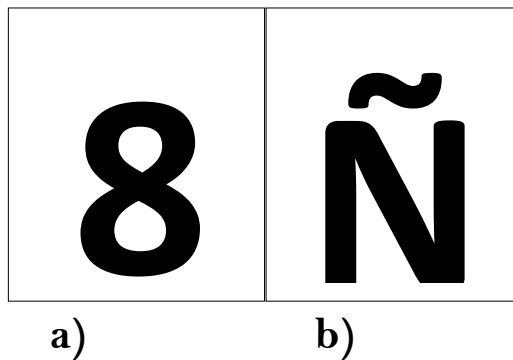


Figura 4: Ejemplos de regiones segmentadas con huecos y componentes conexas. a) Región con dos huecos.  
b) Región con dos componentes conexas.

### 3.10 Proyección

Las proyecciones son representadas por la forma de la región segmentada en los ejes  $x$  e  $y$  sobre el sistema de coordenadas cartesiano. Cada proyección está definida por un vector unidimensional:

$$Pro_x(i) = \sum_{j=jmin}^{jmax} f(i, j) \quad (3.12)$$

$$Pro_y(j) = \sum_{i=imin}^{imax} f(i, j) \quad (3.13)$$

### 3.11 Centro de gravedad o centroide

El centro de gravedad también o también llamada centroide. Su posición debería ser fija en relación con la forma. Si una forma es representada por su función de región. sucentroide( $g_x, g_y$ ) es:

$$\begin{cases} g_x = \frac{1}{n} \sum_{i=1}^n x_i \\ g_y = \frac{1}{n} \sum_{i=1}^n y_i \end{cases} \quad (3.14)$$

Donde  $n$  es el número de píxeles en la región segmentada,  $(x_i, y_i) \in \{(x_i, y_i) | f(x_i, y_i) = 1\}$

### 3.12 Excentricidad: Método de ejes principales

La excentricidad es la medida de la relación de aspecto. Es la relación de la longitud del eje menor. Puede ser calculado por el método de los ejes principales o por el método rectángulo delimitado mínimo.

Los ejes principales de una forma dada pueden ser únicamente como los dos segmentos de líneas que cruzan el uno al otro ortogonalmente en el centroide de la forma y representar las direcciones con correlación cruzada cero (Guillaume Cerutti, 2013). De esta manera, un contorno es visto como un ejemplo de una distribución estadística. Consideremos la matriz de covarianza  $C$  de un entorno:

$$C = \frac{1}{N} \sum_{i=0}^{N-1} \begin{pmatrix} x_i - g_x \\ y_i - g_y \end{pmatrix} \begin{pmatrix} x_i - g_x \\ y_i - g_y \end{pmatrix}^T = \begin{pmatrix} C_{xx} & C_{xy} \\ C_{yx} & C_{yy} \end{pmatrix} \quad (3.15)$$

Donde

$$C_{xx} = \frac{1}{N} \sum_{i=0}^{N-1} (x_i - g_x)^2 \quad (3.16)$$

$$C_{xy} = \frac{1}{N} \sum_{i=0}^{N-1} (x_i - g_x)(y_i - g_y) \quad (3.17)$$

$$C_{yx} = \frac{1}{N} \sum_{i=0}^{N-1} (y_i - g_y)(x_i - g_x) \quad (3.18)$$

$$C_{yy} = \frac{1}{N} \sum_{i=0}^{N-1} (y_i - g_y)^2 \quad (3.19)$$

$G(g_x, g_y)$  es el centroide de la forma.

Las longitudes de los dos ejes principales son iguales a los dos valores propios  $\lambda_1$  y  $\lambda_2$  de la matriz de covarianza  $C$  de un contorno respectivamente.

Así los valores  $\lambda_1$  y  $\lambda_2$  se pueden calcular.

$$\det(C - \lambda_{1,2}I) = \det \begin{pmatrix} C_{xx} - \lambda_{1,2} & C_{xy} \\ C_{yx} & C_{yy} - \lambda_{1,2} \end{pmatrix} = (C_{xx} - \lambda_{1,2})(C_{yy} - \lambda_{1,2}) - C_{xy}^2 = 0 \quad (3.20)$$

Así que

$$\begin{cases} \lambda_1 = \frac{1}{2} \left[ C_{xx} + C_{yy} + \sqrt{(C_{xx} + C_{yy})^2 - 4(C_{xx}C_{yy} - C_{xy}^2)} \right] \\ \lambda_2 = \frac{1}{2} \left[ C_{xx} + C_{yy} - \sqrt{(C_{xx} + C_{yy})^2 - 4(C_{xx}C_{yy} - C_{xy}^2)} \right] \end{cases} \quad (3.21)$$

Entonces la excentricidad se puede calcular:

$$E = \lambda_2 / \lambda_1 \quad (3.22)$$

### 3.13 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:

$$V_i = \begin{pmatrix} x_i - g_x \\ y_i - g_y \end{pmatrix} \quad (3.23)$$

$$d'_i = \sqrt{V_i^T * C^{-1} * V_i} \quad (3.24)$$

$$\mu R = \frac{1}{N} \sum_{i=1}^N d'_i \quad y \quad \sigma R = \sqrt{\frac{1}{N} \sum_{i=0}^n (d_i - \mu R)^2} \quad (3.25) \text{ y } (3.26)$$

Entonces la varianza de elipse se obtiene como:

$$E_{ve} = \frac{\sigma R}{\mu R} \quad (3.27)$$

### 3.14 Momentos frontera

Momentos de contorno, el análisis de un contorno, se puede utilizar para reducir la dimensión de la representación límite (Hu, 1962). Asumir que el contorno de la forma ha sido representado como una función de forma 1-D es una representación de  $z(i)$  como se introdujo en la Sección 3, el  $r^{th}$  momento  $m_r$  y el momento central se pueden estimar como:

$$m_r = \frac{1}{N} \sum_{i=1}^N [z(i)]^r \quad y \quad \mu_r = \frac{1}{N} \sum_{i=1}^N [z(i) - m_1]^r \quad (3.28) \text{ y } (3.29)$$

Donde N es el número de puntos de contorno.

Los momentos normalizados  $\bar{m}_r = m_r / (\mu_2)^{r/2}$  y  $\bar{\mu}_r = \mu_r / (\mu_2)^{r/2}$  son variantes para dar forma a la rotación, traslación y escala. Los descriptores sensibles al ruido se pueden obtener a partir de:

$$F_1 = \frac{(\mu_2)^{1/2}}{m_1}, \quad F_2 = \frac{\mu_3}{(\mu_2)^{3/2}} \quad y \quad F_3 = \frac{\mu_4}{(\mu_2)^2} \quad (3.30), (3.31) \text{ y } (3.32)$$

El otro método momentos de contorno trata la forma 1-D función característica  $z(i)$  como una variable aleatoria  $\mathbf{v}$  y crea un contenedor  $K$  histograma  $p(v_i)$  de  $z(i)$ . Entonces el  $r^{th}$  momento central se obtiene por:

$$\mu_r = \sum_{i=1}^K (v_i - m)^r p(v_i) \quad \text{y} \quad m = \sum_{i=1}^K v_i p(v_i) \quad (3.33), (3.34)$$

La ventaja de descriptores momentos de contorno es que es fácil de implementar. Sin embargo, es difícil asociar momentos con interpretación física mayor.

### 3.15 Momentos invariantes de Hu

Dentro de los métodos que comúnmente se emplean para extraer características invariantes están los momentos de Hu (Hu, 1962). Estas características permiten identificar objetos independientemente de la escala, rotación y posición del objeto en la imagen. Para extraer estas características, se construyen los conjuntos:

$$O_k = \{(x_1^k, y_1^k), \dots, (x_n^k, y_n^k)\} \quad (3.35)$$

Donde  $O_k$  es el conjunto cuyos elementos son las coordenadas de los píxeles que conforman el  $k$ -ésimo objeto segmentado. Con estos conjuntos se obtienen las características que se han mencionado.

#### 3.15.1 Momentos ordinarios

Los momentos ordinarios invariantes son un conjunto de descriptores que permiten reconocer un objeto independientemente de su posición, tamaño y orientación. 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:

$$m_{pq} = \sum_{x=1}^M \sum_{y=1}^N x^p y^q f(x, y) \quad (3.36)$$

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

### 3.15.2 Momentos centralizados

Utilizando segmentos centralizados la fórmula general quedaría:

$$\mu = \sum_{x=1}^M (x - \bar{x})^p (y - \bar{y})^q f(x, y) \quad (3.37)$$

$$\bar{x} = \frac{m_{10}}{m_{00}} \quad \bar{y} = \frac{m_{01}}{m_{00}} \quad (3.38)$$

### 3.15.3 Momentos normalizados

Se definen los momentos centralizados normalizados como:

$$\eta_{pq} = \frac{\mu_{pq}}{\mu_{00}^k} \quad (3.39)$$

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

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

$$\theta_1 = \mu_{20} + \mu_{02} \quad (3.41)$$

$$\theta_2 = (\mu_{20} - \mu_{02})^2 + 4\mu_{11}^2 \quad (3.42)$$

$$\theta_3 = (\mu_{30} - \mu_{12})^2 + (3\mu_{20} - \mu_{03})^2 \quad (3.43)$$

$$\theta_4 = (\mu_{30} - \mu_{12})^2 + (3\mu_{21} - \mu_{03})^2 \quad (3.44)$$

$$\theta_5 = (\mu_{30} - 3\mu_{12})(\mu_{30} - \mu_{12})^2 [(\mu_{30} - \mu_{12})^2 - 3(\mu_{21} + 3\mu_{03})^2] + (3\mu_{21} - \mu_{03})(\mu_{21} + \mu_{03}) [3(\mu_{30} - \mu_{12})^2 - (\mu_{21} + 3\mu_{03})^2] \quad (3.45)$$

$$\theta_6 = (\mu_{20} - \mu_{02}) [(\mu_{30} + \mu_{12})^2 - (3\mu_{21} - \mu_{03})^2] + 4\mu_{11}(\mu_{30} + \mu_{12})(\mu_{21} - \mu_{03}) \quad (3.46)$$

$$\theta_7 = (3\mu_{21} - \mu_{03})(\mu_{30} + \mu_{12}) [(\mu_{30} - \mu_{12})^2 - 3(\mu_{21} + 3\mu_{03})^2] + (3\mu_{12} - \mu_{30})(\mu_{21} - \mu_{03}) [3(\mu_{30} - \mu_{12})^2 - (\mu_{21} + 3\mu_{03})^2] \quad (3.47)$$



### 3.16 Momentos R

Los momentos R fueron definidos por (JingLiuFeaturedExtractionTechnique base don thePerceptiveInvariability, 2008) y son definidos en base a los momentos de Hu como sigue:

$$R_1 = \frac{\sqrt{\theta_2}}{\theta_1}; \quad R_2 = \frac{\theta_1 + \sqrt{\theta_2}}{\theta_1 - \sqrt{\theta_2}}; \quad R_3 = \frac{\sqrt{\theta_3}}{\theta_4}; \quad R_4 = \frac{\sqrt{\theta_3}}{\sqrt{|\theta_5|}}; \quad R_5 = \frac{\sqrt{\theta_4}}{\sqrt{|\theta_5|}};$$

$$R_6 = \frac{|\theta_6|}{\theta_1 \cdot \theta_3}; \quad R_7 = \frac{|\theta_6|}{\theta_1 \cdot \sqrt{|\theta_5|}}; \quad R_8 = \frac{|\theta_6|}{\theta_3 \cdot \sqrt{|\theta_2|}}; \quad R_9 = \frac{|\theta_6|}{\sqrt{\theta_2 \cdot |\theta_5|}}; \quad R_{10} = \frac{|\theta_5|}{\theta_3 \cdot \theta_4}$$

**Comentado [DVL1]:** falta definir que es teta, en general revisar si existe en todas las ecuaciones la descripción de todas las variables

### 3.17 Momentos 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 (Ji-Xiang Du C.-M. Z.-P., 2013).

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

$$u = a_0 + a_1x + a_2y \quad (3.48)$$

$$v = b_0 + b_1x + b_2y \quad (3.49)$$

Ya que los momentos de Flusser parten de los momentos utilizados por Hu, los momentos de orden general  $m_{pq}$  y los momentos centrales  $\mu_{pq}$ , se definen en (3.36) y (3.37) respectivamente.

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

$$I_1 = \frac{(\mu_{20}\mu_{02} - \mu_{11}^2)}{\mu_{00}^4} \quad (3.50)$$

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

$$I_2 = \frac{\mu_{30}^2 \mu_{03}^2 - 6\mu_{30}\mu_{21}\mu_{12}\mu_{03} + 4\mu_{30}\mu_{12}^3 + 4\mu_{21}^3\mu_{03} - 3\mu_{21}^2\mu_{12}^2}{\mu_{00}^{10}} \quad (3.51)$$

Finalmente, de los momentos de segundo y tercer orden:

$$I_3 = \frac{(\mu_{20}[\mu_{20}\mu_{03} - \mu_{12}^2] - \mu_{11}[\mu_{30}\mu_{03} - \mu_{21}\mu_{12}] + \mu_{02}[\mu_{30}\mu_{12} - \mu_{21}^2])}{\mu_{00}^7} \quad (3.52)$$

### 3.18 Descriptores de Fourier

Los descriptores de Fourier consideran al conjunto de puntos que conforman el contorno de un objeto como una secuencia de números complejos, con los cuales se construye una función periódica unidimensional  $f$  que modela el contorno del objeto; los coeficientes de la transformada de Fourier de la función  $f$  caracterizan el contorno de un objeto en el dominio de la frecuencia (Cuevas, Zaldívar & Pérez, 2010).

Sea  $C \subset O$  el conjunto de las coordenadas de los píxeles del contorno de un objeto segmentado. Las coordenadas en

$$O = \{(x_0, y_0), \dots, (x_{M-1}, y_{M-1})\} \quad (3.53)$$

están indexadas de acuerdo a la secuencia en la que se encuentran ubicadas en el contorno en una dirección específica y que comienza en un punto arbitrario. Sea  $f: C \rightarrow \mathbb{C}$  la función que transforma una coordenada en un número complejo como sigue:

$$f((x_k, y_k)) = x_k + iy_k \quad (3.54)$$

Donde  $i = \sqrt{-1}$  y  $\mathbb{C}$  es el conjunto de números complejos. La transformada de Fourier de la función  $f$  se obtiene con:

$$F(u) = \frac{1}{M} \sum_{k=0}^{M-1} f((x_k, y_k)) \exp\left(\frac{-i2\pi uk}{M}\right) \quad (3.55)$$

A los coeficientes complejos  $F(u)$  se les conoce como descriptores de Fourier del contorno. En este trabajo se obtienen 110 coeficientes, esto es,  $F(u)$  es calculada para  $u = 0, 1, \dots, 109$ . Es importante señalar que tanto los descriptores de Fourier como los momentos de Hu son invariantes a los cambios de escala, posición y rotación de los objetos.

## 4 Técnicas de clasificación

En este Capítulo se muestran las diferentes técnicas de clasificación y se hace un énfasis especial en el clasificador Bayesiano. Estas técnicas se utilizaron para comparar los tiempos de entrenamiento de los diferentes modelos de clasificación en comparación con el clasificador Bayesiano.

### 4.1 Clasificador Bayesiano

Los clasificadores Bayesianos están basados en la premisa más sencilla de clasificación. Suponiendo que existe dos clases  $c_1$  y  $c_2$ , si quisiéramos predecir a cuál de ellas pertenecerá el siguiente patrón que se presente conociendo la historia y características tendríamos que hacer uso de la primera regla de decisión. La primera regla de decisión permite predecir en base a la frecuencia con la que han aparecido los patrones de una clase y otra.

La primera regla de decisión nos indica que debemos optar por la clase  $c_1$ , siempre y cuando  $P(c_1) > P(c_2)$ , cuando esto no se presente debemos elegir por la  $c_2$ .  $P(c_1)$  representa la probabilidad a priori de que el patrón pertenezca a la clase  $c_1$ , mientras que  $P(c_2)$  representa la probabilidad a priori de que el patrón pertenezca a la clase  $c_2$ . Tanto  $P(c_1)$  como  $P(c_2)$  son estimadas como la proporción de patrones que pertenecen a una u otra clase que se han presentado hasta el momento. Es claro que al contener tan poca información, la probabilidad de equivocarnos es muy grande. Algunas técnicas para evaluar el desempeño de un clasificador describen a un clasificador como un clasificador que esta adivinando cuando la probabilidad de que acierte es del 0.5. En el mejor de los casos el desempeño de nuestro clasificador de primera regla de decisión sería de 0.5.

En reconocimiento de patrones la decisión de pertenencia de un conjunto de características a una u otra clase está dada por su probabilidad a posteriori ( $P(c_1|x_i)$ ), es decir esta es calculada en base a la evidencia que tenemos.

Para el reconocimiento de caracteres se implementa un clasificador Bayesiano; este clasificador estadístico se basa en el teorema de Bayes donde se asume que el vector de características tiene una distribución Gaussiana multivariable.

Sea  $\{c_1, \dots, c_k\}$  el conjunto de  $k$  clases de caracteres, la probabilidad de que un carácter  $\mathbf{x}$  sea de la clase  $c_i$  se denota por  $p(c_i|\mathbf{x})$ . Del teorema de Bayes:

$$p(c_i|\mathbf{x}) = \frac{p(\mathbf{x}|c_i)p(c_i)}{p(\mathbf{x})} \quad (4.1)$$

Para la clasificación Bayesiana se escoge la clase  $c_i$  para la cual  $p(\mathbf{x}|c_i)p(c_i)$  es el mayor. En otras palabras:

$$\arg \max_i p(\mathbf{x}|c_i)p(c_i) \quad (4.2)$$

De esta forma al carácter observado  $\mathbf{x}$  se le asigna la clase  $c_i$ . Asumimos que la distribución de probabilidad del vector de características del carácter observado  $\mathbf{x}$ , sea de clase  $c_j$ , es Gaussiana con media  $\boldsymbol{\mu}_j$  y con matriz de covarianza  $\boldsymbol{\Omega}_j$ :

$$p(\mathbf{x}|c_j) = \frac{1}{\Delta} \exp \left[ -\frac{1}{2} (\mathbf{x} - \boldsymbol{\mu}_j)^T \boldsymbol{\Omega}_j^{-1} (\mathbf{x} - \boldsymbol{\mu}_j) \right] \quad (4.3)$$

Donde  $\Delta = (2\pi)^{m/2} (\det \boldsymbol{\Omega}_j)^{1/2}$  y  $m$  es la dimensión de los vectores de características.

## 4.2 Redes neuronales artificiales

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 pudieran 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 (G.J. Bowden, 2003), (Himmelblau, 1992). Más adelante, en las décadas de los 50's y 70s, el movimiento de las redes neuronales fue retomado por B. Widrow y M. E. Hoff, quienes trabajaron una maquina llamada *Adaline* (B. Widrow, 1987), (B. Widrow, 1987).

Una Red de Neuronas Artificiales (RNA) es un paradigma de procesamiento de información inicialmente inspirado en el modo en el que lo hace el cerebro (Martínez, 1995), (R. González García, 1998), (Pazos, 1996). Las RNA están compuestas por varios elementos de procesamiento o neuronas que trabajan en conjunto para resolver un problema específico. Las RNA se basan en el modelo propuesto por McCulloch y Pitts en 1943 (Zhang L. Z., 1999), en su modelo cada neurona recibe un conjunto de entradas  $\{x_1, x_2, \dots, x_n\}$  y devuelve una única salida  $y$ . Dentro de cada RNA existen numerosas conexiones entre las distintas neuronas que la forman dichas conexiones simula las conexiones interneuronales del cerebro y, al igual que estas pueden establecerse con mayor o menor intensidad. En las RNA estas intensidades la determinan los pesos sinápticos de tal forma que cada entrada  $x_i$  de una neurona se encuentra afectada por un peso  $w_i$ . Esto se puede representar en la siguiente figura:

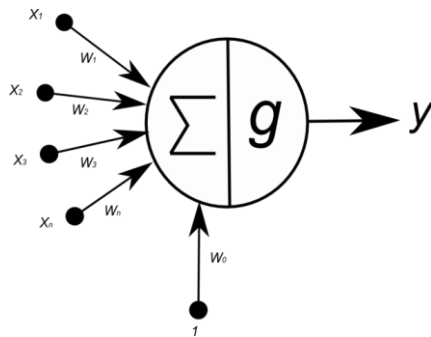


Figura 5: Intensidades en las RNA determinadas por los pesos sinápticos.

Comentado [DVL2]:

La salida  $y$  se calcula a partir de la suma ponderada de las entradas, es decir:

$$a = \sum_{i=1}^n w_i x_i + x_0 \quad (4.4)$$

Donde  $w_0$  de la Ec. (4.4) es un umbral que se usa para compensar la diferencia entre el promedio de las entradas, sobre todo en el conjunto de entrenamiento, y el correspondiente valor medio de las salidas deseadas. Para obtener el valor  $y$  de la RNA se debe aplicar una función llamada función de activación o función de transferencia al valor  $a$ . Es decir:

$$y = g(a) = g\left(\sum_{i=1}^n w_i x_i + x_0\right) = g\left(\sum_{i=1}^n w_i x_i\right) \quad (4.5)$$

Se puede observar que  $w_0$  puede ser tratado como un peso más si se añade un valor de entrada  $x_0 = 1$  como se observa en la Ec. (4.5). La función de activación en el trabajo de McCulloch y Pitts es una función tipo escalón como se muestra en la siguiente Figura:

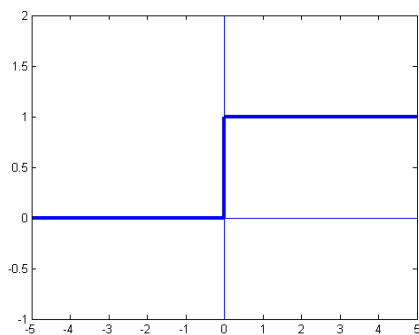


Figura 6: Función tipo escalón.

Algunas otras funciones usadas en RNA como funciones de activación son (Martin T. Hagan, 1996):

Función Lineal

$$g(a) = a \quad (4.6)$$

Función Sigmoidea

$$g(a) = \frac{1}{1 + e^{-a}} \quad (4.7)$$

Función Tangente Hiperbólica

$$g(a) = \tanh(a) = \frac{e^a - e^{-a}}{e^a + e^{-a}} \quad (4.8)$$

Función Gaussiana

$$g(a) = \exp\left(-\frac{(a - \mu)^2}{2\sigma^2}\right) \quad (4.9)$$

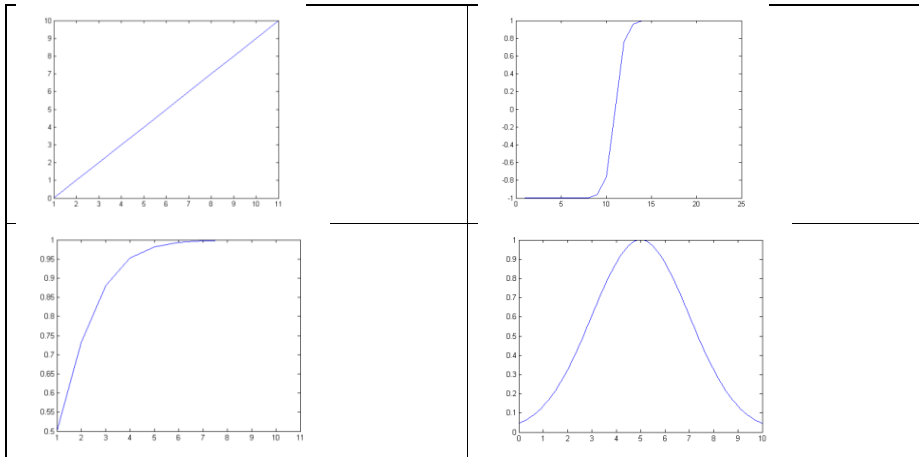


Figura 7: Función Lineal, Función Sigmoidea, Función Tangente Hiperbólica y Función Gaussiana.

Comentado [DVL3]: Falta fuente

### 4.3 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 estadístico. Inicialmente las SVMs fueron utilizadas solamente para resolver problemas de 2 clases. En muchas aplicaciones, las SVM han mostrado tener gran desempeño más que las máquinas de aprendizaje tradicional como las RNA.

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 espacio. La figura 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.

Cuando se tiene un problema de clasificación binaria, donde las dos clases son separables mediante una recta, existe un número infinito de soluciones debido a que pueden existir diversas rectas que separen adecuadamente los datos de las dos clases. La Figura 8 muestra un ejemplo de un problema de clasificación binaria y la Figura 9 es un ejemplo de las rectas que separen correctamente las dos clases.

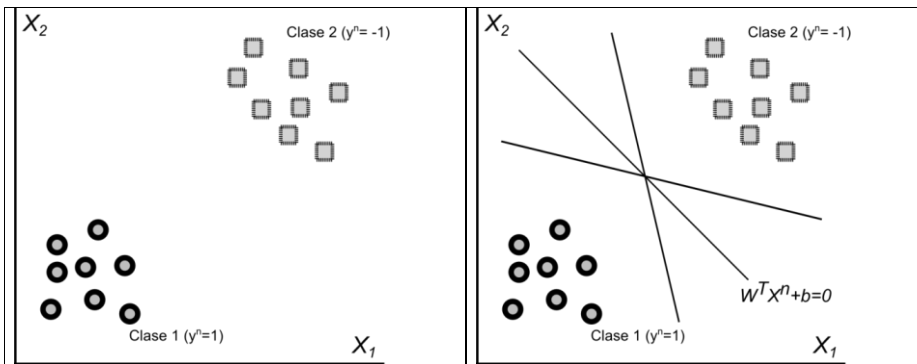


Figura 8: Problema de clasificación binaria. Figura 9: Rectas que separen dos clases.

En las figuras anteriores se puede observar que no todas las rectas que separen correctamente a las dos clases son igualmente óptimas. En las siguientes figuras 10 y 11 se puede observar que tiene una recta que separa correctamente a las dos clases, cada recta discriminadora cuenta con dos rectas paralelas (líneas punteadas) que pasan por los puntos de cada clase más cercanos a la recta discriminante, dichas rectas forman un margen entre los puntos. Se puede observar que las rectas que forman un mayor margen es la que

producirá menos errores con los nuevos datos. Esta afirmación está demostrado en el trabajo de Vapnik (Vladimir Vapnik, 1995).

Comentado [DVL4]:

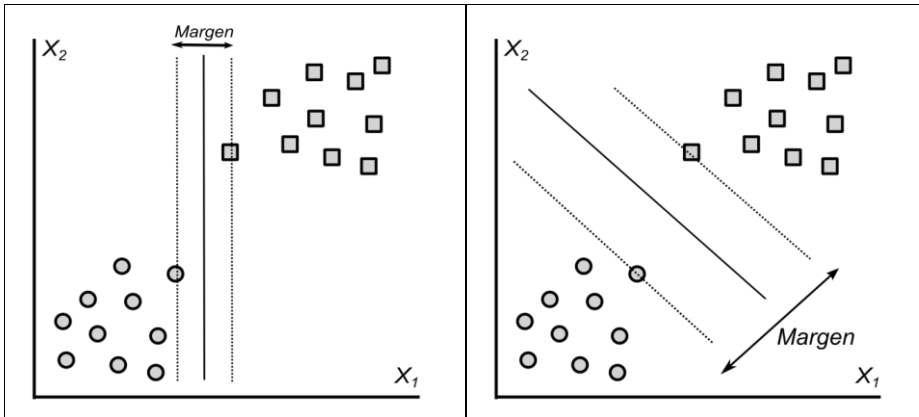


Figura 10: Separación de dos clases por una recta. Figura 11: Separación de dos clases por recta.

Para obtener dicho margen se plantea el siguiente ejemplo: Si se tienen  $n$  datos, que son representados por  $x_1^d, x_2^d, \dots, x_n^d$  donde  $d$  representa la dimensionalidad del conjunto de datos y suponiendo que trabajamos con clasificación binaria, las clases a la que pertenece cada dato está representada por  $y_i = \{-1, 1\}$ . Entonces el hiperplano de separación está dado por

$$\sum_{i=1}^n w_i x_i + x_0 \tag{4.10}$$

$$w^T x + b$$

Dicha recta (4.10) debe ser capaz de discriminar entre ambas clases y debe también obtener el mayor margen de separación con respecto a los datos. Donde  $w$  representa la pendiente de la recta y  $b$  el desplazamiento al origen. Se definen dos rectas paralelas ( $H_1$  y  $H_2$ ) a la recta discriminadora de tal manera que una clase de datos esté por debajo de una recta ( $H_1$ ) y todos los de la otra clase por encima de la segunda recta ( $H_2$ ) como se observa en la Figura 12, es decir:

$$H_1: w^T x_i + b \geq 1, \text{ para } y_i = 1 \tag{4.11}$$

$$H_2: w^T x_i + b \leq -1, \text{ para } y_i = -1 \tag{4.12}$$



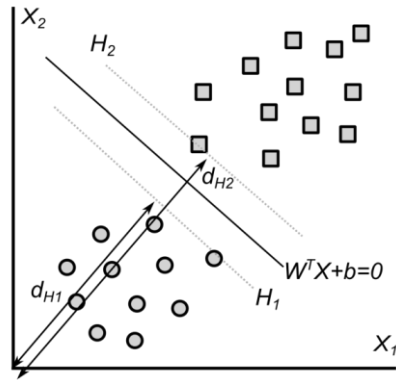


Figura 12: Rectas paralelas a la discriminadora

La distancia perpendicular de  $H_1$  y  $H_2$  al origen es:

$$d_{H_1} = \frac{|b - 1|}{\|w\|} \quad (4.13)$$

$$d_{H_2} = \frac{|b + 1|}{\|w\|} \quad (4.14)$$

A partir de la ecuación (4.15) se obtiene que la distancia o margen entre ambas rectas es:

$$\text{margen} = |d_{H_1} - d_{H_2}| = \frac{2}{\|w\|} \quad (4.15)$$

Este ejemplo presenta un problema de minimización, ya que se requiere encontrar una recta discriminadora que presente un mayor margen, y esa condición se logra obteniendo el valor mínimo de  $\|w\|$ . Es decir:

$$\text{Minimizar } E(w) = \frac{1}{2} \|w\|^2 \quad (4.16)$$

$$\text{Sujeto a: } y_i(w^T x_i + b) - 1 \geq 0, \forall i = 1, 2, \dots, n \quad (4.17)$$

La restricción son las mismas ecuaciones (4.16) pero escrita en una sola ecuación, además la función a minimizar se le ha incluido la constante  $\left(\frac{1}{2}\right)$  y el cuadrado de la norma, esto no influye en la solución óptima pero facilita los cálculos a la hora de resolver el problema ya que la función será derivada.

El problema debe replantearse usando la fórmula dual ya que se tiene un doble objetivo: obtener un problema más fácil de resolver y que permita generalizar más fácilmente el método al caso de datos que no sean linealmente separables.

Para obtener el problema dual se aplican los multiplicadores de Lagrange al problema planteado en la ecuación (4.16) dando lugar a la siguiente función de Lagrange del problema primario (4.18):

$$L(w, b; \alpha) = \frac{1}{2} \|w\|^2 - \sum_{i=1}^n \alpha_i [y_i (w^T x_i + b) - 1] \quad (4.18)$$

Donde  $\alpha = \{\alpha_1, \alpha_2, \dots, \alpha_n\}$  es el vector de los multiplicadores no negativos de Lagrange. A partir de la función lagrangiana se puede definir el problema dual. Se debe obtener el punto estacionario de la función lagrangiana, el cual se logra diferenciando a la función con respecto a las variables primales e igualándolas a cero:

$$\nabla_w L(w, b; \alpha) = w - \sum_{i=1}^n \alpha_i [y_i x_i] = 0 \quad (4.19)$$

$$\nabla_b L(w, b; \alpha) = \sum_{i=1}^n \alpha_i y_i = 0 \quad (4.20)$$

A partir de la ecuación (4.19) se tiene que:

$$w = \sum_{i=1}^n \alpha_i [y_i x_i] \quad (4.21)$$

Sustituyendo, el resultado de la ecuación (4.21) en la función de Lagrange definida previamente en la ecuación (4.18) se obtiene:

$$L(w, b; \alpha) = \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j y_i y_j x_i^T x_j - b \sum_{j=1}^n \alpha_j y_j \quad (4.22)$$

Dado el resultado de (4.19) y el último valor de (4.20) es igual a cero se obtiene:

$$L(w, b; \alpha) = \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j y_i y_j x_i^T x_j \quad (4.23)$$

Como se observa en la ecuación(4.24)depende únicamente de las variables duales. El problema dual se plante como un problema de maximización:

$$\max L(w, b; \alpha) = \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j y_i y_j x_i^T x_j \quad (4.24)$$

Sujeto a: (4.25)

$$\sum_{i=1}^n \alpha_i y_i = 0$$

$$\alpha_i \geq 0, i = 1, \dots, n$$

Una vez obtenidos los parámetrosóptimos, la forma de asignar la clase de cada dato es aplicando la función $\text{sgn}(W^T X + b)$ , donde  $\text{sgn}(\cdot)$  es la función signo.

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.

## 4.4 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.

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.

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.

Los árboles de decisión (DT-DecisionTree) es uno de los métodos de aprendizaje más sencillo de entender y usar, dada esta ventaja los DT son usados en diferentes tareas de minería de datos, como la clasificación (Zorman, 2000) y la regresión (Pang - Ning Tan, 2006). Un DT es una representación gráfica de un procedimiento para clasificar o evaluar un concepto (Quinlan, 1986), es una colección de condiciones organizadas jerárquicamente, donde cada nodo representa una condición sobre algún atributo y cada rama que nace de ese nodo es un posible valor que puede tomar dicho atributo, y al final se encuentran las hojas que son las clases.

Para clasificar una instancia se inicia en el nodo raíz, se aplica el test al atributo que solicita el nodo y se sigue la rama que le corresponda, el proceso se repite hasta alcanzar una de las hojas. Si el atributo clase es discreto el DT recibe el nombre de árbol de clasificación, si el atributo de clase toma valores de rango continuo se dice entonces que se trata de un árbol de regresión.

Algunas de las ventajas de los DT son enumeradas a continuación:

1. En su forma más general, las opciones posibles a partir de una determinada condición son excluyentes, lo que permite llegar a una sola acción o decisión a tomar.
2. Condensan la información de los ejemplos iniciales y permiten la clasificación de nuevos ejemplos siempre y cuando no existan modificaciones sustanciales de las condiciones con las que se entrenó el conjunto de datos.
3. Permite interpretar la solución al conocer el conocimiento utilizado en la toma de decisiones.
4. Reduce el número de variables independientes

Existen dos tipos de particiones las cuales solo están permitidas en los DT:

1. Particiones sobre atributos nominales: si un atributo  $x_i$  es nominal y sus valores son  $\{v_1, v_2, \dots, v_k\}$ , entonces las posibles particiones son de la forma  $x_i = v_j$  siendo  $1 \leq j \leq k$ .
2. Particiones sobre atributos numéricos: generalmente, si el atributo  $x_i$  es numérico (continuo) su rango se parte en dos intervalos, tales que las particiones son de la forma  $x_i \leq v_c$ ,  $x_i > v_c$ , siendo  $v_j$  algún valor intermedio entre los valores máximo  $v_{max}$  y mínimo  $v_{min}$  que tiene el atributo  $x_i$ .  $v_c$  es algún punto de corte y existen diversas aproximaciones para determinararlo. La más sencilla es ordenar los datos de E en orden creciente de acuerdo al valor del atributo  $x_i$ .

Una vez determinadas las posibles particiones, el siguiente paso es elegir la mejor partición para el nodo. Para elegir una partición idónea se basa en funciones heurísticas, las cuales se basan en minimizar la entropía.

La entropía es una medida usada habitualmente para caracterizar el grado de pureza o impureza de los datos. Dada un conjunto de datos  $E$  de un problema de clasificación con  $c$  clases, su entropía se define como:

$$Entropia(E) = \sum_{i=1}^c -p_i \cdot \log_2 p_i \quad (4.26)$$

Donde  $p_i$  es la probabilidad de que un dato sea de la clase  $i$  (número de ejemplos de clase  $i$  dividido por el número de ejemplos de  $E$ ). Una forma cuantitativa de ver el efecto a partir de los datos por un atributo en particular es usar el criterio llamado Ganancia de Información, el cual simplemente calcula la reducción esperada de la entropía por partir los datos de acuerdo a ese atributo.

Formalmente, la ganancia de información,  $Gain(E, x)$ , de un conjunto de datos  $E$  en relación al atributo  $x$ , se define como:

$$Ganancia(E, x) = Entropia(E) - \sum_{v \in V(x)} \frac{|E_v|}{|E|} Entropia(E_v) \quad (4.27)$$

## 5 Metodología

En visión artificial, el reconocimiento de objetos es considerado como una tarea muy importante. El diseño de sistemas de reconocimiento de objetos es comúnmente muy complicado. La fiabilidad y la eficiencia del sistema de reconocimiento de objetos dependen directamente de cada uno de los pasos llevados a cabo, desde las técnicas de pre-procesamiento utilizadas, métodos de segmentación, características extraídas y clasificadores utilizados para el reconocimiento. En este Capítulo, la metodología llevada a cabo se discute en detalle, La Figura 13 muestra el proceso general llevado a cabo.

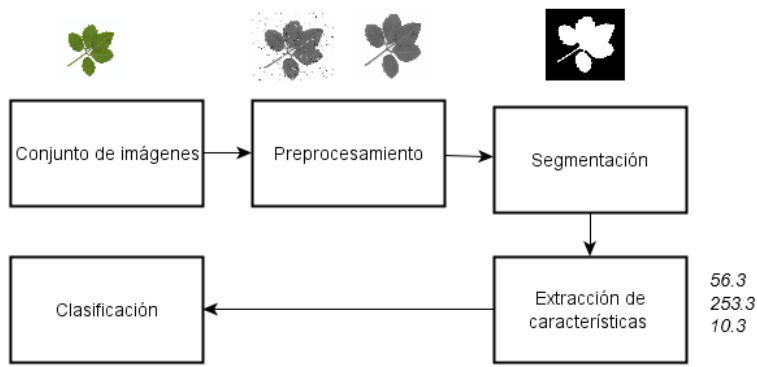


Figura 13. Metodología utilizada en el sistema de identificación.

Comentado [DVL5]:

Las siguientes Secciones describen en detalle cada uno de los pasos empleados en la metodología propuesta.

### 5.1 Conjunto de datos utilizado

En los experimentos realizados, se utilizó el conjunto de imágenes compuesto de 219 clases, el conjunto de datos fue proporcionado por el Departamento de Ciencias de la computación y Tecnología de la Universidad de Tongji en China. El conjunto de datos fue obtenido bajo condiciones específicas en un ambiente completamente controlado. La Figura muestra el conjunto de datos utilizados.



Figura 14. Conjunto de datos proporcionado por el departamento de Ciencias de la computación y Tecnología de la Universidad de Tongji en China.

## 5.2 Pre procesamiento

En este caso, se utilizaron las imágenes originales y se utilizó un filtro Gaussiano de  $3 \times 3$  para quitar algunas imperfecciones o diminutos puntos negros en las imágenes. En los resultados se observó que, los resultados de segmentación no variaron de forma significativa, debido a que las imágenes se tomaron en un ambiente totalmente controlado (imágenes de una sola hoja con fondo blanco).

### 5.3 Segmentación

Una vez pre procesadas las imágenes, estas fueron segmentadas utilizando tres técnicas de segmentación diferentes:

- Segmentación PCA.
- Segmentación OTSU.
- Segmentación K-Means.

La Figura 15 muestra tres imágenes diferentes y las imágenes segmentadas utilizando las técnicas mencionadas. Se observa en la Figura que hay una similitud entre los resultados obtenidos, pero existe un inconveniente cuando se usa segmentación PCA y OTSU en hojas de plantas con formas muy complejas.



Figura 15. Imágenes segmentadas con diferentes técnicas



## 5.4 Extracción de características

Un vector de características es un conjunto de valores o características cuantificables que representan una imagen, este vector es calculado utilizando varias técnicas de extracción de características. Extracción de características se define como el proceso de convertir la imagen en una forma única, distintiva y compacta. Este proceso requiere que la imagen esté segmentada. La extracción de características es un proceso realizado después de la segmentación para obtener las propiedades más importantes de la imagen con el propósito de reconocer el objeto en la imagen. En general, las características se pueden clasificar en el tipo de componente (global o local) o en el tipo de características a extraer (texturales, geométricas o cromáticas): las características locales son las características extraídas a partir de una cierta parte de la imagen. Las características globales se calculan para toda la imagen.

En los experimentos realizados se extrajeron tres tipos de características:

1. Características Cromáticas
2. Características Texturales
3. Características Geométricas

Para las características cromáticas se extrajeron:

- Momentos de Hu con intensidad
- Características Hog piramidales
- Características de Gabor
- Características de Fourier

Para las características texturales se extrajeron:

- Descriptores texturales de Haralick

Para las características geométricas se extrajeron:

- Descriptores de Fourier
- Momentos de Hu
- Momentos y momentos centrales
- Momentos de Gupta
- Características básicas de una imagen (Área, circularidad, rectangularidad, perímetro, número de Euler, etc.)

- Momentos de Flusser
- Características de elipsidad

## 5.5 Clasificación e identificación

En un sistema de reconocimiento de objetos, el clasificador juega un importante papel en el reconocimiento del objeto. La eficiencia y la precisión del sistema dependerá del clasificador utilizado. En los experimentos realizados se utilizaron varios clasificadores y se obtuvo un análisis comparativo de su desempeño.

Los modelos de clasificación utilizados en esta tesis fueron, redes neuronales, SVM, árboles de decisión y NaiveBayes.

Con el objetivo de validar los resultados se utilizó validación cruzada k-fold, con  $k=10$ . La validación cruzada o *cross-validation* (Martin T. Hagan, 1996) es una técnica utilizada para evaluar los resultados de un análisis estadístico y garantizar que son independientes de la partición entre datos de entrenamiento y prueba. Consiste en repetir y calcular la media aritmética obtenida de las medidas de evaluación sobre diferentes particiones. Se utiliza en entornos donde el objetivo principal es la predicción y se quiere estimar que tan preciso es un modelo que se llevará a cabo a la práctica. Es una técnica muy utilizada en proyectos de inteligencia artificial para validar modelos generados. En la validación cruzada de  $k$  iteraciones o k-foldcross-validation los datos de muestra se dividen en  $k$  subconjuntos. Uno de los subconjuntos se utiliza como datos de prueba y el resto ( $k-1$ ) como datos de entrenamiento. El proceso de validación cruzada es repetido durante  $k$  iteraciones, con cada uno de los posibles subconjuntos de datos de prueba. Finalmente se realiza la media aritmética de los resultados de cada iteración para obtener un único resultado.

## 6 Resultados Experimentales

El sistema de reconocimiento propuesto fue implementado utilizando MATLAB y probado con el conjunto de datos definido anteriormente. Los experimentos fueron realizados para probar la eficiencia y precisión del sistema propuesto. En los resultados se obtienen las pruebas con distintos clasificadores como NaiveBayes (NB), Árboles de decisión (DT) Redes neuronales y Máquinas de Vectores de Soporte (SVM).

Las Tablas 1-4 muestran los resultados obtenidos utilizando segmentación por el método de Otsu. Mientras que las Tablas 5-8 muestran los resultados obtenidos utilizando segmentación por el método PCA. Finalmente las Tablas 9-12 muestran los resultados obtenidos utilizando segmentación por el método k-means.

### Otsu

**Tabla 1 Resultados obtenidos con todas las características obtenidas y utilizando el método de segmentación de Otsu**

Clasificador	Acc	F-measure	ROC	Tiempo
NB	91.58	0.779	0.979	42.28
SVM	95.54	0.878	0.986	172.38
NN	92.89	0.917	9.982	1239.62
DT	77.69	0.856	0.977	7.48

**Tabla 2 Resultados obtenidos con características cromáticas y utilizando el método de segmentación de Otsu**

Clasificador	Acc	F-measure	ROC	Tiempo
NB	79.87	0.798	0.994	31.62
SVM	83.72	0.873	0.997	120.37
NN	82.70	0.991	0.850	875.20
DT	72.17	0.710	0.954	8.19

**Tabla 3 Resultados obtenidos con características texturales y utilizando el método de segmentación de Otsu**

Clasificador	Acc	F-measure	ROC	Tiempo
NB	26.58	0.244	0.896	3.97
SVM	62.03	0.625	0.947	139.52
NN	55.26	0.549	0.928	607.39
DT	48.93	0.479	0.88	7.64

**Tabla 4 Resultados obtenidos con características geométricas y utilizando el método de segmentación de Otsu**

Clasificador	Acc	F-measure	ROC	Tiempo
NB	79.70	0.793	0.881	7.51
SVM	87.60	0.904	0.992	154
NN	88.74	0.883	0.983	1062.91
DT	76.57	0.763	0.881	2.37

**PCA**

**Tabla 5 Resultados obtenidos con todas las características obtenidas y utilizando el método de segmentación de PCA**

Clasificador	Acc	F-measure	ROC	Tiempo
NB	88.48	0.885	0.998	30.19
SVM	92.50	0.925	0.999	92
NN	89.52	0.913	0.999	709
DT	79.30	0.786	0.974	7.16

**Tabla 6 Resultados obtenidos con características cromáticas y utilizando el método de segmentación de PCA**

Clasificador	Acc	F-measure	ROC	Tiempo
NB	79.37	0.793	0.994	27.78
SVM	84.59	0.844	0.987	72.79
NN	80.216	0.784	0.973	839.27
DT	73.46	0.725	0.956	6.54

**Tabla 7 Resultados obtenidos con características texturales y utilizando el método de segmentación de PCA**

Clasificador	Acc	F-measure	ROC	Tiempo
NB	40.38	0.40	0.886	8.32
SVM	49.98	0.49	0.899	38.93
NN	56.54	0.56	0.938	583.20
DT	28.22	0.26	0.728	4.02

**Tabla 8 Resultados obtenidos con características geométricas y utilizando el método de segmentación de PCA**

Clasificador	Acc	F-measure	ROC	Tiempo
NB	85.3	0.803	0.981	7.64
SVM	90.27	0.910	0.991	28.15
NN	89.10	0.889	0.985	1017.58
DT	79.77	0.797	0.925	5.82

**K-means**

**Tabla 9 Resultados obtenidos con todas las características obtenidas y utilizando el método de segmentación k-means**

Clasificador	Acc	F-measure	ROC	Tiempo
NB	88.07	0.877	0.991	37.52
SVM	91.39	0.917	0.993	105.3
NN	88.72	0.897	0.997	631
DT	73.59	0.772	0.967	8.32

**Tabla 10 Resultados obtenidos con características cromáticas y utilizando el método de segmentación k-means**

Clasificador	Acc	F-measure	ROC	Tiempo
NB	79.6	0.82	0.985	48.5
SVM	81.5	0.85	0.991	102.31
NN	80.7	0.84	0.987	837.52
DT	71.28	0.71	0.952	7.56

**Tabla 11 Resultados obtenidos con características texturales y utilizando el método de segmentación k-means**

Clasificador	Acc	F-measure	ROC	Tiempo
NB	49.72	0.537	0.899	8.37
SVM	60.41	0.554	0.926	113.7
NN	51.65	0.612	0.845	962.71
DT	47.33	0.501	0.748	5.61

**Tabla 12 Resultados obtenidos con características geométricas y utilizando el método de segmentación k-means**

Clasificador	Acc	F-measure	ROC	Tiempo
NB	85.58	0.823	0.975	9.21
SVM	87.04	0.874	0.983	153
NN	88.75	0.882	0.986	930.39
DT	77.38	0.778	0.968	3.71

## 7 Discusión y conclusiones

En esta tesis, se presenta un análisis de los tiempos de entrenamiento de distintos clasificadores para la identificación de plantas a partir de hojas. En el desarrollo e implementación del sistema de visión propuesto, se extrajeron diferentes características una vez segmentada la imagen. En esta tesis no se prevé el uso de algoritmos de selección de características para reducir el tiempo de entrenamiento y mejorar la precisión de clasificación. Las características extraídas son utilizadas para entrenar el clasificador. En los resultados experimentales se utilizaron 4 clasificadores para comparar los resultados obtenidos con el clasificador Bayesiano.

En los experimentos llevados a cabo se concluye que el tiempo de entrenamiento del clasificador Bayesiano es menor (promedio de tiempo: 21.9092 segundos) que el tiempo llevado por las NN (promedio de tiempo: 857.9825 segundos) y SVM(107.7042 segundos) en todos los casos y ligeramente mayor que los arboles de decisión (tiempo promedio: 6.2017 segundos). Sin embargo la precisión del clasificador Bayesiano (promedio de precisión: 72.8525) es mejor que los arboles de decisión (promedio de precisión: 67.1408).

El principal reto futuro en reconocimiento automático de hojas es desarrollar métodos computacionales que aprendan a distinguir entre una enorme cantidad de clases a partir de ejemplos, algunos de los problemas a los que se enfrentan los sistemas de identificación de plantas actuales son:

- **Integración de conocimiento:** Los sistemas de visión artificial de la siguiente generación deberán integrar capacidades que normalmente son asociadas solo a personas como: detectar la clase de una planta entre diferentes que son muy parecidas entre sí. Agregar conocimiento extra a las características extraídas de la hoja (humedad natural del lugar, planta de temporal o de riego, etc.)
- **Calidad de la imagen:** La calidad de imagen de la hoja capturada tiene un rol muy importante en la precisión de identificación de la hoja. En general, las imágenes son afectadas por tres factores que son contraste, ruido y emborronamiento en imágenes. La presencia de estos tres factores tiene un impacto directo en el desempeño del proceso
- **Variación de la hoja:** Las hojas de plantas de la misma clase pueden tomar un gran número de variaciones biológicas. Es necesario desarrollar procesos

para el análisis e identificación de estos escenarios. Algunos algoritmos para selección de características que mejor influyan en la identificación de estas hojas han sido propuestos, sin embargo, las técnicas futuras deberán distinguir entre hojas similares y obtener las características más discriminantes para evaluar estos conjuntos de datos.

## **Glosario**

**ACP.** Análisis de Componentes Principales

**RNA.** Red Neuronal Artificial

## Referencias

- Cuevas, E., Zaldívar, D., & Pérez Cisneros, M. (2010). *Procesamiento Digital de Imágenes con Matlab y Simulink*. México, D.F.: Alfaomega Grupo Editor.
- Deb, K., Khan, I. M., Saha, A., & Jo, K.-H. (2012). An Efficient Method of Vehicle Licence Plate Recognition Based on Sliding Concentric Windows and Artificial Neural Network. *Procedia Technology* 4, 812-819.
- Du, S., Ibrahim, M., Shehata, M., & Badawy, W. (2013). Automatic License Plate Recognition (ALPR): A State of Art Review. *IEEE Transactions on Circuits and Systems for Video Technology*, 23(2), 311-325.
- Giannoukos, I., Anagnostopoulos, C.-N., Loumos, V., & Kayafas, E. (2010). Operator context scanning to support high segmentation rates for real time license plate recognition. *Pattern Recognition*, 3866-3878.
- Gonzalez, R. C., & Woods, R. E. (s.f.). *Digital Image Processing*. Pearson.
- Hu, M. (1962). Visual pattern recognition by moment invariant. *IRE Transactions on Information Theory*, 179-187.
- Jiao, J., Ye, Q., & Huang, Q. (2009). A configurable method for multi-style plate recognition. *Pattern Recognition* 42, 358-369.
- Kocer, H. E., & Cevik, K. K. (2011). Artificial neural networks based vehicle licence plate recognition. *Procedia Computer Science* 3, 1033-1037.
- Otsu, N. (1979). A threshold selection method from gray-level histograms. *IEEE Transactions on systems, man and cybernetics*, SMC-9(1), 62-66.
- Russ, J. C. (2011). *The Image Processing Handbook*. Boca Raton, FL: CRC Press.
- Sedighi, A., & Vafadust, M. (2011). A new and robust method for character segmentation and recognition in license plate images. *Expert Systems with Applications*, 38(11), 13497-13504.
- Zhang, Y., & Zhang, C. (2003). A New Algorithm for Character Segmentation of License Plate. *IEEE Proceedings of the Intelligent Vehicles Symposium*, (págs. 106-109).



- Zhang, Z., & Wang, C. (2012). The research of Vehicle Plate Recognition Technical Based on BP Neural Network. *AASRI Precedia 1*, 78-81.
- Zheng, L., He, X., Samali, B., & Yang, L. (2013). An Algorithm for accuracy enhancement of licence plate recognition. *Journal of Computer and Systems Sciences*, 79, 245-255.
- Zhu, Y., Huang, H., Xu, Z., He, Y., & Liu, S. (2011). Chinese-style Plate Recognition Based on Artificial Neural Networks and Statistics. *Procedia Engineering 15*, 3556-3561.
- Deb, K., Khan, I. M., Saha, A., & Jo, K.-H. (2012). An Efficient Method of Vehicle Licence Plate Recognition Based on Sliding Concentric Windows and Artificial Neural Network. *Procedia Technology 4*, 812-819.
- Du, S., Ibrahim, M., Shehata, M., & Badawy, W. (2013). Automatic License Plate Recognition (ALPR): A State of Art Review. *IEEE Transactions on Circuits and Systems for Video Technology*, 23(2), 311-325.
- Giannoukos, I., Anagnostopoulos, C.-N., Loumos, V., & Kayafas, E. (2010). Operator context scanning to support high segmentation rates for real time license plate recognition. *Pattern Recognition*, 3866-3878.
- Gonzalez, R. C., & Woods, R. E. (s.f.). *Digital Image Processing*. Pearson.
- Hu, M. (1962). Visual pattern recognition by moment invariant. *IRE Transactions on Information Theory*, 179-187.
- Jiao, J., Ye, Q., & Huang, Q. (2009). A configurable method for multi-style plate recognition. *Pattern Recognition 42*, 358-369.
- Kocer, H. E., & Cevik, K. K. (2011). Artificial neural networks based vehicle licence plate recognition. *Precedia Computer Science 3*, 1033-1037.
- Otsu, N. (1979). A threshold selection method from gray-level histograms. *IEEE Transactions on systems, man and cybernetics, SMC-9*(1), 62-66.
- Russ, J. C. (2011). *The Image Processing Handbook*. Boca Raton, FL: CRC Press.
- Sedighi, A., & Vafadust, M. (2011). A new and robust method for character segmentation and recognition in license plate images. *Expert Systems with Applications*, 38(11), 13497-13504.

Zhang, Y., & Zhang, C. (2003). A New Algorithm for Character Segmentation of License Plate. *IEEE Proceedings of the Intelligent Vehicles Symposium*, (págs. 106-109).

Zhang, Z., & Wang, C. (2012). The research of Vehicle Plate Recognition Technical Based on BP Neural Network. *AASRI Precedia 1*, 78-81.

Zheng, L., He, X., Samali, B., & Yang, L. (2013). An Algorithm for accuracy enhancement of licence plate recognition. *Journal of Computer and Systems Sciences*, 79, 245-255.

Zhu, Y., Huang, H., Xu, Z., He, Y., & Liu, S. (2011). Chinese-style Plate Recognition Based on Artificial Neural Networks and Statistics. *Procedia Engineering 15*, 3556-3561.