

# UNIVERSIDAD AUTONOMA DE BAJA CALIFORNIA

FACULTAD DE CIENCIAS



## **Las Redes Neuronales Artificiales en el Reconocimiento de Imágenes.**

Caso de estudio: Identificación Automática de Matrículas de Automóvil

**TESIS**

Que para obtener el título de

**Licenciado en Ciencias Computacionales**

presenta:

**José María Ramírez Martínez**

Ensenada, Baja California. Octubre de 2001

UNIVERSIDAD AUTONOMA DE BAJA CALIFORNIA

FACULTAD DE CIENCIAS

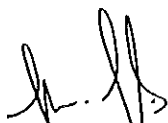
Las Redes Neuronales Artificiales en el Reconocimiento de Imágenes  
Caso de estudio: Identificación Automática de Matrículas de Automóvil

TESIS PROFESIONAL

Que presenta:

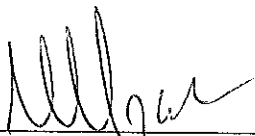
José María Ramírez Martínez

Aprobado por:



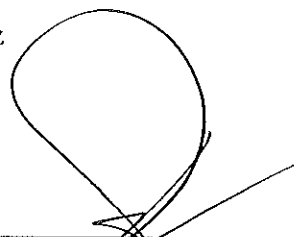
---

M.C. José Ignacio Ascencio López  
Presidente del jurado



---

M.C. María Victoria Meza Kubo  
Secretario



---

Fís. Francisco Juárez García  
1er. vocal

## AGRADECIMIENTOS

Mediante la presente me gustaría expresar profundamente mi agradecimiento a todas aquellas personas que de una u otra forma me han apoyado durante toda mi vida:

Primero que a nadie a mis padres, **Manuel Ramírez** (mi Gran Maestro de la vida, todo lo que tú me has enseñado nunca será enseñado en un salón de clases ni se puede aprender de un libro, cuando se llegue mi turno de ser padre espero ser como tú) y **Ma. de la Luz Martínez** (haces honor a tu nombre siendo la Luz que guía mi camino, junto con mi papá has trabajado y te has esforzado al máximo para sacarnos adelante a pesar de las adversidades, a mis hermanos y a mi), a quienes les debo antes que nada la vida ( y mi carrera ☺ ), y que siempre me han apoyado en cualquier decisión que tome, siempre y cuando sea la correcta, ustedes son mi fuente de inspiración. A mi hermana **Alejandra** (eres mis segunda madre, te agradezco mucho tu apoyo durante mi etapa de crisis, gracias por tus consejos). A mi hermano **Héctor** (sé que también siempre me has apoyado, me gustaría regresar el tiempo para poder cambiarlo y pasar aquellos momentos juntos como hermanos, apoyándonos el uno al otro). A mis sobrinos Paul, Denisse, y Héctor Ivan (el Bebé). A mi mejor amigo de toda la vida, **José Antonio León** (Toño, eres el mejor amigo que he tenido y tú lo sabes, gracias primero que nada por ser mi amigo, creo que con más de 10 años de amistad ya eres como mi tercer hermano, también gracias por tu apoyo durante el tiempo que traía mi autoestima en la suela de mis zapatos ☺, tú me demostraste lo que en realidad significa la palabra “Amistad”, ). A final de cuentas vencí el obstáculo. A ustedes dedico este trabajo.

También a las personas que me han brindado apoyo y amistad durante mi formación profesional:

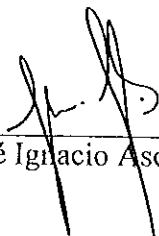
A mi asesor de tesis, el **M.C. José Ignacio Ascencio López**, le agradezco todas las facilidades brindadas, así como su tiempo dedicado a mi persona durante el desarrollo de éste trabajo de investigación, gracias por todo, he disfrutado mucho el trabajar con usted por más de

dos años. A todos los maestros que hicieron su contribución compartiendo sus conocimientos durante mi formación profesional. A las secretarías de la Facultad: Claudia, Bertha, Lulú y Noemí (mis clientas consentidas ☺, si no fuera por ustedes, no hubiera tenido tanto éxito en el negocio, les debo un descuento ☺). A todos mis compañeros de la **XVII generación de Licenciados en Ciencias Computacionales**, en especial a mis dos compañeros de equipo por tres años y medio, los dos tercios restantes del “Trío Dinámico”, también conocido como CeCheDa (tenemos otro nombre de equipo, pero ese no lo puedo escribir aquí ☺): **Daniel Peralta** (el primer LCC y único mérito escolar de la generación) y **César Moreno** (César, eres el siguiente ☺), dos de los mejores compañeros y amigos que he tenido, gracias por haberme dado la oportunidad de haber trabajado con ustedes, disfrute mucho las desveladas (y las desvariadas ☺) que pasamos trabajando (en especial cuando se me olvidó el teclado de la computadora, y las del último semestre). A mis mejores amigas de la carrera: **Alina, Areli, y Cristina**, ustedes son las chicas más lindas que conocí durante la carrera (nunca cambien!), gracias por su amistad.

**RESUMEN** de la tesis de **José María Ramírez Martínez** presentada como requisito parcial para la obtención de la Licenciatura en **Ciencias Computacionales**. Ensenada, Baja California, México. Octubre del 2001.

**Las Redes Neuronales Artificiales en el Reconocimiento de Imágenes.**  
**Caso de estudio: Clasificación Automática de Matrículas de Automóvil.**

Resumen aprobado por:



---

M.C. José Ignacio Ascencio López

Uno de los objetivos de los estudiosos de las ciencias computacionales es el dotar a la computadora de la capacidad para adoptar un comportamiento similar al humano, interactuando con los objetos que la rodean y tomando decisiones a partir de ciertos eventos con la mínima intervención de un intermediario (un usuario), dando a la máquina sentidos artificiales, como la visión y la audición. La visión artificial se apoya en diferentes técnicas de procesamiento digital de imágenes para lograr que una computadora sea capaz de distinguir diferentes figuras.

Las etapas más generales para el procesamiento de imágenes son la adquisición, el preprocesamiento, la segmentación, la codificación y el reconocimiento. En el presente trabajo se muestran las técnicas que resultaron las más apropiadas para realizar nuestra tarea en particular. En la segmentación se analiza la imagen frontal de un vehículo, se localiza y extrae la región que contiene la placa, utilizando como criterios de umbralización la información acerca del color y áreas de forma rectangular. Una vez extraída la placa se obtienen las regiones con los caracteres impresos, (también utilizando la umbralización y el emparejamiento de plantillas). En la etapa de codificación se utiliza un esquema propietario basado en una malla que se coloca sobre el caracter y se analiza la vecindad de píxeles que cubre cada celda de la malla, y un esquema adicional donde se obtiene información de los hoyos que puede tener un caracter. Por último, el reconocimiento se lleva a cabo utilizando un conjunto de redes neuronales artificiales que son entrenadas utilizando el algoritmo de retropropagación, así como un conjunto de Multiperceptrones.

**ABSTRACT** of the thesis of **José María Ramírez Martínez**, presented as a partial requirement to obtain the Bachelors Degree in **Computer Sciences**. Ensenada, Baja California, México. October 2001.

**Artificial Neural Networks in the Image Recognition.**  
**Case study: Automatic Automobile License Plate Classification.**

One of the main goals of the computer scientists is to give the computer the ability to acquire a behaviour similar to that of the humans, interacting with the objects surrounding it and making decisions given some certain events with the minimum intervention of an intermediary (a user), giving the machine artificial senses, like vision. Machine Vision relies on different digital image processing techniques to make a computer able to distinguish different shapes.

The digital image processing steps are the acquisition, preprocessing, segmentation, coding and recognition. In this thesis the most suitable techniques to accomplish our particular tasks are shown. In the segmentation stage, an image containing the frontal image of a vehicle is analyzed, the region with the license plate is obtained using as thresholding criteria color and area information. Once the plate is extracted, the regions containing the characters are obtained (using thresholding and pattern matching). In the coding stage a mesh based scheme is used, the mesh is placed on the character and the pixels in the neighborhood covered by each cell of the mesh is analyzed, a secondary scheme is used, where information about the holes a character may have is calculated. In the last step, the recognition is accomplished using a set of Artificial Neural Networks trained by backpropagation and a subset of Multiperceptrons.

# Índice de contenido

página

1. Introducción.....	1
2. Antecedentes.....	6
3. Segmentación.....	10
3.1 Técnicas de segmentación.....	13
3.1.1. Detección de discontinuidades.....	14
3.1.1.1. Detección de puntos.....	15
3.1.1.2. Detección de líneas.....	15
3.1.1.3. Detección de bordes.....	16
3.1.1.4. Detección combinada.....	19
3.1.2. Segmentación utilizando condiciones de similitud.....	21
3.1.2.1. Umbralización.....	21
3.2. La segmentación en el caso de estudio.....	24
3.2.1. Localización y segmentación de la placa.....	25
3.2.2. Resultados obtenidos en el primer paso de la segmentación.....	32
3.2.3. Extracción de los caracteres contenidos en la placa.....	32
3.2.4. Resultados obtenidos en la segmentación de caracteres.....	36
4. Descriptores.....	38

4.1. Algunos descriptores del borde de una región.....	40
4.1.1. Descriptores sencillos.....	40
4.1.2. Códigos cadena.....	40
4.1.3. Descriptores de Fourier.....	42
4.2. Algunos descriptores de una región.....	43
4.2.1. Descriptores sencillos.....	43
4.2.2. Descriptores topológicos.....	44
4.2.3. Momentos.....	46
4.3. Descriptores utilizados en el caso de estudio.....	49
4.3.1. Esquema de descripción de la región del caracter.....	49
4.3.2. Esquema de descripción secundario.....	50
4.3.2.1. Obtención de la información de los hoyos.....	51
5. Reconocimiento.....	54
5.1. Objetos, patrones y clases.....	54
5.2. Espacio de características.....	55
5.3. Funciones de decisión.....	56
5.4. Clasificador de distancia mínima.....	57
5.5. Emparejamiento mediante correlación.....	58
5.6. Clasificador de Bayes.....	60
5.7. Una técnica alternativa de reconocimiento: Las redes neuronales artificiales.....	65

5.7.1. Definición de red neuronal artificial.....	66
5.7.2. Características de las redes neuronales artificiales....	67
5.7.3. Aprendizaje.....	68
5.7.4. Procesamiento de la información.....	69
5.7.5. Ventajas de las redes neuronales artificiales sobre las técnicas tradicionales.....	70
5.7.6. Aplicaciones de redes neuronales utilizando una computadora secuencial.....	71
5.7.7. Las redes neuronales utilizadas en el caso de estudio.	72
6.Resultados y Conclusiones.....	88
6.1. Etapa de segmentación.....	88
6.2. Descripción.....	89
6.3. Reconocimiento.....	90
6.4. Proceso automático de reconocimiento .....	90
6.5. Conclusiones.....	95
6.6. Aportaciones y trabajo futuro .....	89

# Índice de figuras

página

1. Algunas imágenes adquiridas.....	11
2. Máscara para encontrar un punto.....	14
3. Conjunto de máscaras para detectar líneas en diferentes direcciones.....	15
4. Operadores de Sobel.....	18
5. Máscara del Laplaciano.....	18
6. Proyección de Z sobre W1.....	20
7. Proyección de Z sobre el espacio determinado por W1, W2 y W3.....	22
8. Histograma bimodal.....	23
9. Histograma trimodal.....	23
10. Cubo RGB.....	27
11. Ejemplo de umbralización con el color amarillo.....	28
12. Binarización mediante umbralización de varias variables.....	29
13. Eliminación de hoyos y partículas.....	30
14. Eliminación de regiones y emparejamiento de platillas.....	31
15. Extracción de la placa.....	32
16. Placas segmentadas automáticamente.....	34
17. Binarización de una placa.....	35
18. Ilustración del emparejamiento de plantillas para encontrar una N en la placa.....	36
19. Introducción de ruido durante la binarización de una placa.....	37
20. Esquema de numeración 4 y 8 vecinos.....	41
21. Codificación de un borde.....	41
22. Representación de un borde como una secuencia de números complejos.....	43
23. Región con dos hoyos.....	45
24. Dos componentes conectados.....	45

25. Regiones con Número de Euler igual a 0 y -1.....	46
26. Caracter con malla sobrepuesta.....	50
27. Obtención del hoyo de un dígito.....	52
28. Obtención de la correlación de $f(x, y)$ y $w(x, y)$ en el punto $(s, t)$ .....	59
29. Funciones de densidad de probabilidad para dos clases de patrones unidimensionales.....	62
30. Arquitectura de una red de flujo hacia adelante.....	68
31. Arquitectura de la red para el reconocimiento de dígitos.....	76
32. Proceso de reconocimiento.....	93

# Indice de Tablas

página

I. Resultados obtenidos con redes de una capa oculta para la clasificación de dígitos.....	77
II. Resultados obtenidos con redes de dos capas ocultas para la clasificación de dígitos.....	78
III. Matrículas de vehículos familiares.....	81
IV. Matrículas de vehículos de carga.....	83
V. Resultados del análisis del desempeño de redes con una capa oculta para la clasificación de letras.....	86
VI. Placas segmentadas y matrículas correspondientes reconocidas.....	94

## **1. Introducción**

En los últimos años la computadora se ha convertido en una herramienta insustituible. Ha sido tal su impacto que hoy en día es indispensable en la vida del hombre moderno, ya que mediante su uso se puede auxiliar para realizar un sinnúmero de tareas (que van desde simple procesamiento de texto, hasta difíciles cálculos matemáticos, entre otras cosas), y ahora mediante internet es posible llevar a cabo diversas actividades sin necesidad de salir del hogar (hacer compras, realizar transacciones bancarias, incluso tener su propio negocio).

Como es sabido, la computadora no puede realizar estas actividades por sí sola, requiere de programas y sistemas de cómputo para llevarlas a cabo. Uno de los aspectos más importantes de estos sistemas de cómputo es su interfaz. En los primeros años de existencia de las computadoras, carecían del equipo necesario y de la velocidad suficiente para poder ejecutar programas de cómputo sofisticados, y a causa de la falta de sistemas de despliegue apropiados (tarjetas de video y monitores potentes), limitaba al usuario a entrar en contacto con el sistema mediante una simple y rústica línea de comandos, siendo este tipo de interfaz muy cercana a la naturaleza del equipo, obligando a las pocas empresas que contaban con computadoras a que los usuarios de éstas fueran gente con conocimientos especializados. A medida que se obtuvieron avances tecnológicos en cuanto a equipo computacional, las interfaces de líneas de comandos se convirtieron en interfaces más vistosas y amigables a los usuarios (con menús, cuadros de diálogo, etc).

En la última década, los sistemas computacionales han evolucionado, hasta convertirse en sistemas más efectivos y con interfaces mucho más atractivas (las interfaces gráficas de usuario, GUIs por sus siglas en inglés), las cuales no requieren de una persona especializada para su manipulación, pues estas interfaces gráficas cuentan con botones o iconos en donde se muestra una imagen alusiva a la tarea que se realiza al presionarlo, comprobándose así la frase

“una imagen vale más que mil palabras”. Podemos mencionar como ejemplos de un sistema de este tipo a Xwindows y Windows, que permiten llevar a cabo tareas del sistema operativo (Unix y MS-DOS respectivamente), como manipulación de archivos, directorios, etc., de una forma gráfica, sencilla y más rápida. Otro ejemplo del que podemos hablar son las hojas de cálculo (por ejemplo EXCEL de Microsoft), que podríamos decir que reemplazó al sistema de hojas de cálculo Lotus. Así, es probable que los sistemas vayan evolucionando más y más junto con la tecnología y en un futuro no muy lejano el usuario tendrá un contacto mínimo con ellos, es decir, la computadora podrá capturar por sí sola la información relevante para llevar a cabo su trabajo, tal vez mediante “sentidos artificiales” como cámaras de video, o digitalizadores (scanners), en el caso de visión artificial. O un micrófono, en el caso de audición. En cuanto a visión artificial, en la actualidad existen sistemas capaces de capturar documentos, sin necesidad de que el usuario los transcriba. El usuario simplemente digitaliza el documento mediante un scanner, y el sistema se encarga de pasarlo a un formato electrónico, estos sistemas son llamados Reconocedores Ópticos de Caracteres (OCR, por sus siglas en inglés).

Las características de la computadora la hacen especial, pues tiene ventajas sobre algunas limitaciones humanas: puede calcular millones de operaciones de punto flotante en un segundo. Pero estas máquinas solamente superan a los sistemas neuronales biológicos en este tipo de operaciones, las computadoras digitales más rápidas realizan cerca de  $10^9$  operaciones por segundo, el cerebro de una simple mosca lleva a cabo  $10^{11}$  operaciones sencillas al estar descansando [Churchland y Sejnowski, 1992].

A una computadora, a pesar de lo maravillosa que es, le es imposible realizar ciertas tareas, que para los humanos, por nuestra naturaleza nos resultan sencillas, como el razonar, intuir, o reconocer una persona aún cuando haya pasado tiempo sin verla. Estas son algunas ventajas que tenemos sobre las computadoras, las cuales realizamos de forma tan natural que no

nos preocupamos ni tomamos en cuenta el complicado proceso que se lleva a cabo en nuestro cerebro, el cual, haciendo una analogía inversa, podríamos decir que es nuestra unidad de procesamiento, o más bien, nuestra supercomputadora, ya que cuenta con millones de unidades de procesamiento (nuestras neuronas).

Día con día, investigadores se esfuerzan para lograr más avances tecnológicos, y así poder obtener la siguiente generación de computadoras: computadoras con máquinas procesadoras más pequeñas, rápidas y potentes que las actuales, que puedan procesar cantidades masivas de información y que aprendan, escuchen y piensen, con las que podamos comunicarnos e interactuar como si de un humano se tratara. Pero para crear este tipo de máquinas “parecidas al cerebro”, teorías, tecnologías y arquitecturas como las siguientes son requeridas [Kartalopoulos, 1996]:

- Teorías que expliquen lo que es la inteligencia, cómo procesa información imprecisa, y almacena, recuerda, asocia, correlaciona, infiere y extrae valores imprecisos.
- Tecnologías que con una cantidad de circuitería relativamente pequeña pueda procesar grandes cantidades de información imprecisa en un tiempo muy corto y dé resultados precisos.
- Arquitecturas que contengan las teorías y tecnologías anteriores.

En otras palabras, para poder construir dispositivos que simulen tareas del cerebro es necesario entender y conocer con precisión cómo el cerebro lleva a cabo dichas tareas, y para poder lograrlo se debe entender la arquitectura y funcionamiento de éste [Churchland y Sejnowski, 1992].

Hasta el momento no se ha logrado decifrar los procesos cerebrales en su totalidad, pero sí se han logrado avances en el entendimiento de los sistemas sensoriales, específicamente en

el sistema visual. En el campo de la ciencia computacional, los investigadores han intentado imitar algunos de los procesos de la naturaleza, por ejemplo, la evolución (mediante algoritmos genéticos), o los procesos cognitivos y sensoriales (mediante la Inteligencia Artificial). En cuanto a la simulación de procesos sensoriales, se han dado los primeros pasos, en el reconocimiento de formas, tratando de simular con una computadora y un sistema de adquisición de imágenes a los componentes del sistema visual humano: el ojo (con sus partes esenciales como la retina, la pupila, conos, bastones, etc.) y la región cerebral que se encarga de la interpretación de imágenes. A este proceso se le denomina procesamiento y reconocimiento de imágenes [Hall, 1979].

El procesamiento y reconocimiento de imágenes consta de una serie de pasos o etapas fundamentales, las cuales son:

- La adquisición de imágenes
- El preprocesamiento de las imágenes
- La segmentación
- Representación y descripción de regiones
- El reconocimiento

El tema principal del presente trabajo es el reconocimiento de patrones visuales, en particular de matrículas de automóviles. Aquí se hace mención de cada una de las etapas involucradas en el procesamiento de las imágenes. Sobre cada etapa se presenta la descripción de algunos de los métodos alternativos existentes para llevar a cabo dicha etapa, así como la técnica utilizada en nuestro trabajo de investigación, cómo se realizó, y los resultados obtenidos. Se hace un mayor énfasis en las etapas de segmentación y de reconocimiento, ya que son los temas más importantes de la investigación realizada.

En el capítulo 2 se habla brevemente acerca de las diferentes técnicas de reconocimiento que existen y se presentan ejemplos de proyectos realizados en diferentes partes del mundo donde se han empleado dichas técnicas. En el capítulo tres se da una breve introducción al procesamiento de imágenes y se habla de la primer etapa del mismo, la adquisición de imágenes, qué tiene por objetivo y cómo se llevó a cabo, para posteriormente entrar a una de las etapas más interesantes e importantes de cualquier sistema de análisis de imágenes, que es la segmentación. En este capítulo se hace mención de los dos tipos principales de segmentación (detección de discontinuidades y condiciones de similitud entre pixeles), y algunas técnicas que se encuentran dentro de cada tipo, como lo son la detección de bordes en el caso de detección de discontinuidades, y la umbralización en el segundo caso, para posteriormente hablar acerca de cómo se realizó ésta etapa en nuestro caso de estudio. La etapa posterior a la segmentación es la descripción, de la cual se habla en el capítulo cuatro. El quinto capítulo trata acerca del reconocimiento, donde se da una breve introducción al tema, se definen una serie de conceptos importantes, y se habla acerca de las técnicas tradicionales de reconocimiento. También se da una introducción general a las redes neuronales artificiales, tema principal de éste trabajo de investigación, y por último se da una explicación del módulo de reconocimiento implementado. Por último, en el capítulo seis, se hace un análisis de los resultados obtenidos en cada una de las etapas realizadas para llevar a cabo el reconocimiento, así como una serie de conclusiones.

## 2. Antecedentes

Desde el surgimiento del área de reconocimiento de patrones visuales se han utilizado diferentes tipos de técnicas para llevar a cabo la tarea. Algunos de los métodos de reconocimiento más utilizados son el método de correlación o emparejamiento de plantillas (template matching) y el análisis estructural, que fueron los usados en los primeros experimentos para el reconocimiento de caracteres. Una tercera técnica, relativamente más reciente en el campo del reconocimiento de caracteres (se empezó a utilizar a principios de los '80s) [Sinha, 1999], son las redes neuronales artificiales

El método de correlación considera una base de datos de plantillas o figuras predeterminadas, que servirán de modelo para realizar la clasificación de la imagen. Una vez aisladas las regiones a reconocer (mediante la segmentación), se compara con las figuras almacenadas para determinar de qué figura se trata. En el reconocimiento de caracteres por ejemplo, se tendrían almacenadas figuras que representan cada letra y número, al extraerse una región de la imagen, se compara con los caracteres almacenados. En las comparaciones se obtiene una probabilidad mediante la cual se determina a qué carácter se asemeja más la muestra tomada de la imagen. Esta probabilidad se calcula contabilizando los bits que tienen el mismo valor en la misma posición de la muestra y el patrón almacenado. Al dividir esta suma entre el número máximo de puntos de estudio se obtiene un parámetro entre cero y uno que nos da una idea de la bondad del emparejamiento [Maravall, 1994].

Con esta técnica de emparejamiento de plantillas, en 1990 y 1991 se desarrolló en la Universidad Politécnica de Madrid, España, un sistema automático para el reconocimiento de matrículas de vehículos industriales [Maravall, 1994]. Según indica la referencia, este proyecto mostró una alta efectividad en el reconocimiento, obteniéndose cerca de un 90% de matrículas

reconocidas correctamente. La referencia que describe este proyecto, lo hace de manera muy somera, concentrándose exclusivamente en los errores en el proceso de reconocimiento, no en el error global (al considerar otros posibles errores como los provocados por una mala segmentación).

El análisis estructural utiliza un árbol de decisión para determinar las características geométricas de la región. Tomando nuevamente como ejemplo el reconocimiento de caracteres, y utilizando los caracteres B, D, 6 y 9. Dos características que pueden ser utilizadas para distinguirlos son el número de hoyos en cada caracter (uno o dos), y la posición vertical de los hoyos (arriba, en el centro o abajo). Dos hoyos apunta hacia una B, y uno solo lleva a la siguiente rama en la estructura del árbol. Un hoyo arriba indica un 9, si está en el centro el caracter es una D, un hoyo abajo significa un 6 [Nelson, 1998].

El reconocimiento de caracteres puede clasificarse en dos categorías [Sinha, 1999]: reconocimiento de caracteres impresos por máquinas, y el reconocimiento de caracteres escritos a mano. Para la primer categoría los métodos clásicos de reconocimiento son una buena opción, ya que de antemano se tiene información acerca de los caracteres (un estilo, una posición y tamaño determinados). Para resolver un problema de la segunda categoría los métodos clásicos no representan una solución viable, debido a la gran variabilidad de estilos de escritura. En los últimos tiempos han aparecido algoritmos neuronales que son hoy en día una excelente alternativa para el reconocimiento de caracteres de cualquiera de las dos categorías.

En 1983 Kunihiko Fukushima [Fukushima, 1983], y colegas, desarrollaron una red neuronal de propósito particular, conocida como Neocognitrón, la cual puede reconocer caracteres escritos, en especial los números arábigos (0, ..., 9). Esta red es insensible a variaciones en la posición y estilos en los cuales esté escrito el caracter. El sistema tiene la habilidad de generalizar, es decir, genera una respuesta correcta a una entrada que no se le ha presentado

anteriormente, y la presentación de sólo algunos ejemplos de patrones es suficiente para entrenarla .

En 1993, Eric W. Brown [Brown, 1993], elaboró un sistema de reconocimiento óptico de caracteres, aplicando una red neuronal de retropropagación como reconocedor de caracteres. La red se entrenó utilizando tres conjuntos de 84 caracteres cada uno, siendo cada conjunto un estilo diferente de caracter. Los caracteres se representaban en matrices de 8x8. En la referencia se indica que ocurrieron dificultades al entrenar la red y que la convergencia fue tardía, utilizando en el entrenamiento la retropropagación estándar. Sin embargo, también se menciona que una vez entrenada la red, ésta generó respuestas de una forma bastante rápida, en comparación a otro sistema de reconocimiento de caracteres desarrollado por el autor, en donde utilizaba técnicas tradicionales para la clasificación, teniendo un diccionario de patrones, y el reconocimiento se llevaba a cabo comparando el caracter a clasificar con cada uno de los almacenados, y haciendo uso de los mismos conjuntos de caracteres utilizados para entrenar la red neuronal. A diferencia del método tradicional, la red neuronal aprendió lo necesario para poder realizar el reconocimiento en su etapa de implementación, aunque con un grado de precisión un poco más bajo que el del método tradicional (54% de caracteres reconocidos correctamente, contra un 72%).

En el Instituto de Tecnología de Massachusetts, se diseñó e implementó un sistema de reconocimiento de dígitos, en 1999. Este sistema reconoce el monto (la cantidad en números) de extracción escrito en cheques brasileños [Sinha, 1999], utilizando como módulo de reconocimiento un arreglo de cuatro redes neuronales artificiales, implementando el modelo del multiperceptrón de varias capas, y entrenándose mediante el algoritmo de retropropagación. El arreglo constaba de dos redes que no utilizaban extracción de descriptores de los dígitos, es decir, se enviaba los mapas de bits (de 16 x 16) obtenidos durante la segmentación directamente

a estas redes. Para las redes restantes, los dígitos eran codificados previamente. El objetivo del arreglo de redes es el incremento de la probabilidad de éxito en el reconocimiento, el primer par de redes (sin codificación del dígito) corrigen errores del segundo par y viceversa. Las redes fueron entrenadas con un conjunto de 3103 dígitos. Se hicieron pruebas con este conjunto y otro aparte de 1444 dígitos, obteniéndose resultados con una tasa de reconocimiento promedio de 82.7%

Las características propias de una red neuronal (aprendizaje adaptativo, la generalización, la tolerancia a fallas) hacen pensar que evidentemente el problema de reconocimiento de caracteres, ya sean impresos por una máquina o escritos por un humano, puede tratarse de forma conveniente con estos algoritmos.

### 3. Segmentación

El procesamiento, en términos generales, se puede definir como una transformación o tratamiento de datos. Tomando en cuenta este concepto, entonces, el procesamiento de imágenes se encarga del tratamiento de los datos que definen a una imagen. Las técnicas de procesamiento de imágenes son utilizadas para el mejoramiento de la información pictórica, para su análisis posterior, ya sea por un humano, o por una máquina [González y Woods, 1992]. Las diferentes etapas previas al reconocimiento de imágenes son, como ya se había mencionado: la adquisición, el preprocesamiento, la segmentación y la extracción de descriptores (o características).

La primera etapa consiste en la adquisición de las imágenes a interpretar. En esta etapa es necesario contar con un sistema de adquisición (como una cámara, un scanner, un microscopio, un satélite etc.) y un dispositivo para transformar la imagen a un formato digital (en caso de que el sistema de adquisición no cuente con uno). En el caso de estudio en particular, la adquisición de imágenes se llevó a cabo utilizando una cámara digital. Se tomaron aproximadamente un total de cien imágenes con una cámara que captura imágenes True Color de 24 bits en formato JPEG, con dimensiones de 640 pixeles de ancho por 480 de alto. Las imágenes capturadas fueron transformadas al formato TIFF (mediante un editor de imágenes), ya que las bibliotecas utilizadas para el procesamiento de imágenes (Matrox Imaging Library), no soporta el formato JPEG, también fueron redimensionadas por razones de economía de recursos (memoria y velocidad de procesamiento, principalmente), quedando en 320 pixeles de ancho por 200 de alto. Las imágenes fueron capturadas en las instalaciones de las casetas de cobro de los Caminos y Puentes Federales, colocando la cámara a 100 metros aproximadamente de retirado de los vehículos. Se decidió tomarlas en este sitio por el hecho que al momento de

pagar, los automovilistas se detienen por un periodo de tiempo de aproximadamente un minuto, tiempo suficiente para capturar la imagen sin problemas. En la figura 1 se muestran algunas de las imágenes capturadas.



Fig 1. Algunas imágenes capturadas



Fig 1. Algunas imagenes capturadas  
(continuación)

Una vez obtenida una imagen a analizar, en nuestro caso en particular, el siguiente paso es la extracción de regiones de interés.

En el reconocimiento o interpretación automático de imágenes, obviamente no es necesario analizar todo lo que se encuentre en la imagen, sino sólo los objetos que sean de interés para la aplicación, por lo tanto es necesario eliminar todo aquello que no se considere de importancia. Este es precisamente el objetivo principal de la segmentación.

La segmentación consiste en particionar la imagen que se está analizando en los diferentes objetos que se encuentran en ella y así obtener las regiones en las que se está interesado [Gonzalez y Woods, 1992]. En el caso de estudio, los objetos de interés que se desean extraer de la imagen son los caracteres (letras, números, etc.) que se encuentran impresos en la placa. La segmentación es uno de los pasos más importantes en el análisis de imágenes, y es una de las tareas más difíciles en el procesamiento de imágenes [Gonzalez y Woods, 1993]. De ella depende el éxito en etapas posteriores. Existen diferentes técnicas para la extracción de regiones en el proceso del reconocimiento de imágenes, a continuación se hace mención de algunas de ellas.

### **3.1 Técnicas de segmentación**

Los diferentes algoritmos de segmentación están basados en una de dos propiedades básicas de los valores de niveles de gris: *discontinuidad* y *similitud*.

En la segmentación utilizando la discontinuidad, la partición se realiza basándose en cambios abruptos de los niveles de gris, considerándose estos cambios como transiciones del fondo de la imagen a un objeto y viceversa. En la segunda categoría la segmentación se lleva a

cabo agrupando píxeles, los cuales su valor de nivel de gris se encuentre en un rango preestablecido, es decir, que tengan un nivel similar hasta cierto grado.

### 3.1.1. Detección de discontinuidades

Existen tres tipos de discontinuidades básicas: *puntos aislados, líneas y bordes*. Estos tipos de discontinuidades se localizan desplazando una máscara sobre la imagen. Una máscara es una matriz (de 3x3, por ejemplo, dependiendo del tamaño de la vecindad que se quiere encerrar) donde cada componente corresponde a un coeficiente que se multiplica por el nivel de gris correspondiente al pixel con el que se traslapa en la imagen. Los resultados de las multiplicaciones de los coeficientes por los niveles de gris son sumados, obteniéndose un valor, llamado *respuesta de la máscara*. Esto es, la respuesta de una máscara sobre un punto en la imagen es

$$R = W_1Z_1 + W_2Z_2 + \dots + W_9Z_9 = \sum W_iZ_i$$

donde  $Z$  es el nivel del pixel asociado con el coeficiente  $W$ . La respuesta de la máscara se define con respecto al pixel central de la vecindad. Los coeficientes de mayor valor son colocados en posiciones donde se desee se tenga un valor de respuesta mayor. Por ejemplo, para detectar un punto, el coeficiente central de la máscara debe ser el mayor, como se muestra en la figura 2.

-1	-1	-1
-1	8	-1
-1	-1	-1

Fig 2. Máscara para encontrar un punto.

### 3.1.1.1. Detección de puntos

Detectamos un punto en una imagen de fondo constante utilizando una máscara similar a la de la figura anterior, y si la respuesta de la máscara cumple con el criterio

$$|R| > T$$

donde T es un umbral no negativo. Lo que se realiza es una medición de las diferencias ponderadas entre el punto central y sus vecinos. El umbral T determina las diferencias que son lo suficientemente grandes como para considerarse el punto como aislado, y con la idea de que un punto aislado tendrá un nivel de gris diferente al de sus vecinos.

### 3.1.1.2. Detección de líneas

Para la detección de líneas, al igual que otro tipo de discontinuidades, se asignan coeficientes mayores en las posiciones de la máscara donde se desea que los puntos de la vecindad que encierra respondan más fuertemente. Así, el siguiente conjunto de máscaras es de utilidad para detectar líneas (de un pixel de grosor) en diferentes direcciones.

-1	-1	-1
2	2	2
-1	-1	-1

-1	-1	2
-1	2	-1
2	-1	-1

-1	2	-1
-1	2	-1
-1	2	-1

2	-1	-1
-1	2	-1
-1	-1	2

Fig 3. Conjunto de máscaras para detectar líneas en diferentes direcciones

Si la primer máscara se pasa sobre una imagen, tendrá una respuesta mayor en puntos que se encuentren sobre líneas horizontales. La segunda responderá con más fuerza en puntos sobre líneas a  $+45^\circ$  de inclinación, y así sucesivamente.

Si  $R_1$ ,  $R_2$ ,  $R_3$  y  $R_4$  denotan las respuestas de cada máscara (de izquierda a derecha, arriba a abajo) y suponiendo que todas son desplazadas sobre una imagen, si en un cierto punto

$$|R_i| > |R_j|$$

para toda  $i$  diferente de  $j$ , se dice que ese punto está más probablemente asociado con una línea en la dirección de la máscara  $i$ . Por ejemplo, si en un punto se tiene que  $|R_1| > |R_j|$ , para  $j = 2, 3, 4$ , se dice que ese punto está más probablemente asociado con una línea horizontal.

### 3.1.1.3. Detección de bordes

Debido a que los puntos aislados y líneas no ocurren con mucha frecuencia en la mayoría de las aplicaciones, la detección de bordes es el método más utilizado para detectar discontinuidades en una imagen [González y Woods, 1992]

La forma más sencilla para distinguir un objeto del fondo donde se encuentra en una imagen es verificar un cambio en los niveles de intensidad, esto es, si se tiene un objeto oscuro sobre un fondo claro se busca una transición de un nivel de intensidad menor a un nivel mayor, habiéndose encontrado así un borde. Un borde se define como una frontera entre dos regiones con niveles de gris relativamente distintos [González y Woods, 1992], estas dos regiones pueden ser el objeto y el fondo de la imagen. Los cambios de intensidad de niveles de gris pueden ser detectados mediante el uso de operadores de derivación. La magnitud de la primer derivada puede ser utilizada para detectar la presencia de un borde, y el signo de la segunda derivada para determinar de qué lado del borde se encuentra un pixel.

### *El Gradiente*

Considerándose una imagen como una función bidimensional  $f(x,y)$ , entonces se puede definir el gradiente de una imagen como

$$\nabla f = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \partial f / \partial x \\ \partial f / \partial y \end{bmatrix}$$

Es bien sabido del análisis vectorial que la dirección del gradiente es la dirección de la máxima tasa de cambio de la función  $f$  en  $(x, y)$ , y la magnitud del gradiente corresponde a la magnitud de esa tasa máxima de cambio. La magnitud del gradiente (o simplemente gradiente), se obtiene mediante la fórmula

$$\nabla f = \text{mag}(\nabla f) = \sqrt{G_x^2 + G_y^2}$$

El valor del gradiente es una cantidad de importancia en la detección de bordes. Este valor se puede aproximar utilizando los valores absolutos de sus componentes, ya que son más sencillos de calcular

$$\nabla f \approx |G_x| + |G_y|$$

Como se había indicado, la detección de discontinuidades se lleva a cabo utilizando máscaras y calculando la respuesta de la vecindad que encierra. Y es mediante esta forma que también el gradiente de una imagen (las derivadas parciales que lo componen) puede ser calculado. Existen varias formas de calcular las derivadas de una imagen digital, pero una de las más utilizadas son los operadores de Sobel [González y Woods, 1992], debido a las propiedades que los caracterizan: aparte del cálculo de las derivadas, da un efecto de suavizado a la imagen. Esta última es la más sobresaliente, ya que las derivadas tienden a realzar el ruido en la imagen. Las máscaras utilizadas como operadores de Sobel se muestran en la siguiente figura.

-1	-2	-1
0	0	0
1	2	1

-1	0	1
-2	0	2
-1	0	1

Fig 4. Operadores de Sobel

La primera es utilizada para calcular la derivada parcial con respecto a x, y la segunda para calcular la parcial con respecto a y. Las derivadas son calculadas mediante las siguientes fórmulas

$$G_x = (Z_7 + 2Z_8 + Z_9) - (Z_1 + 2Z_2 + Z_3)$$

$$G_y = (Z_3 + 2Z_6 + Z_9) - (Z_1 + 2Z_4 + Z_7)$$

Estas máscaras son recorridas en todas las posiciones de los puntos en la imagen que se analiza, al terminar el proceso se tiene una imagen de gradientes (una imagen con los bordes resaltados)

También se puede calcular el laplaciano de una imagen. El laplaciano de una función  $f(x,y)$  es una derivada de segundo orden, definido como

$$\nabla^2 f = \partial^2 f / \partial x^2 + \partial^2 f / \partial y^2$$

Una máscara para aproximar el laplaciano en cada punto de una imagen se muestra en la figura

0	-1	0
-1	4	-1
0	-1	0

Fig 5. Máscara del laplaciano

Y se calcula mediante

$$\nabla^2 f \approx 4Z_5 - (Z_2 + Z_4 + Z_6 + Z_8)$$

Se tienen tales coeficientes en la máscara ya que el coeficiente asociado con el pixel central deber ser positivo, y los asociados con los pixeles exteriores deben ser negativos. Debido a que el laplaciano es una derivada, la suma de sus coeficientes debe ser igual a cero. Por tanto, la respuesta es cero cuando el punto en cuestión y sus vecinos tienen el mismo valor (corresponde a la derivada de una constante).

El laplaciano no es muy recomendable para utilizarse como detector de bordes, ya que es sensible al ruido, produce bordes dobles y no detecta la dirección del borde.

#### 3.1.1.4. Detección combinada

Utilizando las técnicas de detección de los tres tipos básicos de discontinuidades en una imagen proporciona un detector más general, obteniéndose así mejores resultados. En la detección combinada las máscaras y las regiones que estas encierran son representadas como vectores, esto es

$$\mathbf{W} = \begin{bmatrix} W_1 \\ W_2 \\ \vdots \\ W_9 \end{bmatrix} \quad \mathbf{Z} = \begin{bmatrix} Z_1 \\ Z_2 \\ \vdots \\ Z_9 \end{bmatrix}$$

donde  $W_i$  corresponde a los coeficientes de la máscara y  $Z_i$  a los niveles de gris. Utilizando esta notación la respuesta de una máscara es el producto punto de los vectores  $\mathbf{W}$  y  $\mathbf{Z}$ . Así,

$$R = \sum W_i Z_i = \mathbf{W}^T \mathbf{Z}$$

T indica la transposición del vector.

Si se escogen vectores (máscaras) que sean ortogonales y además normalizados (que tengan longitud unitaria), éstas pueden funcionar como la base para formar un espacio. Por

ejemplo, si se tienen tres máscaras,  $W_1$ ,  $W_2$ ,  $W_3$ , de tres componentes cada una y tienen las características mencionadas, entonces forman un espacio tridimensional. Además la respuesta de cada máscara corresponde a la proyección del vector  $Z$  sobre cada eje que éstas forman, es decir, la respuesta de la máscara  $W_1$  es la proyección del vector sobre el eje de esta máscara. Esto es por que

$$R = W_1 \bullet Z = \|W_1\| \|Z\| \cos \theta$$

donde  $\theta$  es el ángulo formado por los dos vectores. Pero como la magnitud de  $W_1$  ( $\|W_1\|$ ) es 1, entonces

$$\|Z\| \cos \theta = W_1^T \bullet Z$$

que es la proyección de  $Z$  sobre  $W_1$ . Como se muestra en la figura 6

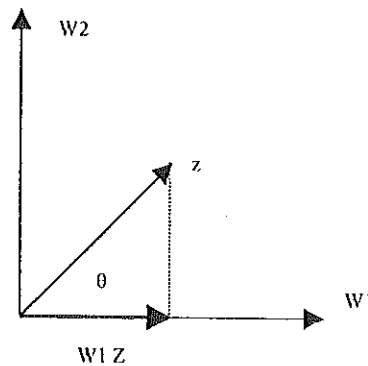


Fig 6. Proyección de  $z$  sobre  $w_1$ .

Suponiendo ahora que dos máscaras,  $W_1$  y  $W_2$ , están definidas para detectar líneas, y la tercera,  $W_3$ , para detectar bordes. Si se desea saber si una región en una imagen corresponde a una línea, se debe proyectar su vector sobre el plano que forman las máscaras para detectar líneas.

Lo mismo ocurre para detectar un borde. Al proyectar el vector sobre el plano, se forma un ángulo ( $\theta$ ) entre éste y su proyección, la misma situación se tiene al proyectar el vector sobre

W3, obteniéndose un ángulo  $\phi$ . Si el vector  $Z$  tiene características de una línea, éste y su proyección formarán un ángulo menor que el que forma con su proyección sobre W3, esto es

$$0 < \phi.$$

En otras palabras, los ángulos formados entre  $Z$  y sus proyecciones determinan si tiene características de una línea o de un borde. Los ángulos se calculan de la siguiente forma:

$$\begin{aligned}\theta &= \arccos \{ \sqrt{[(W1^T Z)^2 + (W2^T Z)^2]} / \sqrt{[(W1^T Z)^2 + (W2^T Z)^2 + (W3^T Z)^2]} \} \\ &= \arccos \{ \sqrt{[(W1^T Z)^2 + (W2^T Z)^2]} / \|Z\| \} \\ \phi &= \arccos \{ \sqrt{[(W3^T Z)^2]} / \sqrt{[(W1^T Z)^2 + (W2^T Z)^2 + (W3^T Z)^2]} \} \\ &= \arccos \{ |W3^T Z| / \|Z\| \}\end{aligned}$$

En la figura 7 se muestra gráficamente la técnica anterior.

El método se generaliza para dimensiones mayores, siempre y cuando las máscaras cumplan con las características mencionadas.

### **3.1.2. Segmentación utilizando condiciones de similitud**

Otras técnicas de segmentación se llevan a cabo utilizando condiciones de similitud entre pixeles. Una de las técnicas más importantes es la umbralización [González y Woods, 1992].

#### **3.1.2.1. Umbralización**

La umbralización consiste en encontrar un valor umbral (de aquí el nombre de la técnica) mediante el cual se puedan diferenciar puntos en la imagen que corresponden al fondo o a objetos. Comparando los puntos en la imagen contra este umbral se toma la decisión si forman parte de un objeto o de otra región de la imagen. Viéndolo de otra forma, si la función

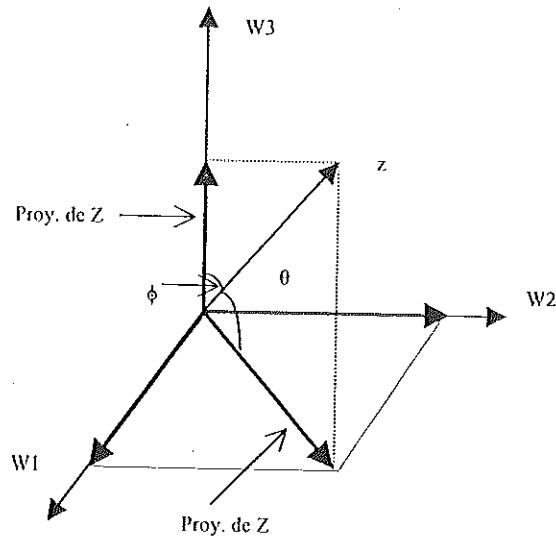


Fig. 7. Proyecciones de Z sobre el subespacio determinado por W1 y W2 , y sobre W3

$f(x, y)$  representa la imagen original y  $g(x, y)$  la imagen umbralizada, entonces

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

Esto es, si la intensidad del punto  $(x, y)$  es mayor al umbral, entonces es parte de un objeto, de otra forma pertenece al fondo (o viceversa).

Del valor del umbral depende la efectividad del método. Una forma de determinar el umbral es analizando el histograma de la imagen. El histograma de una imagen es una función que relaciona un nivel de gris con la cantidad de píxeles que tienen ese nivel de gris. En una imagen donde se tenga un objeto sobre un fondo constante, en el histograma predominarán dos grupos de píxeles, los del fondo y los del objeto. El histograma de dicha imagen tendría la forma como la que se muestra en la figura 8, este tipo de histogramas son llamados *histogramas bimodales*.

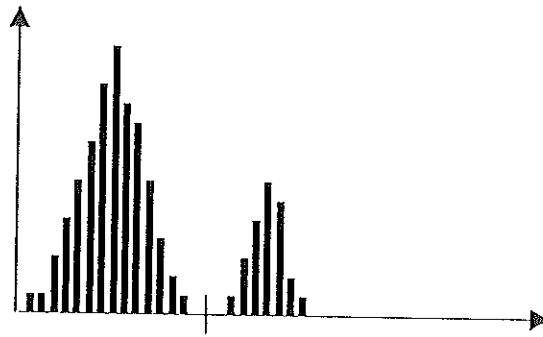


Fig 8. Histograma bimodal.

Se observa claramente que una forma de separar los objetos del fondo de la imagen es seleccionando un umbral que separe las dos regiones del histograma. Existen casos donde el histograma tenga más regiones, como se muestra en la siguiente figura.

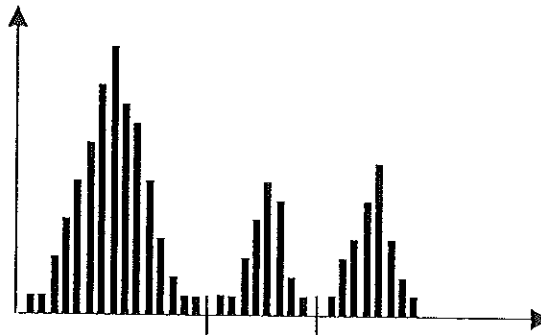


Fig 9. Histograma trimodal.

En la imagen a la que pertenece este histograma puede haber dos tipos de objetos claros sobre un fondo oscuro. En este caso se deben definir dos umbrales para clasificar los puntos. Por ejemplo, si  $T_1 < f(x, y) \leq T_2$ , el pixel pertenece a una clase de objeto, a otra clase si  $f(x, y) > T_2$ , y al fondo si  $f(x, y) \leq T_1$ .

El valor del umbral va a depender del nivel de gris en un punto  $(x, y)$  (en este caso el umbral es llamado *global*). Además del nivel de gris también puede depender de una propiedad local en el punto  $(x, y)$ , por ejemplo el nivel promedio de una vecindad centrada en  $(x, y)$  (aquí el umbral es llamado *local*). O puede depender de las coordenadas espaciales  $(x, y)$  (en este caso se denomina umbral *dinámico*). Así el valor del umbral puede verse como una función de la forma

$$T = T[x, y, p(x, y), f(x, y)]$$

Donde  $f(x, y)$  denota el nivel de gris en el punto  $(x, y)$ , y  $p(x, y)$  es una propiedad local.

### **3.2. La segmentación en el caso de estudio**

En el caso de estudio en particular, la etapa de segmentación se llevó a cabo en dos pasos. El resultado de este proceso, deben ser los caracteres que forman parte de la matrícula del vehículo que se encuentra en la imagen capturada. Para poder obtener estos caracteres, primeramente se debe localizar la región en donde se encuentran la placa del automóvil, y que es el objetivo del primer paso en la segmentación. Para localizar y extraer la imagen con placa se utilizó la información de color, ya que simplifica la identificación y extracción de objetos en una escena [González y Woods, 1992]. Debido a que puede haber regiones con el mismo color de la placa, implicando que se pudiesen obtener más de una región durante este proceso, se utilizó una segunda propiedad, el área de la placa, para distinguirla de aquellas regiones que correspondan a otro tipo de objetos.

Una vez obtenida la placa, se trabajó sobre ella para localizar el objetivo principal, que son los caracteres que componen la matrícula impresa en ella. Esta tarea se realizó en el segundo paso de la segmentación. La extracción de caracteres se realizó de forma similar a la extracción de la placa, se tomó en cuenta el color de los caracteres para localizarlos, y de

acuerdo a ciertas observaciones realizadas ( en base de algunas características de los vehículos, el departamento vehicular del estado les asigna un tipo de matrícula, por ejemplo, para automóviles familiares, como autos compactos, las placas fronterizas inician con una secuencia de tres dígitos, y enseguida un conjunto de tres letras, donde invariablemente la primera es la letra 'N'. Para vehículos de carga, la matrícula inicia con un conjunto de tres letras, de las cuales la primera resulta ser una 'Z') que facilitó la localización de un caracter y a partir de ahí detectar los caracteres restantes. A continuación se explica de forma detallada todo el proceso.

### **3.2.1. Localización y segmentación de la placa**

Los modelos de color son esquemas de representación de las percepciones visuales que que el ojo humano es capaz de distinguir a partir de un conjunto de colores básicos. De acuerdo con estudios hechos con respecto al sistema visual humano, se sabe que existen neuronas especializadas para responder a estímulos provocados por ondas que se encuentran en el rango del espectro electromagnético visible (los conos, contenidos en la retina). Existen tres tipos de conos, unos responden a ondas con longitud de onda correspondiente al rojo, otro tipo de conos presentan excitación al presentarles ondas que definen el color verde, y los últimos responden a estímulos de ondas que se encuentran en el rango del azul [Churchland y Sejnowski, 1992], estos tres colores son llamados *colores primarios*. El ser humano puede percibir todos los colores visibles mediante combinaciones de estos tres colores[González y Woods, 1992], es decir, los tres tipos de conos responden de forma combinada para producir los demás colores. Por ejemplo, al ver un objeto azul, responden un mayor número de conos que se estimulan con este color, y al ver un objeto amarillo, un número de conos rojos y un número igual de conos verdes responden, para percibir el amarillo. De aquí surge el modelo de color RGB (rojo, verde y azul), el cual es utilizado en televisores a color, así como en sistemas de despliegue de

computadoras. Un modelo de color es una especificación de un sistema de coordenadas tridimensional y un subespacio dentro de este sistema donde cada color se representa por un punto [González y Woods, 1992]. En el modelo RGB se tiene un cubo en el espacio tridimensional donde cada uno de sus vértices corresponden a los colores rojo, verde, azul, magenta, amarillo, cian, negro y blanco (ver la figura 10). Las coordenadas de cada punto representan la cantidad con la que contribuye cada color primario para formar el color representado por dicho punto. Este modelo también es utilizado para generar imágenes digitales a color. En este tipo de imágenes, el color de cada pixel que las conforma está definido por los tres componentes RGB, es decir, una imagen a color está compuesta por tres imágenes que corresponden a cada color primario, y un punto en cada una de estas imágenes representa la intensidad con la que contribuye cada componente RGB para formar el color en ese mismo punto en la imagen resultante. Precisamente en la propiedad de color de las imágenes es en la que se basó el módulo de segmentación del sistema de cómputo para llevarse a cabo la localización y extracción de la placa del vehículo en la imagen.

Los colores de una placa del estado de Baja California pueden ser amarillo (placas fronterizas) o blanco (placas nacionales), estos colores fueron tomados en cuenta para la localización de la placa, también se debe considerar el color de las letras y números en la matrícula, el verde. Cabe mencionar que se acotó el dominio del problema tomando en cuenta solamente placas fronterizas. En base a estos colores se analizó la imagen para determinar las regiones donde sea probable se encuentre localizada la placa. Para llevar a cabo esto primeramente se tomó cada uno de los componentes del color correspondiente de cada pixel en la imagen para posteriormente determinar aquellos que formen parte de la región buscada. Para determinar si un pixel es de color blanco, amarillo o verde, se compararon los componentes RGB contra ciertos valores umbrales, y en base a esa condición se tomó la decisión de

pertenencia a la región de interés. Si ubicamos un valor de color (un punto) en el espacio tridimensional RGB y medimos su distancia al color con que se compara, podemos determinar en base a la umbralización si este color es algún matiz de amarillo por ejemplo, para ser considerado un posible componente de la placa. En la figura 11 se da un ejemplo donde se muestra la frontera dentro de la cual deben estar los valores de los pixeles para poderse considerar como amarillos. El método descrito es llamado umbralización de varias variables [González y Woods, 1992], ya que cada componente es comparado contra un umbral para considerarse de importancia o no. En la figura 12 se muestra un ejemplo de umbralización, la entrada al módulo de segmentación es la imagen a color, y mediante el proceso de umbralización se binariza la imagen.

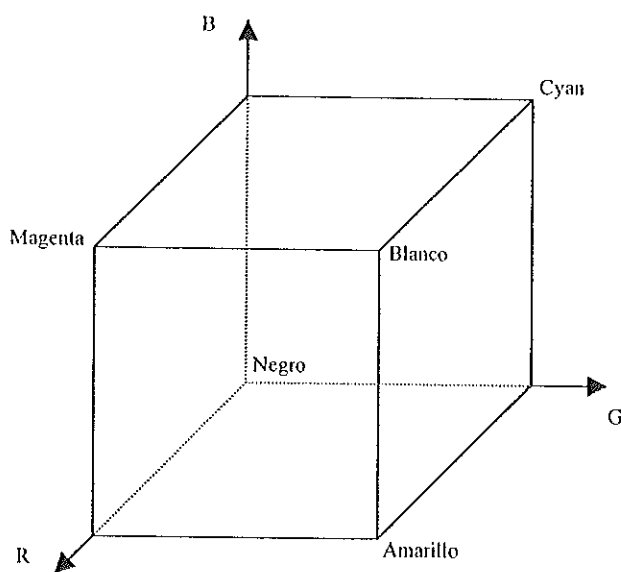


Fig 10. Cubo RGB.

Una vez obtenida la imagen binaria se procedió a eliminar pequeños hoyos que tienen un cierto radio (en este caso dos pixeles), para rellenar o conectar pixeles que pertenecen a una región y de alguna manera durante la umbralización se separaron. También se remueven partículas (puntos aislados), ya que son puntos que no pertenecen a ninguna región en

particular. Esto se lleva a cabo mediante funciones morfológicas de cerrado (closing) y apertura (opening). En la figura 13 se ilustra éste proceso.

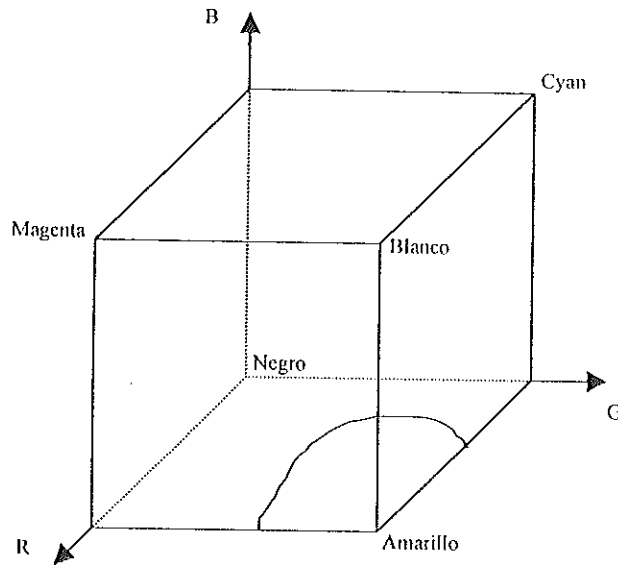


Fig 11 Ejemplo de umbralización con el color amarillo. Los puntos que se encuentren dentro de la frontera son tomados en cuenta. De forma similar se tratan los colores restantes

En algunos casos, se encontraron más de una región que cumplían con las condiciones anteriores (una región se considera como un conjunto de píxeles conectados). Era necesario eliminar las regiones que no correspondían a la placa, para esto se tomó en cuenta el área de la placa, aquellas regiones cuya área no se encontraba en el rango de 3000 a 10,000 píxeles (el área de una región es la cantidad de píxeles conectados en una región, estos umbrales corresponden a píxeles 'prendidos' de la placa, en las dimensiones de las imágenes adquiridas). En el sistema de cómputo desarrollado, los dos pasos anteriores (eliminación de hoyos y partículas y la búsqueda de regiones que tengan una cierta área), se llevaron a cabo mediante rutinas establecidas en la biblioteca de procesamiento de imágenes de Matrox (Matrox Imaging Library), y en las cuales nos apoyamos en algunas etapas del procesamiento de imágenes.

Al término de este proceso aún había posibilidades de obtener una o más regiones y de alguna manera se debe verificar cuál de éstas corresponde precisamente a la lámina de la matrícula. Esto se llevó a cabo mediante la técnica de emparejamiento de plantillas. En general, el emparejamiento de plantillas consiste en barrer la imagen que se está analizando con un conjunto de plantillas predeterminadas, estas plantillas representan objetos que se buscan en la imagen, y determinar si un objeto en la imagen se asemeja al representado por la plantilla.

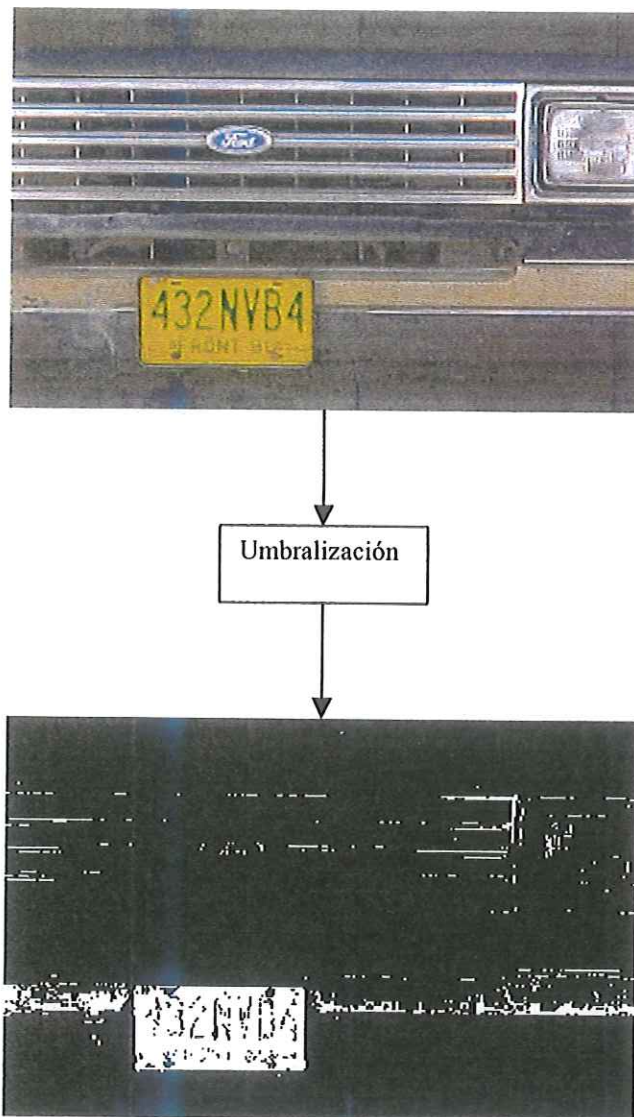


Fig. 12. Binarización mediante la umbralización de varias variables

En particular, en este caso se desea encontrar un rectángulo. Por lo tanto, solamente se necesitó una plantilla, la cual es un rectángulo de 85 píxeles de ancho por 43 de alto, las dimensiones aproximadas de una placa en las imágenes capturadas. Como ya se tenían las regiones donde probablemente se encontraba la placa solamente se analizaron éstas, no fue necesario barrer la imagen en su totalidad, en la figura 14 se muestra un ejemplo del resultado obtenido mediante esta técnica, los rectángulos punteados en la imagen resultante representan el recorrido que se hace con la plantilla para encontrar la zona rectangular. Una vez encontrada una región rectangular se calcularon las coordenadas de la posición inicial de ésta, y éstas coordenadas son utilizadas en la imagen original, para apartir de ahí realizar el “recorte” de la placa, como lo ilustra la figura 15.

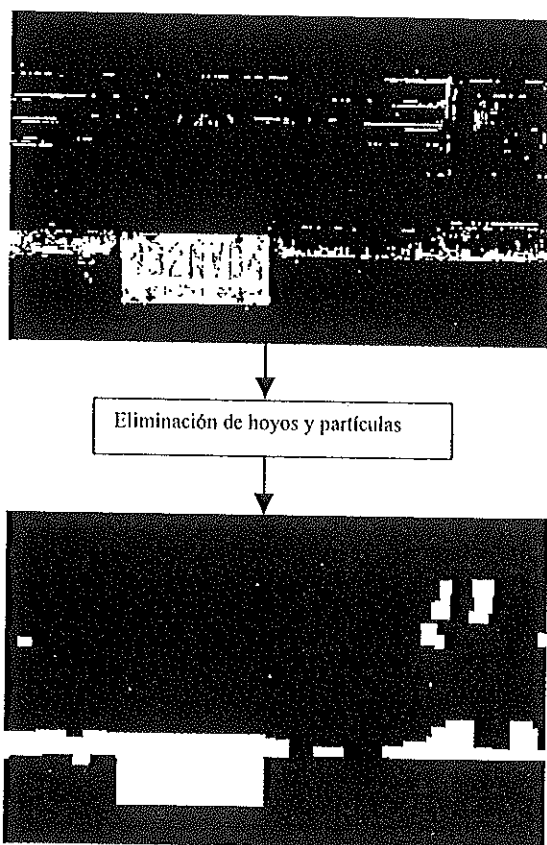


Fig. 13. Eliminación de hoyos y partículas.

El resultado de todo este proceso fue la obtención de la placa segmentada. El siguiente paso, que también es parte de la segmentación, consistió en extraer los caracteres impresos dentro de la subimagen con la placa.

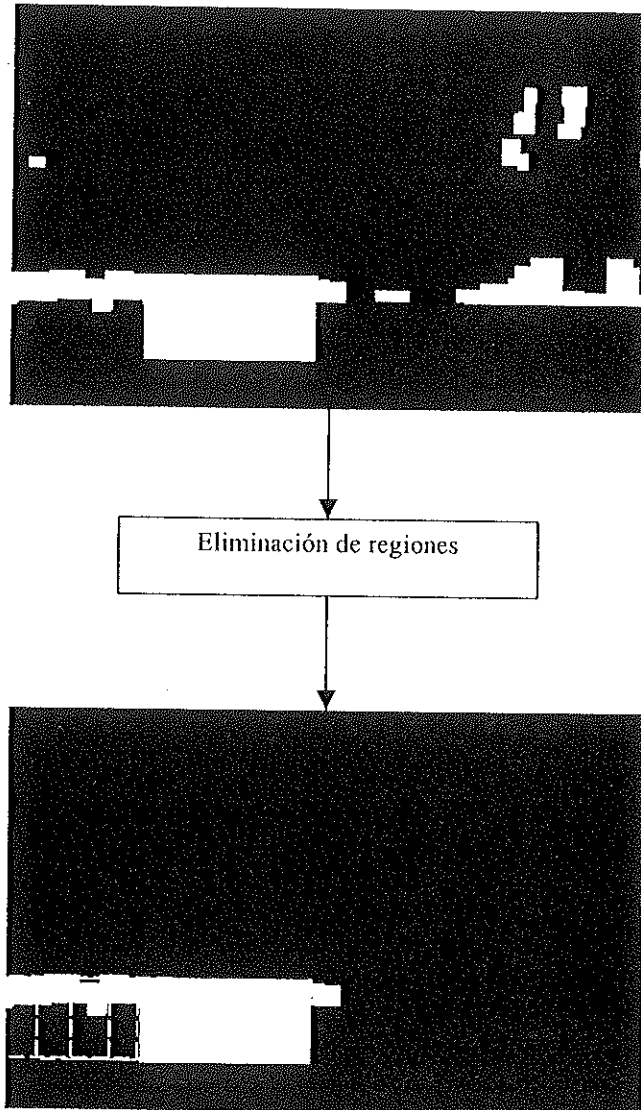


Fig. 14. Eliminación de regiones y enparejamiento de plantillas.

### 3.2.2. Resultados obtenidos en el primer paso de la segmentación

La técnica de segmentación que se acaba de describir llevó a cabo la segmentación de la placa de una forma exitosa en la mayoría de las imágenes muestra (aproximadamente el 92% de las imágenes). No se obtuvieron resultados en aquellas imágenes donde las regiones que se encontraban en ella no cumplían con la condición establecida, a pesar del hecho que los automóviles sí contaban con placa, pero estas tenían deformaciones (por ejemplo, estaban cortadas o dobladas), reduciendo de esta forma su área, causando así su eliminación. Fueron sólo 3 imágenes en las que surgió este problema. En la figura 16 se muestran algunas imágenes con las placas segmentadas.

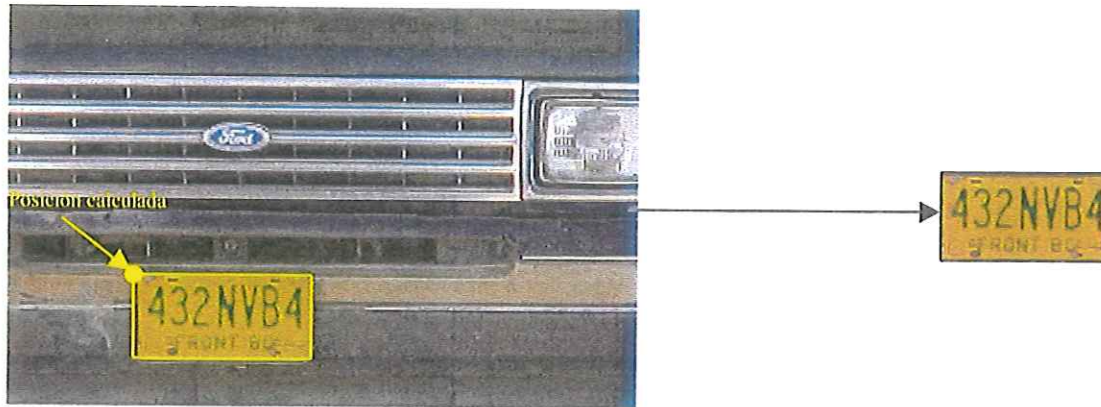


Fig. 15. Extracción de la placa.

### 3.2.3. Extracción de los caracteres contenidos en la placa

Una vez que se localiza y se obtiene la placa de la imagen del vehículo, el siguiente paso es extraer los caracteres a interpretar. Esta etapa se llevó a cabo de la siguiente forma:

Primeramente se binarizó la imagen de la placa con la matrícula a interpretar, utilizándose un procedimiento similar al aplicado en la segmentación de la placa, esto es, analizando los componentes RGB del color de los pixeles que conforman la imagen.

Debido a que los caracteres contenidos en las placas son de color verde, solamente se toman en cuenta pixeles que se consideren verdes, o cercanos al vértice en el cubo RGB correspondiente a este color, llevándose acabo así la umbralización. Nuevamente, los pixeles que satisfacen esta condición son tomados en cuenta, asignándoles el color blanco. A los pixeles restantes se les asigna el color negro, obteniéndose al final la imagen binaria. En la figura 17 se muestra la imagen de una placa y su imagen binaria resultante.



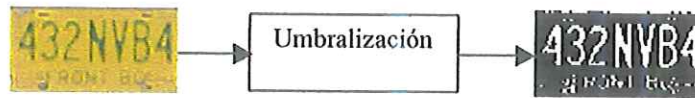


Fig. 17. Binarización de una placa

Para extraer los caracteres se utilizó nuevamente la técnica de emparejamiento de plantillas. Se analizó un conjunto de placas y se concluyó que las matrículas contenían ya fuera la letra Z o la letra N en ellas, se binarizaron una de cada placa que incluyen estas letras, se copiaron las letras mencionadas (Z y N) y se guardaron en un archivo para tenerse como modelos (plantillas). Para segmentar los caracteres de la placa ésta fue barrida utilizando las dos plantillas y se obtuvo un valor de correlación para cada una, realizando la suma de pixeles que tienen en común la plantilla y la región en la imagen contra la cual se compara. El valor de la suma indica en qué parte de la imagen se obtuvo una mayor correlación (qué parte se asemeja más a una de las plantillas), al final se comparan los valores y se toma el mayor, indicando que se encontró un caracter similar al modelo asociado con ese valor de correlación. Una vez realizado esto, se toman las coordenadas de la caja mínima que encierra al caracter encontrado, y a partir de ahí se calculan las coordenadas del primer caracter. Como no se sabe con certeza si la posición calculada de la caja encierra exactamente al caracter, fue necesario barrer con una plantilla adicional y solamente de forma horizontal la región calculada, con una holgura de 5 pixeles a la izquierda de la posición calculada y dos pixeles a la derecha, para detectar con una mayor precisión los caracteres. La plantilla utilizada engloba las formas de todos los caracteres a buscar (letras y dígitos). El modelo de la plantilla se basó en los número de un reloj digital, donde a partir de la figura de un número, el 8, se pueden formar los números restantes, borrando algunas de las líneas que lo componen. Así se mejoró la segmentación de los caracteres,

centrándolos más en la región donde se recortan, y a la vez incrementó la tasa de éxito en el proceso de clasificación. En la figura 18 se ilustra visualmente cómo se realiza el emparejamiento de plantillas para encontrar una región con forma de una N, el caracter marcado en la placa es contra el cual se compara la plantilla.

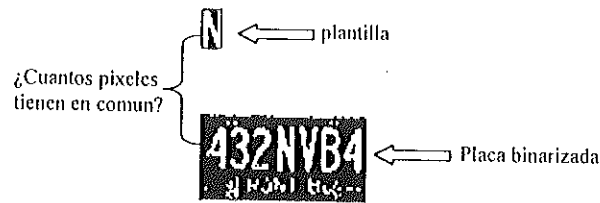


Fig. 18. Ilustración del emparejamiento de plantillas para encontrar una N en la placa

El siguiente paso en nuestro sistema (y en general, en cualquier otro sistema de reconocimiento de imágenes), fue la obtención de las características que describirán a cada uno de los caracteres que se desean interpretar.

### 3.2.4. Resultados obtenidos en la segmentación de caracteres

La segmentación de los caracteres se realizó exitosamente en aproximadamente el 84% de las imágenes obtenidas durante la segmentación de la placa. Se tuvieron fallas en aquellas placas donde a causa de la binarización se introducía ruido, se tenían píxeles blancos esparcidos en toda la placa (como lo muestra la figura 19), provocando distorsión a los caracteres, esto a su vez causó la localización de partes de la placa que tenían correlación con las plantillas utilizadas, sin embargo, estas partes no correspondían a caracteres.

La introducción de ruido durante la binarización fue a causa de un factor muy importante en los sistemas de reconocimiento de imágenes, *la iluminación*. Las imágenes fueron capturadas al aire libre, por lo tanto la fuente de iluminación fue el sol, y al momento de tomar

las imágenes donde se tuvieron problemas en la segmentación éste fue cubierto por nubes, disminuyendo la iluminación, o se encontraba de frente al vehículo, aumentando el brillo de la placa.

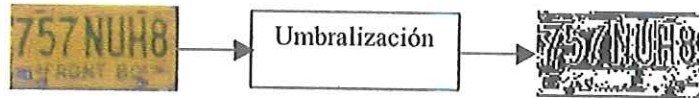


Fig. 19. Introducción de ruido durante la binarización de una placa con poca iluminación

## 4. Descriptores

Habiéndose realizado el proceso de segmentación de la imagen, el resultado es un conjunto de píxeles los cuales corresponden a las regiones de interés, es decir, a las regiones a las cuales se desea dar un significado, en nuestro caso de estudio, las que contienen un carácter. Este conjunto de píxeles se podría enviar inmediatamente al módulo de reconocimiento. Sin embargo, es recomendable utilizar algún sistema de descripción para evitar utilizar el total de píxeles de la región y de este modo aligerar la carga computacional del reconocedor, compactando la información obtenida durante la segmentación. La obtención de descriptores consiste en encontrar y obtener las características más relevantes de los objetos a reconocer, estas características deben satisfacer los siguientes requerimientos [Castleman, 1979]:

1. *Discriminación.*- Deben tomar valores diferentes para objetos que pertenecen a diferentes clases.
2. *Confiabilidad.*- Deben tomar valores similares para todos los objetos de una misma clase.
3. *Independencia.*- No deben estar correlacionadas una con otra. Por ejemplo, si en algún sistema se desea reconocer diferentes tipos de frutas, el diámetro y el peso de una fruta son características muy correlacionadas
4. *Números pequeños.*- La complejidad de un sistema de reconocimiento de patrones incrementa rápidamente con el número de características utilizadas. La cantidad de datos requeridos para entrenar el clasificador y medir su desempeño incrementa exponencialmente con el número de características.

Existen diferentes tipos de esquemas de representación, pero antes de seleccionar uno apropiado para la aplicación, se debe determinar cómo se representarán las regiones segmentadas. Básicamente, el representar una región involucra dos opciones [González y Woods, 1992]:

- Se puede representar la región en términos de sus características externas, es decir, su borde.
- Se puede representar mediante sus características internas, esto es, los píxeles que comprenden la región.

Una vez tomada la decisión de la representación, se procede a determinar los descriptores adecuados.

Debido a que en algunos sistemas de reconocimiento de imágenes, los objetos a reconocer de la imagen no siempre se encontrarán a una misma distancia (el sistema de adquisición se encontrará a distancias diferentes del objeto), o en la misma posición, se recomienda seleccionar descriptores invariantes a traslación, rotación y escala [Hall, 1979; González y Woods, 1992] En el caso de estudio, estas invariantes son mínimas, ya que las imágenes fueron tomadas a una misma distancia (lo que da como resultado caracteres de un mismo tamaño, o con una diferencia insignificante de tamaño de carácter a carácter), además que en la mayoría de vehículos la placa se encuentra colocada en una misma posición horizontal, a excepción de algunos, donde se encuentra un poco inclinada.

A continuación se presentan algunos esquemas de descripción encontrados en la literatura.

## **4.1. Algunos escritores del borde de una región**

Si se toma la decisión de representar las regiones mediante su borde, alguno de los siguientes descriptores puede ser utilizado:

### **4.1.1. Descriptores sencillos**

Uno de los descriptores más sencillos de un borde es su *longitud*. La longitud de un contorno se puede aproximar simplemente contando el número de píxeles a lo largo de éste.

Otro tipo de descriptor simple de un contorno es el valor de *su diámetro y la orientación de la línea* que conecta a los dos puntos extremos que comprenden el diámetro (esta línea es llamada el eje mayor del borde). El diámetro de un borde B se define como

$$\text{Diam}(B) = \max[ D(p_i, p_j) ]$$

donde D es una medida de distancia (por ejemplo, la distancia Euclidiana) y  $p_i$  y  $p_j$  son puntos en el borde.

La *curvatura* es otro descriptor que se puede utilizar para el borde de una región. La curvatura en términos matemáticos se define como la tasa de cambio de pendiente. En una imagen digital se puede utilizar como descriptor de curvatura la diferencia de pendientes de segmentos adyacentes del borde en el punto de intersección de estos segmentos.

### **4.1.2. Códigos cadena**

Los *códigos cadena* son utilizados para describir un borde mediante una secuencia conectada de segmentos de líneas rectas de una longitud y dirección específica [González y Woods, 1992]. La dirección de cada segmento se codifica utilizando un esquema de numeración, que depende del esquema de vecinos que se tome en cuenta (4 vecinos u 8

vecinos). En la figura 20 se presenta el esquema de numeración para cada número de vecinos.

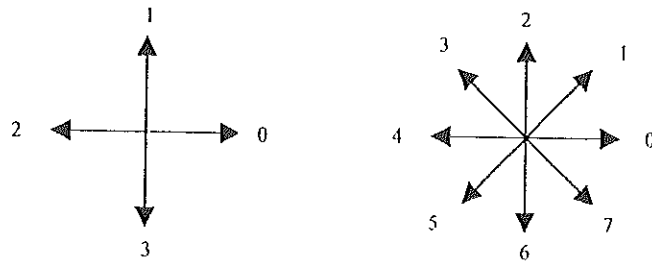


Fig 20. Esquema de numeración 4 y 8 vecinos, respectivamente.

Para obtener el código cadena de un borde se va recorriendo éste en el sentido de las manecillas del reloj y se van conectando mediante segmentos puntos adyacentes, asignando a cada segmento una dirección, una vez recorrido el borde en su totalidad, los segmentos se codifican dependiendo de su dirección, y del esquema de numeración seleccionado, empezando en un punto arbitrario y recorriéndose en sentido de las manecillas del reloj. En la figura 21 se muestra un ejemplo, ahí se parte del punto blanco, el resultado es una secuencia de números, el código .

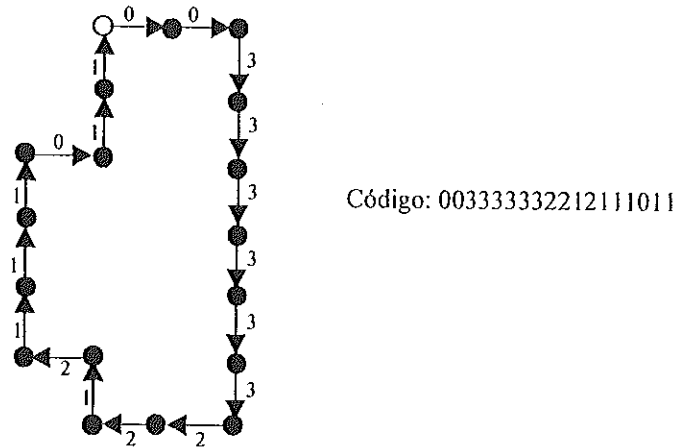


Fig 21. Codificación de un borde.

### 4.1.3. Descriptores de Fourier

La transformada de Fourier es una herramienta muy poderosa la cual se utiliza principalmente en el análisis de señales. Esta herramienta también puede ser utilizada para poder describir el borde de una región en una imagen, los descriptores obtenidos mediante el uso de la transformada de Fourier son llamados precisamente *descriptores de Fourier*.

El procedimiento para utilizar los descriptores de Fourier consiste en expresar las coordenadas de un contorno como números complejos, calcular la transformada de Fourier de estos números y tomar en cuenta solamente algunos componentes de la transformada como descriptores del contorno [González y Woods, 1992]. Esto es, si se tienen los puntos de un contorno, como se muestra en la figura 22, se puede obtener una secuencia de coordenadas  $(X_k, Y_k)$ , es decir una secuencia  $(X_0, Y_0), \dots, (X_{N-1}, Y_{N-1})$ , donde  $N$  es el número de puntos que conforman al contorno. Estas coordenadas se pueden expresar de la forma  $X(k) = X_k$  y  $Y(k) = Y_k$ . De esta forma, el contorno se puede representar como una secuencia de coordenadas  $S(k) = [X(k), Y(k)]$ . A su vez, cada par de coordenadas puede verse como un número complejo de la forma

$$S(k) = X(k) + iY(k)$$

para  $k = 0, 1, \dots, N-1$ . Viéndose al eje  $X$  como el eje real y al eje  $Y$  como el imaginario.

Se puede calcular la transformada de Fourier de  $S(k)$ , la cual es

$$a(u) = \sum S(k) \exp(-i2\pi uk/N)$$

Para  $u = 0, 1, \dots, N-1$ . Los coeficientes  $a(u)$  son llamados descriptores de Fourier. El objetivo es obtener estos descriptores y utilizar los  $M$  primeros para representar el borde.

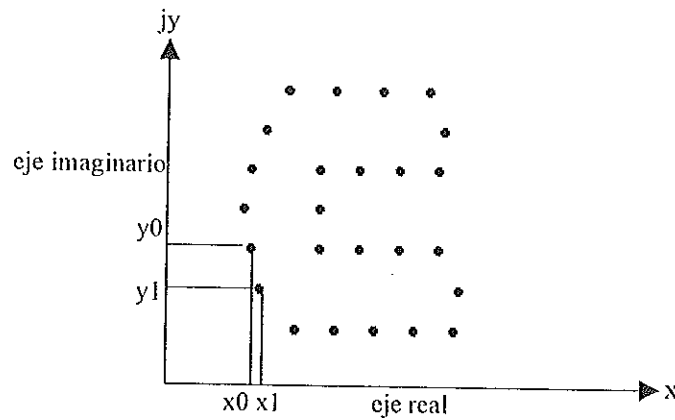


Fig 22. Representación de un borde como una secuencia de número complejos.

## 4.2. Descriptores de una región

Si las regiones se representarán utilizando todos los píxeles que las comprenden, alguno de los siguientes esquemas puede ser el apropiado para describirla.

### 4.2.1. Descriptores sencillos

Dos de los descriptores más simples de una región son su *área* y su *perímetro*. El área de una región se obtiene contando el número de píxeles dentro de (y tomando en cuenta) su borde, en una imagen binaria, el área corresponde al número de píxeles blancos en la región. El perímetro es la longitud de su borde. Estos descriptores son utilizados en conjunto para definir otro tipo de descriptor, la medida de *estrechez* (compactness) de la región, definida como

$$C = P^2/A$$

donde P es el perímetro y A el área.

La estrechez es un descriptor de los llamados medida de circularidad, ya que su valor es mínimo para figuras circulares [Castleman, 1979]. Su magnitud refleja la complejidad de la

región, para una región circular toma el valor  $4\pi$ . Figuras más complejas tienen un valor mayor.

Otro tipo de descriptor de región es la *densidad óptica integrada* (IOD, por sus siglas en inglés). La cual es la suma de los niveles de gris de los píxeles dentro de la región, y refleja la “masa” o “peso” del objeto. El IOD se obtiene mediante la fórmula

$$\text{IOD} = \sum k N_k$$

Donde  $k$  es el  $k$ -ésimo nivel de gris y  $N_k$  es el número de píxeles en la región con el  $k$ -ésimo nivel de gris. Analizando la fórmula del IOD se puede observar que se relaciona con el concepto de histograma de una imagen. El histograma de una imagen es una función que relaciona un nivel de gris en la imagen con la frecuencia en que ocurre en la misma, es decir, el número de píxeles que tienen ese nivel de gris, en base a esto, el término  $N_k$  es precisamente el histograma. De esta forma, la fórmula para obtener el IOD se puede reescribir como

$$\text{IOD} = \sum k H(k)$$

Donde  $H(k)$  es la función histograma.

#### **4.2.2. Descriptores topológicos**

Otro tipo de descriptor son los descriptores topológicos. Las propiedades topológicas son útiles para descripciones globales de regiones en una imagen. La topología es el estudio de propiedades de una figura que no sea afectada por ninguna deformación, siempre y cuando la figura no se rompa o se doble. Si un descriptor topológico está definido por el *número de hoyos* en una región, esta propiedad obviamente no será afectada por variaciones en el tamaño de la región, o rotaciones de la misma. En la figura 23 se muestra una región con dos hoyos.

Otro descriptor topológico útil es el número de *componentes conectados*. Un componente está conectado si dos píxeles cualquiera  $p$  y  $q$  que forman parte del componente

están conectados. Si  $p$  tiene coordenadas  $(x, y)$  y  $q$  tiene coordenadas  $(s, t)$ , están conectados si existe un camino de píxeles, de coordenadas  $(x_0, y_0), \dots, (x_1, y_1)$  que unan  $p$  con  $q$ , donde  $(x, y) = (x_0, y_0)$  y  $(s, t) = (x_n, y_n)$ , si se cumple esta condición para todos los píxeles del componente, entonces está conectado [González y Woods, 1992]. En la figura 24 se muestran dos componentes conectados.

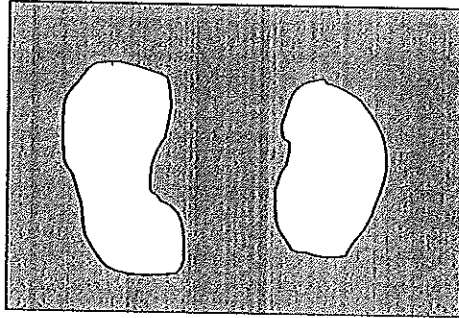


Fig 23. Región con dos hoyos

El número de hoyos y el número de componentes conectados en una figura pueden ser utilizados para definir otro descriptor, el *número de Euler*, definido como

$$E = C - H$$

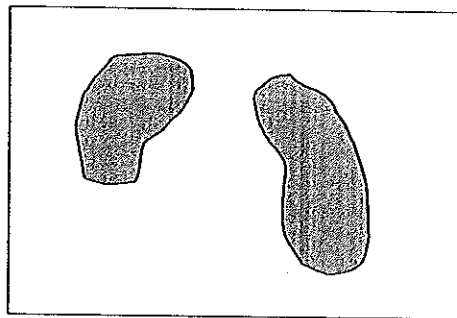


Fig 24 Dos componentes conectados

Donde C es el número de componentes conectados y H el número de hoyos. En la figura 25 se muestran regiones con número de Euler igual a 0 y -1.

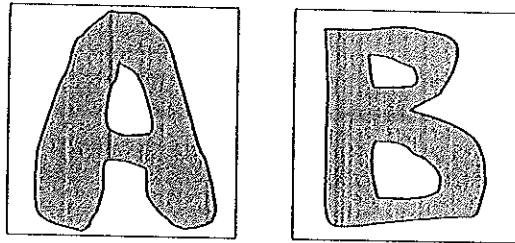


Fig 25. Regiones con número de Euler igual a 0 y -1, respectivamente.

### 4.2.3. Momentos

La teoría de momentos es utilizada principalmente en probabilidad. Los momentos de una función tiene la cualidad de ser únicos para esa función [Papoulis, 1965], es decir, el conjunto de momentos que genera una función son suficientes para especificar esa función. Por tanto, los momentos también pueden ser utilizados para describir un objeto en una imagen. El conjunto de momentos de una función bidimensional  $f(x, y)$  está definido por

$$M_{pq} = \iint x^p y^q f(x, y) dx$$

para  $p, q = 0, 1, 2, \dots, N$ . Y  $p + q$  es el orden del momento.

Para la descripción de regiones supongamos que  $f(x, y)$  toma el valor de 1 dentro del objeto y 0 en cualquier otra parte, mostrando la silueta del objeto.

Existe solamente un momento de orden 0

$$M_{00} = \iint f(x, y) dx$$

El cual es el área del objeto. Los momentos de primer orden y de orden superior se pueden hacer invariantes al escalamiento dividiéndolos entre el momento de orden 0 [Castleman, 1979].

Las coordenadas del centro de gravedad del objeto están dadas por

$$\bar{x} = M_{10} / M_{00} \quad \bar{y} = M_{01} / M_{00}$$

Los momentos centrales son calculados utilizando el centro de gravedad como el origen

$$\mu_{pq} = \iint (x - \bar{x}) (y - \bar{y}) f(x, y) dx$$

Para una imagen digital la ecuación de momentos centrales se expresa como

$$\mu_{pq} = \sum \sum (x - \bar{x}) (y - \bar{y}) f(x, y)$$

Los momentos centrales hasta el tercer orden son:

$$\mu_{10} = \sum \sum (x - \bar{x}) (y - \bar{y}) f(x, y)$$

$$= M_{10} - (M_{10}/M_{00})(M_{00}) = 0$$

$$\mu_{11} = \sum \sum (x - \bar{x}) (\bar{y} - y) f(x, y)$$

$$= M_{11} - (M_{10}M_{01} / M_{00})$$

$$\mu_{20} = \sum \sum (x - \bar{x}) (\bar{y} - y) f(x, y)$$

$$= M_{20} - (2M_{210} / M_{00}) + (M_{10} / M_{00}) = M_{20} - (M_{210} / M_{00})$$

$$\mu_{02} = \sum \sum (x - \bar{x}) (\bar{y} - y) f(x, y)$$

$$= M_{02} - (M_{201} / M_{00})$$

$$\mu_{30} = \sum \sum (x - \bar{x}) (\bar{y} - y) f(x, y)$$

$$= M_{30} - 3\bar{x} M_{20} + 2\bar{x}^2 M_{10}$$

$$\mu_{12} = \sum \sum (x - \bar{x}) (y - \bar{y}) f(x, y)$$

$$= M_{12} - 2\bar{y} M_{11} + \bar{x} M_{02} + 2\bar{y}^2 M_{10}$$

$$\mu_{21} = \sum \sum (x - \bar{x}) (y - \bar{y}) f(x, y)$$

$$= M_{21} - 2\bar{x} M_{11} + \bar{y} M_{02} + 2\bar{x}^2 M_{10}$$

$$\begin{aligned}\mu_{03} &= \sum \sum (x - \bar{x})^0 (y - \bar{y})^3 f(x, y) \\ &= M_{03} - 3\bar{y} M_{02} + 2\bar{y}^2 M_{01}\end{aligned}$$

Se puede obtener un conjunto de momentos centrales normalizados, mediante la siguiente fórmula

$$\eta_{pq} = \mu_{pq} / \mu_{00}^\gamma$$

donde  $\gamma = (p+q / 2) + 1$ , para  $p+q = 2, 3, \dots$

A partir de los momentos centrales normalizados de segundo y tercer orden se puede obtener un conjunto de siete momentos invariantes [González y Woods, 1992]

$$\Phi_1 = \eta_{20} + \eta_{02}$$

$$\Phi_2 = (\eta_{20} - \eta_{02})^2 + 4\eta_{11}$$

$$\Phi_3 = (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - \eta_{03})^2$$

$$\Phi_4 = (\eta_{30} + \eta_{12})^2 + (\eta_{21} + \eta_{03})^2$$

$$\begin{aligned}\Phi_5 &= (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12}) [(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] \\ &\quad + (3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03}) [3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2]\end{aligned}$$

$$\begin{aligned}\Phi_6 &= (\eta_{20} - \eta_{02}) [(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] \\ &\quad + 4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{21} + \eta_{03})\end{aligned}$$

$$\begin{aligned}\Phi_7 &= 3(\eta_{21} - \eta_{03})(\eta_{30} + \eta_{12}) [(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] \\ &\quad + (3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03}) [3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2]\end{aligned}$$

### **4.3. Descriptores utilizados en el caso de estudio**

En el caso de estudio se utilizaron dos vectores de descriptores, que sirven como entrada al módulo de reconocimiento con el uso de redes neuronales, los vectores son utilizados por separado, esto para aumentar la tasa de éxito en el reconocimiento. El primer vector contiene información calculada utilizando un esquema de descripción de la región del carácter. Los componentes del vector corresponden a descripciones locales de vecindades de  $3 \times 3$  píxeles en diferentes posiciones del carácter. El segundo vector contiene información de los hoyos que pueden presentarse en un carácter (número de hoyos, su área y su posición).

#### **4.3.1. Esquema de descripción de la región del carácter**

El esquema principal de descripción utilizado en nuestra aplicación consistió en el uso de una malla (como descriptor de región del carácter). La dimensión de los caracteres segmentados es de  $9 \times 21$ , y sobre cada imagen se colocó una malla de  $3 \times 7$ , así cada celda de la malla cubre una vecindad de  $3 \times 3$  como se muestra en la figura 26. De cada celda de la malla se obtiene un promedio de los píxeles blancos que se encuentran en dicha celda. Es decir, se cuentan los píxeles blancos y se dividen entre el número de píxeles de la vecindad encerrada por cada celda, obteniéndose un vector de descriptores con 21 componentes. Este esquema es bastante sencillo y además tiene un poder de generalización considerable. Una debilidad es su sensibilidad a la rotación y escalamiento, pero como se había dicho, estas invariantes son mínimas en nuestra aplicación. Una de sus principales ventajas es la rapidez con que se obtiene, ya que los cálculos involucrados son sumas y divisiones.

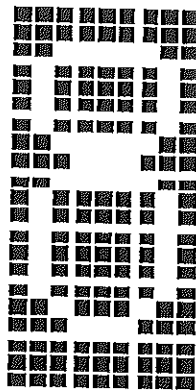


Fig 26 Caracter con malla sobrepuesta.

Se realizaron pruebas con el conjunto de dígitos obtenido durante la segmentación, utilizando un módulo de reconocimiento preliminar. En este módulo se implementa un multiperceptrón de una sola capa, con una arquitectura de 21 unidades de entrada (las correspondientes a cada componente en el vector de descriptores), y 10 salidas (una para cada dígito), con lo que se obtuvo un porcentaje de éxito del 91% (con un total de 142 dígitos segmentados), un resultado bastante aceptable.

#### 4.3.2. Esquema de descripción secundario

Debido a que existen probabilidades de que la red neuronal cometa un error, o tenga indecisión en algún caso donde se encuentre confundido con un caracter, por ejemplo al presentarse un 3 y la red neuronal no esté totalmente convencida de si se trata precisamente de un tres o algún otro dígito parecido a éste, como un 8, es necesario utilizar otro tipo de descriptor adicional (y por lo tanto una nueva red neuronal que implemente este descriptor en el reconocimiento del caracter), y reducir con ello la probabilidad de una mala clasificación. Los descriptores secundarios que se utilizaron son el número de hoyos de los caracteres, su posición

vertical (arriba, abajo, o enmedio) y el área de estos hoyos. El número de hoyos ayuda a distinguir entre caracteres que son muy similares y donde la primer red puede confundirse, como en el caso del 3 y el 8. La posición vertical de los hoyos es utilizada para poder distinguir entre caracteres que tengan un sólo hoyo, pero en posiciones distintas, como los dígitos 0, 6, y 9. Se puede observar que el hoyo del 0 se encuentra enmedio de él, el hoyo del seis en la parte inferior y el del 9 en la parte superior.

Para llevar a cabo la distinción entre las posiciones de los hoyos, se tomaron en cuenta todas las imágenes con dígitos que contenían un hoyo obtenidas durante la segmentación, se obtuvo el hoyo y se analizó su posición, y se determinaron los rangos que definen a cada una de éstas. Por último, el interés en utilizar el área, es debido a que en caracteres como el 0 y 4 el hoyo se encuentra en el rango de posiciones que definen a la parte media, y de alguna forma se debía hacer la distinción para evitar ambigüedades, y el área resultó ser el descriptor ideal para esta tarea. En el caso de caracteres con dos hoyos, como el 8, se tomaron en cuenta la distancia que existe entre los hoyos, así como el área de cada uno de ellos.

#### **4.3.1.1. Obtención de la información de los hoyos**

La biblioteca de procesamiento de imágenes utilizada en el sistema de cómputo cuenta con rutinas para determinar el número de hoyos de una región en una imagen, sin embargo éstas fallaban en casos cuando por alguna causa, debido a ruido o algún otro factor, durante la binarización los pixeles que se encontraban en el borde de un hoyo se desconectaban, llevándose a cabo la operación erróneamente, para evitar este tipo de fallas se trabajó directamente con las imágenes.

Para obtener los hoyos, primeramente se obtuvo la imagen binaria inversa de los dígitos, es decir, el fondo de los caracteres contenía pixeles blancos, y el caracter estaba definido por pixeles negros, por lo tanto, la región de los hoyos está conformada por pixeles blancos. Después de binarizarse, se buscó el contorno del caracter (los pixeles negros), renglón por renglón en la matriz de la imagen, y al encontrarse un pixel negro, se eliminaban los pixeles blancos que se encontraban en ese mismo renglón, en el caso donde al recorrerse todo el renglón no se encontraban pixeles negros, también se eliminaban todos los pixeles de ese renglón, ya que pertenecían al fondo. Este proceso se llevó a cabo recorriendo la imagen de izquierda a derecha hasta encontrar el lado izquierdo del hoyo, y de derecha a izquierda, hasta encontrar el lado derecho del hoyo, obteniéndose al final el hoyo (u hoyos) del caracter. En la figura 27 se muestra gráficamente un ejemplo de cómo se localiza el hoyo de un 4 en este caso, en la figura de la izquierda se muestra el dígito, las flechas indican hacia donde se lleva a cabo el recorrido. En la figura de la derecha se muestra un resultado parcial del proceso, donde se han eliminado los pixeles blancos de los primeros cuatro renglones, obsérvese cómo se va aislando la región del hoyo.

Ya teniendo los hoyos, el total de éstos y su área se obtuvo mediante rutinas de la biblioteca mencionada. Esta información se guardó en un vector para enviarse a la red neuronal dedicada al reconocimiento mediante estos descriptores.

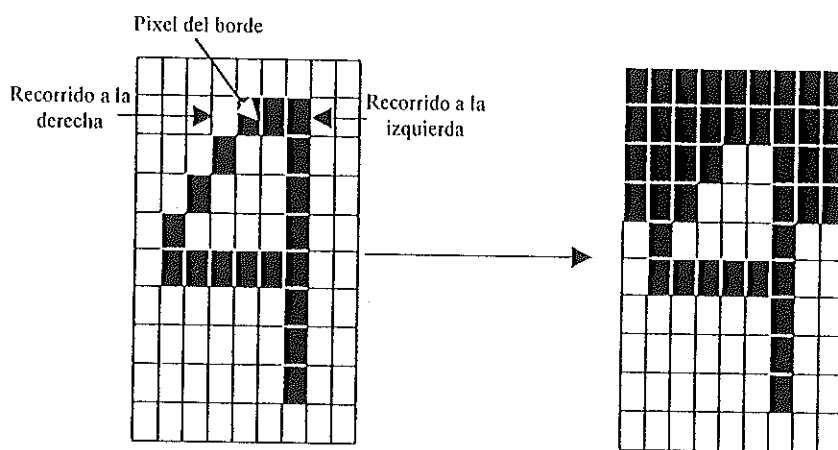


Fig 27. Obtención del hoyo de un dígito

Previo a la selección de los esquemas descritos se hicieron pruebas con otros esquemas de descripción, como el conjunto de momentos invariantes mencionados anteriormente. Se descartó este esquema ya que mediante pruebas realizadas se concluyó que este tipo de descriptores no es apropiado para esta aplicación en particular, puesto que los momentos de cada caracter toman todos valores similares, y no cumple con la característica de discriminación entre objetos de diferentes clases, a pesar de que los momentos sean, de acuerdo con [Gonzalez y Woods, 1992], invariantes a rotación, traslación y escalamiento. Tal vez sean mas apropiados para aplicaciones donde se utilicen imágenes con niveles de gris, y no binarias.

## **5. Reconocimiento**

Terminadas las etapas anteriores, se ha llegado a la última etapa del análisis de imágenes, que es el reconocimiento e interpretación de los datos obtenidos durante la segmentación ya descritos anteriormente. La tarea del reconocimiento es clasificar un patrón que se presente, es decir, se debe asignar a una de varias clases disponibles, estas clases corresponden a los objetos que el módulo de reconocimiento debe identificar. En esta sección se dará una breve explicación de los algoritmos clásicos utilizados en el reconocimiento de patrones, así como la técnica utilizada para la clasificación en el presente proyecto. Antes de entrar de lleno a las técnicas de reconocimiento, es de importancia definir algunos conceptos fundamentales.

### **5.1. Objetos, patrones y clases**

En la literatura de procesamiento de imágenes, algunos autores se refieren a la etapa de clasificación como reconocimiento de imágenes, otros como reconocimiento de patrones visuales, o como clasificación de objetos. En esencia, el objetivo de esta etapa es proporcionar a una máquina la capacidad para dar un significado a los objetos que se encuentran en una imagen, una capacidad que se asemeje lo mejor posible a aquella de los humanos [Gonzalez y Woods, 1992]. En el contexto de análisis de imágenes, un objeto, y un patrón toman significados similares pero a la vez con una marcada diferencia. Un objeto se refiere al objeto físico (valga la redundancia) que se encuentra en la imagen que se analiza, como un auto, una fruta, una letra, etc. Como de los datos en una imagen no se puede dar directamente significado a un objeto, se deben obtener las características esenciales que los describan, tarea que se realiza durante la etapa de descripción dando como resultado un vector de características. Un

patrón es una descripción cuantitativa o estructural de un objeto o alguna otra entidad de interés en una imagen [Gonzalez y Woods, 1992], y está formado por uno o más descriptores, en otras palabras, el vector obtenido durante la descripción es un patrón el cual describe al objeto en la imagen, y que es utilizado por el reconocedor para asignarlo a una clase. Por último, una clase es un conjunto de patrones que comparten algunas características en común, y se denotarán como  $\omega_i$ .

## 5.2. Espacio de características

Otro concepto importante es el espacio de características. El espacio de características es un espacio  $n$ -dimensional donde cada eje corresponde a cada una de las características tomadas en cuenta para describir los diferentes tipos de objetos a reconocer, y un punto en el espacio corresponde al patrón que representa a un objeto. En este espacio se encuentran agrupaciones de puntos que tiene características similares, y cada una de estas agrupaciones corresponden a las clases de objetos. Por ejemplo en el espacio de características de caracteres se tienen grupos de puntos que definen a la letra A, otra agrupación para la B, etc. En ocasiones se dá el caso donde las clases no son distinguibles, es decir, se mezclan los puntos pertenecientes a dos (o más), dificultando su distinción, esto se debe a que las características seleccionadas no cumplen con la condición de discriminación, de esta forma se enfatiza la importancia de la selección del esquema de descripción adecuado para dicha tarea. Sabemos que en realidad no existe un esquema de descripción óptimo [Jahne, 1997], no hay un esquema mediante el cual se obtenga una separación perfecta de las diferentes clases en el espacio de características.

### 5.3. Funciones de decisión

La parte importante en la etapa de reconocimiento es encontrar un conjunto de funciones, llamadas *funciones de decisión o de discriminación*, que son utilizadas para determinar a qué clase pertenece un patrón que se presente. Estas funciones deben tener la cualidad de cometer la mínima cantidad de errores posible al momento de la clasificación. Cada clase cuenta con su función de decisión  $d_i(x)$ , donde  $i$  denota la clase, y al presentarse un patrón  $x$  a reconocer, la decisión se toma comparando los valores obtenidos al aplicar cada función de decisión al patrón, y se dice que pertenece a la  $i$ -ésima clase  $\omega_i$  si se cumple la condición

$$d_i(x) > d_j(x), \text{ para toda } i \neq j$$

esto es, si se obtiene el mayor valor numérico con la función de la  $i$ -ésima clase, entonces el patrón pertenece a ésta.

Todas las clases en el espacio de características se encuentran separadas por una frontera, llamada *frontera de decisión*. La frontera de decisión que separa a las clases  $\omega_i$  y  $\omega_j$  está dada por los valores de  $x$  para los cuales

$$d_i(x) = d_j(x)$$

o equivalentemente, los valores de  $x$  para los que

$$d_i(x) - d_j(x) = 0.$$

Es común el utilizar una sola función para identificar la frontera de decisión

$$d_{ij}(x) = d_i(x) - d_j(x) = 0.$$

Entonces, si  $d_{ij}(x) > 0$ , el patrón pertenece a la clase  $\omega_i$ , si  $d_{ij}(x) < 0$ , pertenece a la clase  $\omega_j$ .

A continuación se presentan las funciones de decisión más utilizadas para la clasificación.

## 5.4. Clasificador de distancia mínima

En el espacio de características, una clase puede ser representada por el vector más significativo, este vector, llamado *vector prototipo*, es el vector promedio de los vectores contenidos en dicha clase, y es calculado mediante la fórmula de la media matemática,

$$m_i = 1/N_i \sum_{x \in \omega_i} x \quad ; i = 1, 2, \dots, M$$

donde  $N_i$  es el número de vectores de la clase  $\omega_i$ , y la sumatoria se realiza sobre estos vectores,  $M$  es el número total de clases. De esta forma, una manera de clasificar un patrón desconocido es calculando la distancia euclidiana del punto que define al patrón desconocido hacia cada uno de los puntos de los vectores prototipo de cada clase, y asignándose a aquella clase cuya distancia sea la menor. En este caso las funciones de decisión están dadas por la fórmula

$$D_i(x) = \|x - m_i\|$$

donde  $\|a\| = (a^T a)^{1/2}$  es la norma euclidiana.

En base a lo anterior, la condición de membresía está dada por

$$D_i(x) < D_j(x)$$

y que difiere de la condición dada anteriormente. Sin embargo, se pueden hacer algunos arreglos a las funciones para que la condición concuerde con la definición de función de decisión.

Se tiene que para que un patrón desconocido pertenezca a la clase  $\omega_i$ , se debe cumplir

$$\|x - m_i\| < \|x - m_j\|$$

aplicando la definición de norma euclidiana tenemos

$$[(x - m_i)^T (x - m_i)]^{1/2} < [(x - m_j)^T (x - m_j)]^{1/2}$$

desarrollando y elevando al cuadrado

$$x^T x - 2x^T m_i + m_i^T m_i < x^T x - 2x^T m_j + m_j^T m_j$$

el término  $x^T x$  puede ser eliminado, sin afectar la desigualdad, y dividiendo entre  $-2$  se obtiene

$$x^T m_i - \frac{1}{2} m_i^T m_i > x^T m_j - \frac{1}{2} m_j^T m_j$$

esta desigualdad se puede expresar como

$$d_i(x) > d_j(x)$$

La frontera de decisión entre las clases  $\omega_i$  y  $\omega_j$  para un clasificador de distancia mínima está dada por

$$d_{ij}(x) = d_i(x) - d_j(x) = x^T(m_i - m_j) - \frac{1}{2}(m_i - m_j)^T(m_i - m_j) = 0$$

La superficie de decisión dada por esta función es el bisector perpendicular del segmento de línea que une a  $m_i$  y  $m_j$ . Para el caso bidimensional, el bisector perpendicular es una línea, cuando se utilizan tres características, se trata de un plano, para más de tres características es un hiperplano.

En la práctica este clasificador muestra un buen desempeño cuando la distancia entre vectores prototipo es grande comparado al esparcimiento o aleatoriedad de cada clase con respecto a su media [Gonzalez y Woods, 1992].

## 5.5. Emparejamiento mediante correlación

Existe otro tipo de función de decisión llamada función de correlación. La correlación de dos funciones  $f(x)$  y  $g(x)$  es una propiedad de la transformada de Fourier, y está definida por

$$f(x) \circ g(x) = \int f^*(\alpha) g(x+\alpha) d\alpha$$

donde  $\alpha$  es una variable temporal de integración y el asterisco (\*) denota el conjugado complejo (recordando que la transformada de Fourier se implementa utilizando números complejos). Esta integral representa el área de las regiones donde se intersecta  $f(x)$  y  $g(x)$ , al desplazarse la segunda función sobre la primera, en el rango donde se encuentra la variable  $x$ .

En la clasificación de patrones mediante el uso de plantillas se utiliza un método análogo para encontrar en una imagen (la imagen en proceso de análisis) la región que sea más parecida a otra imagen (la plantilla). Denotando a la imagen y a la plantilla como funciones  $f(x, y)$  y  $w(x, y)$ , y de dimensiones  $M \times N$  y  $J \times K$ , respectivamente, el valor de correlación se calcula por medio de

$$c(s, t) = \sum_x \sum_y f(x, y) w(x-s, y-t)$$

donde  $s = 0, 1, \dots, M$  y  $t = 0, 1, \dots, N$  y la sumatoria se realiza en el área donde  $f(x, y)$  y  $w(x, y)$  se traslapan. En otras palabras, la plantilla se recorre sobre la imagen y el valor de correlación se obtiene multiplicando los valores de los píxeles del área cubierta por la plantilla por los valores de los píxeles correspondientes a ésta. Se deben definir los puntos de inicio tanto de la plantilla como de la imagen, los más comunes son la esquina superior izquierda o la posición central, para determinar el punto sobre el cual se obtiene la correlación. En la figura 28 se muestra gráficamente cómo se lleva a cabo este procedimiento.

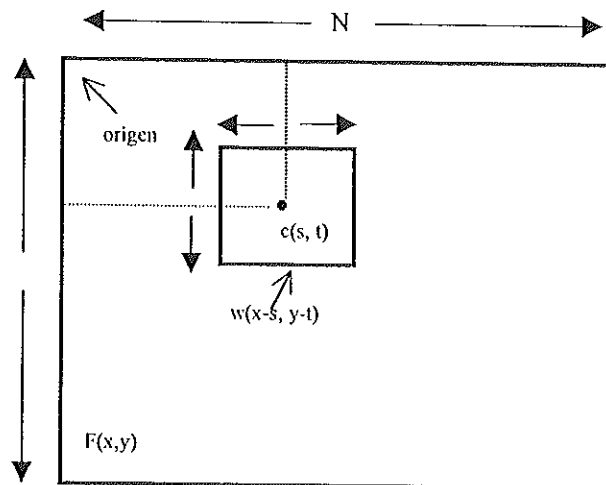


Fig 28. Obtención de la correlación de  $f(x, y)$  y  $w(x, y)$  en el punto  $(s, t)$ .

Este método fue utilizado en el presente trabajo, mas no como reconocedor, sino como complemento en la etapa de segmentación, para encontrar en la placa una región similar a una letra N o Z.

La función de correlación anterior tiene algunas desventajas. Por ejemplo, si se duplican los valores de  $f(x, y)$ , también se duplican los valores de  $c(s, t)$ . Existe otra función donde no se tiene este tipo de problema, llamada *coeficiente de correlación*, y dada por

$$\frac{\sum \sum [f(x, y) - \bar{f}(x, y)][w(x-s, y-t) - \bar{w}]}{[\sum \sum [f(x, y) - \bar{f}(x, y)][w(x-s, y-t) - \bar{w}]}$$

donde  $\bar{w}$  es el promedio de valores de los pixeles en la plantilla, y  $\bar{f}(x, y)$  es el promedio de valores de pixeles en el área de la imagen cubierta por la plantilla. Se puede observar que el coeficiente de correlación tiene mayor complejidad matemática que la correlación simple, ya que involucra las cuatro operaciones matemáticas básicas, y además se debe calcular el promedio de pixeles de cada región sobre la cual se coloca la plantilla.

## 5.6. Clasificador de Bayes

El orden en el que los patrones a clasificar se presentan al reconocedor es incierto, por lo tanto en este proceso está involucrada la aleatoriedad. Tomando en cuenta esto, se puede utilizar la teoría de probabilidad para obtener un clasificador basado en ella. Este tipo de clasificador utiliza principalmente la probabilidad condicional, recordando que de ella se obtiene la probabilidad de que ocurra un evento A, a partir de la ocurrencia de otro evento B, y se denota por  $p(A/B)$ . En este caso se calcula la probabilidad de que dado un patrón en

particular  $x$ , el patrón a reconocer, la clase a la que pertenezca sea la clase  $\omega_i$ , esto es,  $p(\omega_i/x)$ , así como la probabilidad de ocurrencia de cada clase,  $P(\omega_i)$ . Este clasificador es óptimo en el sentido de que, en promedio, su utilización dá como resultado la probabilidad más baja de cometer errores de clasificación [Gonzalez y Woods, 1992].

Las funciones de decisión utilizadas por este clasificador tienen la forma

$$d_i(x) = p(x/\omega_i)P(\omega_i).$$

donde  $p(x/\omega_i)$  es la función de densidad de probabilidad de la  $i$ -ésima clase, y  $P(\omega_i)$  es su probabilidad de ocurrencia.

Estas funciones minimizan la probabilidad de una mala clasificación, sin embargo, para esto es necesario conocer de antemano los factores que la componen. En el caso de la probabilidad de ocurrencia, son sencillas de obtenerse, o suponerse. Por ejemplo, si se supone que todas las clases son equiprobables, todas las clases tendrían como probabilidad de ocurrencia  $1/M$ . También se pueden obtener realizando estadísticas sobre una muestra significativa de los patrones de cada clase.

La función de densidad de probabilidad más utilizada por este clasificador es la función Gaussiana [Gonzalez y Woods, 1992]. En el caso unidimensional, donde cada clase tiene como media  $m_j$  y desviación estándar  $\sigma_j$ , las funciones de decisión tienen la forma

$$d_j(x) = \frac{1}{\sqrt{2\pi} \sigma_j} \exp\left[-\frac{(x - m_j)^2}{2 \sigma_j^2}\right] P(\omega_i)$$

Como un ejemplo, en el caso donde  $M = 2$ , la frontera de decisión entre las dos clases es un punto  $x_0$  donde  $d_1(x) = d_2(x)$ . Suponiendo que las dos clases son igualmente probables de ocurrir,  $P(\omega_1) = P(\omega_2) = 1/2$ , la frontera de decisión está definida por el punto  $x_0$ , tal que  $p(x/\omega_1)$

=  $p(x/\omega_2)$ , este punto es el punto donde se intersectan las funciones de densidad de cada clase.

En la figura 29 se muestra la gráfica de cada función y la frontera de decisión.

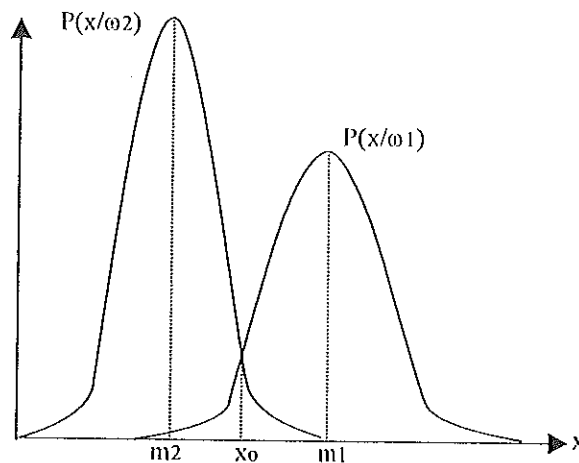


Fig 29. Funciones de densidad de probabilidad para dos clases de patrones unidimensionales. El punto  $x_0$  es la frontera de decisión

Claramente se puede observar que la frontera de decisión de este caso juega un papel de umbral, similar al caso de la técnica de segmentación por umbralización. El clasificador va a asignar un patrón  $x$  como proveniente de la clase  $\omega_1$  si tiene un valor mayor que  $x_0$ , si tiene un valor menor, lo clasifica como perteneciente a la clase  $\omega_2$ . Si las dos clases no son equiprobables, el punto de decisión se mueve a la izquierda si la clase  $\omega_1$  es más probable de ocurrir, o a la derecha si lo fuera la clase  $\omega_2$ .

En el caso general  $n$ -dimensional, la densidad Gaussiana de los vectores en la  $j$ -ésima clase tiene la forma

$$d_j(\mathbf{x}) = \frac{1}{(2\pi)^{n/2} |C_j|^{1/2}} \exp\left[-\frac{1}{2}(\mathbf{x} - \mathbf{m}_j)^T C_j^{-1} (\mathbf{x} - \mathbf{m}_j)\right]$$

Cada densidad está definida por su vector medio  $\mathbf{m}_j$  y su matriz de covarianza  $C_j$ , definidos como

$$\mathbf{m}_j = E_j\{\mathbf{x}\}$$

$$\mathbf{C}_j = E_j\{(\mathbf{x} - \mathbf{m}_j)(\mathbf{x} - \mathbf{m}_j)^T\}$$

donde  $E\{\cdot\}$  es la esperanza matemática del argumento sobre los patrones de la  $j$ -ésima clase.  $|\mathbf{C}_j|$  es el determinante de la matriz de covarianza.

El vector medio y la matriz de covarianza se pueden aproximar promediando las cantidades en cuestión

$$\mathbf{m}_j = 1/N_j \sum_{\mathbf{x} \in \omega_j} \mathbf{x}$$

$$\mathbf{C}_j = 1/N_j \sum_{\mathbf{x} \in \omega_j} \mathbf{x} \mathbf{x}^T - \mathbf{m}_j \mathbf{m}_j^T$$

$N_j$  es la cantidad de vectores en la clase  $\omega_j$ , y la suma se realiza sobre estos vectores.

La matriz de covarianza es simétrica y positiva semidefinida. Los elementos de la diagonal principal,  $c_{kk}$ , corresponden a la varianza del  $k$ -ésimo elemento de los vectores patrón. Los elementos fuera de la diagonal principal,  $c_{jk}$ , son la covarianza de  $x_j$  y  $x_k$ . Cuando  $x_j$  y  $x_k$  son estadísticamente independientes,  $c_{jk}$  tiene valor de cero. En este caso la función de densidad Gaussiana multivariable se reduce al producto de la densidad Gaussiana de una variable de cada elemento del vector  $\mathbf{x}$ .

Debido a la forma exponencial de la función de densidad Gaussiana, es más conveniente trabajar con su logaritmo natural. Las funciones de decisión toman la forma

$$d_j(\mathbf{x}) = \ln[p(\mathbf{x}/\omega_j)P(\omega_j)] = \ln[p(\mathbf{x}/\omega_j)] + \ln[P(\omega_j)]$$

El orden numérico de esta función y la original es el mismo. Sustituyendo la función de densidad Gaussiana se tiene

$$d_j(\mathbf{x}) = \ln[P(\omega_j)] - n/2 \ln(2\pi) - 1/2 \ln[|\mathbf{C}_j|] - 1/2(\mathbf{x} - \mathbf{m}_j)^T \mathbf{C}_j^{-1}(\mathbf{x} - \mathbf{m}_j)$$

Como el término  $n/2 \ln(2\pi)$  es común para todas las clases, puede ser eliminado

$$d_j(\mathbf{x}) = \ln[P(\omega_j)] - 1/2 \ln[|\mathbf{C}_j|] - 1/2(\mathbf{x} - \mathbf{m}_j)^T \mathbf{C}_j^{-1}(\mathbf{x} - \mathbf{m}_j)$$

para  $j = 1, 2, \dots, M$ . Estas ecuaciones representan las funciones de decisión de Bayes para clases Gaussianas. Las superficies de decisión de estas funciones son hipercuadráticas, y coloca una superficie de decisión de segundo orden entre cada par de clases.

Si todas las matrices de covarianza fueran igual para todas las clases,  $C_j = C$ , y eliminando todos los términos independientes de  $j$ , las funciones de decisión se transforman en

$$d_j(x) = \ln[P(\omega_j)] + x^T C^{-1} m_j - \frac{1}{2} m_j^T C^{-1} m_j$$

Si además la matriz de covarianza es igual a la matriz identidad,  $C = I$ , y las clases tienen la misma probabilidad de ocurrencia,  $P = 1/M$ , la función de decisión se reduce a

$$d_j(x) = x^T m_j - \frac{1}{2} m_j^T m_j$$

Esta función de decisión es similar a la del clasificador de distancia mínima. Por lo tanto, el clasificador de distancia mínima es un clasificador de Bayes si las clases son Gaussianas, todas las matrices de covarianza son igual a la matriz identidad, y por último, si todas las clases tienen la misma probabilidad de ocurrir. Este tipo de clasificador es óptimo en el sentido de que comete la menor cantidad de errores de clasificación, sin embargo, para que se cumpla la optimalidad se restringe demasiado el problema, las condiciones obviamente se cumplen sólo en el mejor de los casos, lo cual, por lo general, no se da con regularidad, o de alguna manera se manipula el dominio del problema para forzarlo a cumplir estas condiciones.

Los clasificadores aquí descritos son algunos de los más utilizados en el área del reconocimiento de imágenes en lo que se refiere a métodos basados en teoría de decisión. A continuación se presenta el método de reconocimiento utilizado en el presente trabajo.

## **5.7. Una técnica alternativa de reconocimiento: Las redes neuronales artificiales.**

Desde la invención de la computadora, los científicos computacionales han intentado que ésta adquiriera un comportamiento similar al de nosotros los humanos, que sea capaz de tomar decisiones a partir de ciertos eventos, sin necesidad de que una persona le diga qué hacer, y que a partir de un número de ilustraciones del trabajo que deba realizar, tenga la habilidad de enfrentar a situaciones nuevas, y no solamente enfrentarlas sino llevarlas a cabo de forma exitosa. En otras palabras, los científicos intentan dotar a una máquina con un “cerebro artificial”.

Para poder construir una réplica de alguna cosa, primero se debe conocer obviamente lo que se va a replicar. Por ejemplo, cuando alguna compañía saca un producto al mercado, digamos un chip, y otras compañías del mismo ramo desean vender un producto similar, aplican la llamada ingeniería inversa, lo que hacen es comprar un ejemplar del producto en el que están interesados, y estudian su funcionamiento para obtener “un clon” (un chip, en este caso, que funcione de forma similar al original) para así ellos poder fabricar su producto y venderlo. De forma análoga, para que los científicos puedan obtener un clon, por lo menos en el aspecto funcional, del cerebro primero deben conocer la forma en la que éste lleve a cabo sus funciones, si no en su totalidad, al menos las funciones en las que están interesados, como el razonamiento, la forma en la que interpreta imágenes, etc.

Hasta el momento, aún no se ha descubierto completamente como es que el cerebro humano realiza sus funciones, pero sí se han logrado avances, y más en lo que concierne al procesamiento de información obtenida mediante los sentidos. Existen varias ciencias que se encargan del estudio del cerebro, una de ella es la ciencia cognitiva, que, como su nombre

indica, se encarga del estudio del proceso de cognición del cerebro, y la cual es una ciencia interdisciplinaria, ya que reúne a investigadores de distintas áreas, como psicólogos, filósofos, lingüistas, científicos computacionales, entre otros. Otra ciencia encargada del estudio del cerebro es la neurociencia computacional, que trata de entender los procesos cerebrales mediante la ayuda de una computadora [Churchland y Sejnowski, 1992]. Los neurocientíficos computacionales estudian las regiones cerebrales en la que están interesados, realizan modelos de éstas regiones y simulan su trabajo en la computadora. Estos modelos están basados en el uso de redes neuronales artificiales, el tema principal de discusión de este trabajo. Sin embargo, en la neurociencia computacional, las redes neuronales artificiales tienen un uso diferente al que se les da en el caso del reconocimiento de patrones, ya que los modelos de simulación tratan parecerse lo más posible a la región cerebral que representan, tanto en estructura como en funcionamiento, así se mide su efectividad. En el caso del reconocimiento de patrones, el desempeño se mide en base a la rapidez y precisión con la que realiza la clasificación, sin importar si la topología de la red se asemeje a alguna región del encéfalo.

A continuación se da una breve introducción general a las redes neuronales artificiales, para posteriormente entrar a la etapa de reconocimiento de nuestro trabajo de investigación.

### **5.7.1. Definición de red neuronal artificial**

Empecemos dando la definición de red neuronal artificial, una de las más populares es la definición de Teuvo Kohonen [Kohonen, 1988]:

Las redes neuronales artificiales son redes interconectadas masivamente en paralelo de elementos simples (usualmente adaptativos) y con organización jerárquica las cuales intentan interactuar con los objetos del mundo real del mismo modo que lo hace el sistema nervioso biológico.

## 5.7.2. Características de las redes neuronales artificiales

Analicemos la definición anterior:

*Son redes de elementos simples*, las redes neuronales artificiales están compuestas por un conjunto de nodos (los elementos simples), debido a que estas redes tienen como inspiración las redes neuronales biológicas, estos nodos corresponde a las neuronas, y también se les conoce como neuronas artificiales, o unidades de procesamiento. Y, como su contraparte biológica, su trabajo es recibir información, procesarla y transmitirla a otros nodos.

*Con organización jerárquica*, las redes neuronales artificiales están organizadas en diferentes capas, cada capa tiene un número determinado de nodos. Esto define la arquitectura de la red, la cual es una característica de la misma. En base a la arquitectura existen dos tipos de redes, las *de propagación hacia adelante*, y las *recurrentes*. En el caso más simple, una red puede tener solamente dos capas, una de entrada, encargada de recibir los datos del exterior, y una de salida, que se encarga de dar los resultados. En otros casos las redes contiene capas intermedias, las cuales realizan los procesamientos internos para llevar a cabo tareas complejas. En la figura 30 se muestra un modelo genérico de red neuronal.

*Interconectadas masivamente en paralelo*. Definidos los dos tipos de arquitecturas, en el primer tipo de redes, las de flujo hacia adelante, los nodos de una capa están totalmente conectados con los nodos de la siguiente capa, en algunos casos nodos de una misma capa también se conectan, pero nunca con nodos de una capa anterior. En contraste, en las redes recurrentes cualquier nodo puede estar conectado con cualquier otro, incluso con nodos de capas anteriores.

*Intentan interactuar con los objetos del mundo real del mismo modo que lo hace el sistema nervioso biológico.* En otras palabras, tratan de simular el comportamiento de las redes neuronales biológicas, dando una respuesta a uno o varios estímulos provenientes del exterior.

### 5.7.3. Aprendizaje

Y ¿cómo saben responder correcta o aproximadamente a una entrada en particular?, las redes neuronales llevan a cabo su trabajo debido a que son ‘capacitadas’ previamente, es decir, aprenden la información relevante para realizar su tarea. Este proceso de ‘capacitación’ es llamado *entrenamiento (o aprendizaje)*. En las conexiones entre nodos existe un valor asociado a ellas, llamado *peso*, y en ellos está contenida la información que permite a la red desempeñar su trabajo, en otras palabras, en las conexiones se encuentra distribuído su conocimiento, al igual que en las sinapsis en las redes biológicas. Existen dos tipos de pesos, los *inhibitorios* (pesos con valor negativo), y los *excitatorios* (pesos con valor positivo). En el proceso de entrenamiento estos pesos son modificados hasta que la red responda correctamente a las diferentes entradas.

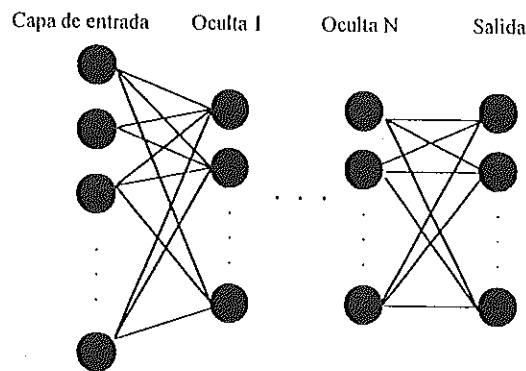


Fig. 30. Arquitectura de una red de flujo hacia adelante.

### 5.7.3.1 Tipos de aprendizaje

Existen dos tipos de aprendizaje, el aprendizaje *supervisado* y el *no supervisado*. En el primer caso la red cuenta con un agente externo (un maestro) que le indique las respuestas que debe obtener a partir de ciertas entradas, se le proporciona a la red la entrada y la salida deseada que debe calcular, y en base a ello realiza las modificaciones de sus pesos hasta obtener los resultados correctos para cada entrada, mediante reglas de aprendizaje que incrementan o decrementan el valor del peso en cuestión en base al error cometido por la neurona a la que está asociado. Existen varios algoritmos de entrenamiento supervisado, uno de los más utilizados en redes de flujo hacia adelante es el de retropropagación [Cummins y Dellarosa, 2000]. En el aprendizaje no supervisado, la red no cuenta con un maestro, simplemente se le dan las entradas que debe reconocer y ella misma modifica sus pesos, de tal forma que entradas que tengan características similares sean agrupadas en una sola clase, cada neurona de salida representa una de estas clases, tanto en redes recurrentes como en redes de flujo hacia adelante.

### 5.7.4. Procesamiento de la información

La función de las neuronas biológicas es recibir información (señales químicas), procesarla y transmitirla a todas aquellas neuronas con las que forma una conexión. Las neuronas artificiales (los nodos de la red), realizan un trabajo similar. En este caso la información que reciben es un valor numérico. Este valor numérico, llamado *entrada neta del nodo*, es el resultado de la suma de la multiplicación de los valores que transmiten los nodos fuente por los pesos asociados con las conexiones que van hacia el nodo receptor. Este proceso es representado matemáticamente mediante la ecuación

$$Y\_netaj = \sum_{i=1}^M x_i W_{ij}$$

donde  $Y_{netaj}$  es la entrada neta a la  $j$ -ésimo nodo (receptor),  $x_i$  es la salida del  $i$ -ésimo nodo,  $W_{ij}$  es la matriz de pesos, que, como su nombre lo indica, es una matriz que contiene los pesos de las conexiones entre un par de capas, el índice  $i$  recorre los nodos en la capa transmisora, y  $j$  los nodos de la capa receptora.

Los nodos pueden encontrarse en uno de dos estados, en *estado de activación*, cuando genera una señal, o en *estado de reposo*, no genera un valor. Este estado depende de una función, llamada *función de activación*, y la cual determina si el nodo va a ser activado o no, dependiendo del valor de la entrada neta al nodo en cuestión.

En algunos tipos de redes, los nodos solo pueden generar valores discretos, estos valores son bivalentes (0, 1) o (-1, 1), el nodo responde con un 1 al activarse, y con un 0 (ó -1) al estar en reposo. En este caso, se dice que es un proceso de todo o nada (responde o no responde), y está determinado por un umbral, el cual debe sobrepasar la entrada neta al nodo, para que éste se active. Una función de activación para este tipo de nodos tiene la forma

$$f(\text{neta}) = \begin{cases} 0 & \text{si neta} \leq \text{umbral} \\ 1 & \text{si neta} > \text{umbral} \end{cases}$$

Hay otros tipos de redes donde los nodos generan valores continuos, aquí no se utiliza el criterio del umbral, los valores generados se encuentran en el rango  $[0, 1]$  o  $[-1, 1]$ . En este caso los nodos utilizan una función no lineal continua, una de las más utilizadas es la función sigmoideal (una función con forma de S), para el caso binario la función tiene la forma

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

#### 5.7.5. Ventajas de las redes neuronales artificiales sobre las técnicas tradicionales

Existen ciertas características que distinguen a las redes neuronales de otras técnicas. Una de ellas es la tolerancia a fallas. En el caso biológico, diferentes neuronas en diferentes

regiones cerebrales mueren por varias causas, y a pesar de ello, las regiones desempeñan su trabajo perfectamente (dependiendo del número de neuronas perdidas), esto es debido a que otras neuronas en buen estado llevan a cabo la función que les correspondía a las neuronas muertas, además del trabajo que éstas desempeñan, a este fenómeno se le llama *plasticidad*. Un ejemplo de esta situación, muestra que en el sistema visual de las ranas, si se daña el 50% de la región cerebral conectada con la retina, encargada de la interpretación de las imágenes, toda la imagen captada por la retina es representada en el 50% funcional [Rosenzweig y Leiman, 1989]. De forma similar, en las redes neuronales artificiales, si algunos nodos fallan, otros nodos realizan su trabajo, evitando que la red se colapse y falle totalmente.

Otra característica importante es la tolerancia al ruido, esto se refiere a que algunos patrones a reconocer tiene algunas distorsiones, en el caso del reconocimiento de imágenes, a veces las imágenes capturadas son de mala calidad, y así se realiza el procesamiento sobre ellas, al llegar a la etapa de reconocimiento, la red es capaz de clasificar correctamente el patrón correspondiente, a pesar de la distorsión de la imagen.

Una última característica es la capacidad de generalización. La generalización es la cualidad que tienen las redes neuronales de poder responder correctamente a patrones que no se le habían presentando con anterioridad.

#### **5.7.6. Aplicaciones de redes neuronales utilizando una computadora secuencial**

Hasta el momento se ha hablado del paralelismo masivo inherente en las redes neuronales. Sin embargo, no solamente mediante el uso de una supercomputadora se implementan las redes neuronales, pueden simularse en una computadora secuencial tradicional, utilizando los diferentes algoritmos para cada arquitectura que existe actualmente.

### **5.7.7. Las redes neuronales utilizadas en el caso de estudio**

Existen diferentes redes neuronales en la actualidad. A cada una la distinguen las características antes mencionadas; su arquitectura, su algoritmo de aprendizaje, la función de salida utilizada. Se implementó una red para reconocer dígitos y un conjunto de redes para reconocer letras.

#### **5.7.7.1. Características de las redes utilizadas**

Las redes utilizadas son redes con arquitectura de flujo hacia adelante de varias capas, el tipo de aprendizaje es supervisado, y utilizan la regla de aprendizaje de *descenso de gradiente*, los nodos generan respuestas continuas utilizando como función de activación la función sigmoideal logística. Se trata de la llamada *red de retropropagación (backpropagation)*.

#### **5.7.7.2. Algoritmo de aprendizaje**

El nombre de retropropagación proviene del algoritmo de aprendizaje utilizado por la red. En él se involucran tres etapas. La primera consiste en la propagación hacia adelante de la información procesada a partir del patrón de entrada, los nodos de la primer capa oculta procesan la información proveniente de la capa de entrada, y la transmiten a las neuronas de la siguiente capa, así sucesivamente hasta que los nodos de la capa de salida procesan información y dan una respuesta.

Una vez obtenida la respuesta inicia la segunda etapa, comparando la salida obtenida por la red contra la salida deseada, y se estima el error cometido. El error es propagado de la capa de salida hacia la última capa oculta, es decir, se retropropaga, de aquí el nombre de la red, distribuyendo el error cometido por cada nodo de salida con cada nodo al que se conecta en la capa anterior. Los nodos de la capa previa a la de salida reciben la información proveniente de

los nodos de la capa de salida del mismo modo que lo hace en la etapa propagación hacia adelante, realiza la suma de la multiplicación del error proveniente de los nodos de salida con los que forma una conexión por el peso asociado a esa conexión, y se va propagando de capa en capa hasta llegar a la primer capa oculta. Este proceso puede verse como si se invirtiera la red y ahora la capa de entrada es la capa de salida, y la entrada recibida es un vector que contiene los porcentajes de error cometido por el nodo correspondiente.

La última etapa consiste en la actualización de pesos utilizando la regla de descenso de gradiente. Los pesos se ajustan de acuerdo a la cantidad proporcional a la primer derivada (el gradiente) de la función de error. Es sabido que el vector gradiente apunta hacia la tasa máxima de cambio de una función, como el objetivo es minimizar el error, entonces se toma el negativo del gradiente, que apunta a la tasa mínima de cambio, el mínimo, de la función de error en este caso.

Matemáticamente, y simplificando la red al caso donde se tiene una sola capa oculta, este proceso se puede ver de la siguiente forma:

*Propagación hacia adelante.-*

Se calculan las entradas netas de cada nodo en la capa oculta

$$\text{netaj} = \sum_{i=1}^M x_i W_{ij}$$

donde  $j=1, 2, \dots, N$  recorre los nodos de la capa oculta,  $i$  recorre los nodos de la capa de entrada.  $x$  es el vector que contiene las salidas de la capa de entrada, y  $W$  es la matriz de pesos entre las dos capas.

Y se obtienen las activaciones de cada nodo

$$f(\text{netaj}) = 1 / [ 1 + \exp(-\text{netaj}) ]$$

Los nodos de la capa de salida calcula sus entradas netas:

$$\text{netak} = \sum_{j=1}^N z_j V_{jk}$$

en este caso  $z$  es la salida de la capa oculta, y  $V$  la matriz de los pesos entre la capa oculta y la de salida.  $k$  recorre los nodos de la capa de salida.

Se obtienen las salidas de cada nodo

$$y_k = f(\text{net}_k) = 1 / [ 1 + \exp(-\text{net}_k) ]$$

*Cálculo del error y retropropagación*

Se calcula el error cometido por cada nodo de salida

$$\delta_k = (t_k - y_k) f'(\text{net}_k)$$

donde  $t$  es el vector de la salida deseada y  $f'(\text{net}_k)$  es la derivada de la función de activación, evaluada en el valor de  $\text{net}_k$ .

Calcula el incremento (o decremento) de los pesos para su actualización posterior

$$\Delta V_{jk} = \alpha \delta_k z_j$$

$\alpha$  es un valor conocido como tasa de aprendizaje, ayuda en la rapidez de convergencia de los pesos.

Cada nodo calcula la suma de error del nodo al que se conecta en la capa de salida por su respectivo peso

$$\delta_{\text{net}_j} = \sum_{k=1}^N \delta_k V_{jk}$$

Y calcula su error

$$\delta_k = \delta_{\text{net}_j} f'(\text{net}_k)$$

y el valor de modificación de sus pesos

$$\Delta W_{ij} = \alpha \delta_j x_i$$

*Modificación de pesos*

Cada unidad de salida actualiza sus pesos

$$V_{jk}(\text{nuevo}) = V_{jk}(\text{anterior}) + \Delta V_{jk}$$

Lo mismo ocurre con los nodos de la capa intermedia

$$W_{ij}(\text{nuevo}) = W_{ij}(\text{anterior}) + \Delta W_{ij}$$

Este proceso es repetido hasta que se encuentre un mínimo de la función de error. Al encontrarse el mínimo, significa que los pesos han encontrado un balance, y que clasificará correctamente tanto a los patrones del conjunto con el que fue entrenada, como a nuevos patrones que se le presenten. Se utiliza un cierto criterio como condición de paro, en nuestro caso se utilizó la función de cuadrados medios. Se obtiene la suma de los cuadrados de los errores cometidos por la red en cada patrón en una iteración, divididos entre dos, si esta suma es menor que un valor de error preestablecido, se termina la etapa de aprendizaje.

Como conjunto de aprendizaje se utilizaron 40 patrones de dígitos obtenidos durante la etapa de segmentación, corresponden a 4 por cada dígito.

### **5.7.7.3. Red neuronal para el reconocimiento de dígitos**

#### *Arquitectura*

Como se había mencionado anteriormente, la arquitectura de una red consiste del número de capas de nodos con las que cuenta, y el número de nodos por capa. El número de capas determina la complejidad de las fronteras de decisión entre clases, a mayor número de capas, más complejas son las fronteras de decisión, y mejor desempeño tiene la red. El número de nodos por capa oculta influye en la eficacia de aprendizaje y de generalización de la red [Hilera y Martínez, 1995].

El número de nodos de la capa de entrada depende del esquema de codificación utilizado, en nuestro caso, se tienen 21 nodos de entrada, correspondientes a los 21 componentes del vector obtenido por la codificación mediante el esquema de la malla. El número de nodos de salida en este caso es de 10, uno por cada dígito.

Para poder encontrar la arquitectura más óptima se hicieron una serie de pruebas a diferentes redes con diferentes cantidades de capas ocultas, y con diferente número de nodos en cada capa, se analizó el tiempo que cada una tomaba para entrenarse, y una vez entrenada se analizó su desempeño. En el análisis del entrenamiento de cada red se utilizaron varios valores para la tasa de aprendizaje. En cuanto al desempeño, se calculó el tiempo de respuesta de cada red.

La arquitectura seleccionada cuenta con una sola capa oculta con 18 unidades de procesamiento. En la figura 31 se muestra gráficamente la red utilizada para el reconocimiento de dígitos.

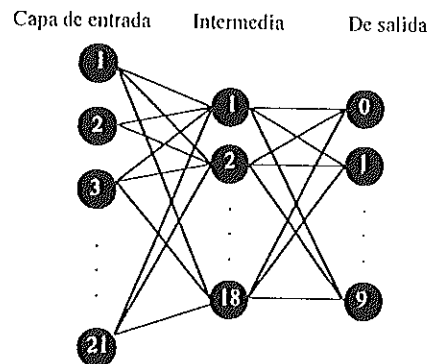


Fig. 31. Arquitectura de la red para el reconocimiento de dígitos.

### *Resultados obtenidos durante el proceso de selección y el análisis de desempeño*

A continuación se presentan los resultados obtenidos durante las pruebas realizadas a tres redes con una, dos y tres capas ocultas, modificando el número de nodos por capa, así como los valores de la tasa de aprendizaje. En las siguientes tablas se muestra la relación. La primer tabla corresponde a las redes con una sola capa oculta, la segunda a las redes con dos capas ocultas, el número de nodos en cada capa se encuentran separados por un guión, el primer número corresponde a la primer capa, y el segundo número a la segunda capa oculta. En la notación utilizada  $E_p$  es el número de ciclos realizadas durante el aprendizaje, T corresponde al

tiempo de entrenamiento y P es el porcentaje de patrones reconocidos correctamente utilizando un conjunto de prueba con un total de 115 patrones.

Tasa \ # Nodos	11	13	15	18	20
0.05	Ep=13276 T = 330 seg P ≈ 83%	Ep=12862 T = 382 seg P ≈ 84%	Ep=12308 T = 408 seg P ≈ 81.7%	Ep=11221 T = 382 seg P ≈ 80.9%	Ep=12933 T = 559 seg P = 80%
0.25	Ep=2823 T = 71 seg P ≈ 83%	Ep=2812 T = 82 seg P ≈ 84%	Ep=2607 T = 87 seg P = 80%	Ep=2393 T = 94 seg P ≈ 86%	Ep=1949 T = 84 seg P ≈ 84%
0.5	Ep=1284 T = 32 seg P ≈ 84%	Ep=1213 T = 36 seg P ≈ 77.4%	Ep=1301 T = 44 seg P ≈ 81.7%	Ep=1311 T = 51 seg P ≈ 86%	Ep=973 T = 42 seg P ≈ 85.3%

Tabla I. Resultados obtenidos utilizando redes con 11, 13, 15, 18 y 20 nodos, para la clasificación de dígitos

En las tablas I y II se puede observar que al aumentar el número de capas a dos, el desempeño de la red se degradó, disminuyendo el porcentaje de patrones clasificados correctamente, por lo que se decidió utilizar solamente una sola capa oculta con 18 nodos, ya que fue la arquitectura que mostró un mayor porcentaje de éxito. Se realizaron pruebas también a una red con tres capas ocultas con 18, 15 y 12 nodos respectivamente, pero no se obtuvieron mejores resultados que los correspondientes a la red seleccionada. En cuanto a la tasa de aprendizaje, se seleccionó un valor de 0.25, ya que al utilizar un valor de 0.05 el aprendizaje fue demasiado lento, y las redes mostraron un desempeño similar. Al escoger 0.5, el aprendizaje fue más rápido, pero se

sacrificaba precisión, es decir, al clasificar un patrón, el valor generado por la neurona de salida correspondiente a la clase de dicho patrón es muy bajo.

Tasa \ # Nodos	15-12	18-15	20-18
0.1	Ep=4947 T = 231 seg P ≈ 74.8%	Ep=4041 T = 243 seg P ≈ 84%	Ep=3661 T = 265 seg P ≈ 77.4%
0.25	Ep=2064 T = 96 seg P ≈ 83%	Ep=1556 T = 94 seg P ≈ 78.3%	Ep=1731 T = 132 seg P = 80%
0.5	Ep=1036 T = 48 seg P ≈ 80%	Ep=940 T = 56 seg P ≈ 84%	Ep=701 T = 51 seg P ≈ 74%

Tabla II. Resultados obtenidos de redes con 2 capas ocultas para la clasificación de dígitos

#### 5.7.7.4. Redes neuronales para la clasificación de letras

Se sabe de antemano el patrón de posiciones de dígitos y letras en las matrículas que se tomaron en cuenta (las fronterizas), y que es de la siguiente forma:

Para automóviles familiares, un conjunto de tres dígitos, en seguida un conjunto de tres letras, donde la primera siempre es una N, y por último un dígito. Una matrícula genérica sería de la siguiente manera

$$D1 D2 D3 N L2 L3 D4$$

donde D denota un dígito y L una letra

Para vehículos de carga (pick-ups, camiones, etc.), un conjunto de tres letras, de las cuales la primera resulta ser una Z, y posteriormente un conjunto de 4 dígitos

*Z L2 L3 D1 D2 D3 D4.*

Tomando esta información en cuenta ya se sabe a qué red enviar cada caracter, simplemente se debe verificar si el primero corresponde a una Z o a un dígito. De esta forma, se tiene una red exclusivamente para identificar una Z, que nos indica simplemente si los datos del patrón que se le presenta corresponden o no a esta letra. Si la respuesta de esta red es positiva, significa que los siguientes dos caracteres también son letras, e inmediatamente los enviamos a una de las redes para reconocer letras, y los 4 caracteres restantes se envían a la red para clasificar dígitos.

Si la respuesta es negativa, se tiene que mandar a la red de dígitos, al igual que los dos siguientes caracteres, y el último de la matrícula. Los siguientes tres caracteres a las redes para reconocer letras.

De acuerdo a estadísticas realizadas sobre un conjunto de aproximadamente 500 matrículas que fueron recabadas en trabajo de campo, anotando la mayor cantidad de matriculas posibles de vehículos que se observaron por toda la ciudad, y las cuales se muestran en las tablas III y IV, se obtuvo información acerca de qué letras son más probables de aparecer enseguida de una Z, en el caso de matrículas de vehículos de carga, y enseguida de una N, en el otro caso. La K y la J son las más probables a encontrarse después de una Z. En seguida de una N las letras con mayor probabilidad de presentarse son las letras U, V y T. De tal forma que se construyó una red para reconocer exclusivamente estas letras. La última red se utilizó para el reconocimiento de las letras restantes.

El reconocimiento de letras se lleva a cabo utilizando diferentes redes debido a que si se tuviera una sola red para todas las letras, ésta sería demasiado grande, afectando el proceso de

aprendizaje y el de reconocimiento, le tomaría mas tiempo a la red aprender, y una vez entrenada, dar una respuesta.

### *Arquitectura*

Al igual que en el caso de la red para el reconocimiento de dígitos, fue necesario llevar a cabo pruebas sobre distintas redes con distintas arquitecturas para poder seleccionar las más apropiadas. Todas tienen en la capa de entrada 21 unidades de procesamiento.

En el caso de la red para reconocer la Z, se utiliza una red sin capas ocultas, y un solo nodo de salida, similar a un perceptrón, solamente difiere en la forma de entrenamiento. Para el caso del reconocimiento de las letras enseguida de la primer letra, se utiliza una red con una capa oculta, con 11 nodos en ésta, y 5 nodos en la capa de salida, uno para cada clase de letra. En la tercer red también se tiene una sola capa oculta que cuenta con 20 unidades. Debido a que en el conjunto muestra de imágenes de matrículas con que se cuenta no aparecen todas las letras de alfabeto, ésta red reconoce solamente 14 letras, que son la B, D, E, F, G, H, L, M, N, P, S, W, X, y la Y, por lo tanto tiene 14 nodos en la capa de salida.

### *Resultados obtenidos durante el proceso de selección y el análisis de desempeño*

La red para el reconocimiento de Z aprendió en un periodo de 2 segundos, bastante rápido. No solamente se entrenó con patrones de Z, también fue necesario indicarle qué patrones no debía reconocer (que su respuesta fuera 0 o aproximada a 0), en especial patrones correspondientes a dígitos, ya que esta red también va a recibir este tipo de patrones y podría clasificar como Z al 2, por ejemplo, afectando el reconocimiento en general. En el conjunto de entrenamiento se contó con un total de 21 pares de entrada-salida deseada, de los cuales 3

Matrículas

1 9 2 N U H 9	5 2 5 N U L 2	3 5 0 N U K 8
6 8 5 N U L 1	2 6 3 N S T 1	8 5 4 N U V 7
0 7 8 N V D 2	4 2 8 N U H 5	3 9 0 N S M 5
5 7 2 N U K 5	8 1 7 N V L 9	6 8 1 N U K 9
3 5 6 N U K 6	4 1 0 N V D 2	3 8 4 N V G 6
3 5 3 N U V 9	1 1 1 N U W 6	3 9 0 N U J 6
4 4 6 N U K 6	5 7 4 N U H 7	4 8 9 N V L 9
2 6 6 N U K 4	0 3 8 N R P 5	9 3 7 N R W 6
8 4 1 N V B 1	3 2 1 N T H 1	7 1 1 N U V 9
4 1 4 N U X 4	0 3 7 N U K 2	7 7 3 N U W 5
3 3 1 N T P 7	8 3 2 N U K 6	6 4 4 N T B 2
3 0 6 N U W 1	9 8 1 N U V 9	3 1 7 N U L 1
2 1 6 N U K 2	4 1 4 N V D 3	4 9 1 N U A 2
3 4 7 N U J 2	6 4 7 N S G 5	2 9 8 N R G 4
0 0 3 N V D 7	0 0 5 N U W 7	3 3 5 N U L 1
5 6 2 N V L 4	9 0 1 N U W 7	6 0 5 N R X 1
8 2 9 N V B 1	6 2 6 N V D 8	3 6 9 N S N 2
0 2 9 N S R 3	8 7 4 N R F 2	3 2 3 N U W 2
2 6 0 N U J 8	3 7 2 N V B 4	0 3 5 N R X 2
0 2 2 N V K 4	9 0 8 N V B 3	5 5 1 N U J 9
3 2 9 N U V 9	5 5 0 N U H 9	5 7 2 N U K 5
1 6 2 N T N 7	6 9 8 N U K 5	6 5 4 N T P 5
2 2 5 N S P 3	6 1 5 N U W 6	0 2 9 N V D 3
3 3 3 N U Z 2	8 5 8 N T G 4	1 2 3 N V R 5
3 8 5 N T P 9	1 9 2 N V D 2	0 0 8 N U D 2
8 5 3 N V L 9	8 2 8 N U T 2	5 4 4 N U V 9
3 1 1 N V B 2	8 8 6 N V K 4	3 8 5 N U K 1
8 5 7 N V K 4	8 3 9 N U K 3	5 7 5 N V L 4
4 9 4 N V D 2	7 5 9 N U L 2	9 3 1 N U J 3
5 7 8 N U K 7	7 0 1 N U S 9	8 8 6 N U K 1
6 5 8 N S E 1	3 5 2 N U K 7	6 8 5 N U H 6
9 6 8 N T L 6	6 5 1 N U T 2	0 7 6 N T C 7
0 1 9 N U L 2	9 9 1 N V H 8	4 7 8 N U F 7
3 7 6 N U J 9	1 7 4 N R G 6	6 8 5 N U H 6
8 0 0 N V H 8	6 8 1 N R G 4	5 4 3 N U W 4
0 9 6 N U W 5	8 0 1 N T G 1	2 2 2 N U U 3
8 7 5 N U L 2	9 3 9 N U K 3	2 8 9 N V B 3
9 5 7 N V H 7	9 3 7 N U J 7	8 9 0 N V D 2
8 2 3 N U W 5	8 4 9 N T N 9	3 6 3 N T T 7
7 7 3 N U K 3	0 9 4 N U L 1	0 7 6 N T P 7
6 0 9 N V F 8	7 9 1 N U K 8	0 2 9 N U K 8
0 4 1 N U K 9	4 4 3 N V D 8	0 3 9 N U J 4
0 9 7 N T N 7	2 2 4 N T H 1	6 3 3 N V R 5
2 9 8 N V L 9	0 9 7 N T B 9	1 0 9 N U T 2
6 9 2 N U W 2	6 3 8 N U K 4	2 0 9 N U J 1
6 3 0 N V D 3	4 6 5 N U W 3	8 2 8 N U J 7
0 2 1 N U J 3	8 3 0 N V L 9	3 3 7 N V P 2
1 3 5 N U H 4	0 3 0 N V R 5	
4 4 4 N U W 7	6 8 6 N R M 7	
3 3 0 N U K 4	0 8 3 N U W 1	

4 3 7 N U J 1	5 4 6 N S G 4
7 3 7 N T P 5	9 1 9 N U J 9
7 1 2 N V M 2	0 6 2 N V D 3
6 5 2 N T N 4	1 0 2 N U W 1
8 5 6 N V B 4	4 6 8 N U J 8
8 2 5 N R T 9	6 1 7 N V P 2
6 2 7 N U J 6	2 2 7 N U K 4
4 8 2 N U K 5	5 4 8 N U K 7
2 8 5 N T P 9	9 6 8 N T C 7
7 0 1 N V B 1	2 0 4 N T N 7
2 5 6 N V L 1	3 3 8 N R B 7
7 7 4 N U V 9	7 3 9 N U T 1
1 2 6 N T N 7	1 2 7 N V K 7
5 5 6 N V A 8	0 6 2 N U V 7
8 3 4 N R L 3	9 5 1 N V D 3
8 4 9 N R B 8	7 1 0 N V K 4
2 2 2 N U K 4	4 2 7 N T K 7
7 5 0 N U K 2	5 0 0 N T N 9
6 8 6 N R M 7	6 5 3 N S E 2
9 6 5 N U J 1	5 0 0 N U K 1
2 1 4 N V E 8	1 1 0 N U J 9
9 6 0 N V N 5	8 3 4 N U J 5
5 2 4 N U K 1	6 4 4 N V P 2
5 1 4 N U J 5	7 8 2 N U L 2
3 5 9 N U K 4	4 2 1 N S E 3
3 1 8 N R B 9	0 2 7 N U T 2
3 5 9 N L V 9	4 1 9 N U G 2
5 3 0 N T H 6	4 1 2 N S E 2
0 1 3 N V M 8	3 0 0 N U J 4
4 6 7 N U J 8	6 3 9 N U J 7
8 8 6 N U T 1	
8 7 9 N V P 2	A B C D E F G H J
0 0 9 N U M 5	2 13 3 14 6 3 9 15 23
8 6 9 N S T 2	K L M N P R S T
0 4 5 N U K 5	39 21 8 232 12 22 22 37
5 4 7 N U H 6	U V W X Y Z
3 3 0 N U J 9	112 65 16 3 0 1
5 4 8 N U K 2	1 2 3 4 5 6 7 8 9
6 5 2 N V H 8	87 108 97 88 87 85 86 92 91
1 6 1 N V L 1	0
9 7 9 N V R 2	71
6 8 6 N R M 7	
4 9 8 N V E 6	NU NV NT NS NR
8 0 4 N R T 8	111 56 26 12 17
7 4 3 N V M 8	
1 4 6 N T C 2	

Tabla III. Matrículas correspondientes a vehículos familiares. Muestra la ocurrencia de cada letra y número

Matrículas

Z K K 9 8 2 7	Z J H 6 6 7 4	Z K M 2 5 5 0
Z K K 8 3 5 7	Z K L 6 7 3 1	Z K S 8 4 1 0
Z K L 6 1 9 9	Z K P 7 4 9 6	Z K L 2 4 4 6
Z K L 5 7 9 2	Z K W 6 2 1 8	Z K K 5 6 6 1
Z K S 8 8 2 3	Z K K 8 3 1 1	Z K M 5 8 2 6
Z K U 3 7 5 4	Z K L 7 8 3 7	Z J X 8 2 3 4
Z K P 7 7 4 0	Z K M 2 2 2 2	Z K K 6 7 2 2
Z K N 5 9 3 5	Z K X 5 7 3 5	Z K K 9 9 4 2
Z K T 4 9 5 2	Z K N 7 9 8 9	Z K P 8 0 0 2
Z K K 6 5 6 5	Z K L 7 4 7 7	Z J W 7 2 5 5
Z K L 3 1 0 7	Z K W 6 0 5 5	Z K L 1 1 2 8
Z K K 8 7 2 4	Z K L 3 1 1 2	Z K M 2 1 3 5
Z K T 4 1 5 5	Z J S 8 7 2 8	Z K V 6 8 5 6
Z K L 5 4 4 8	Z K N 4 0 1 6	Z K K 6 0 9 7
Z K T 4 9 8 8	Z K P 7 4 4 8	Z K V 6 9 0 1
Z K S 8 9 0 6	Z K X 2 4 5 0	Z K L 1 3 9 2
Z K T 8 7 6 3	Z K L 4 4 3 9	Z K A 1 5 5 6
Z K P 0 0 7 7	Z K L 1 8 5 6	Z J N 3 5 2 9
Z K L 7 5 2 5	Z K P 8 1 4 7	Z K W 6 4 5 0
Z K N 5 3 4 6	Z K L 6 0 8 2	Z K K 5 7 8 0
Z J V 2 4 0 1	Z K T 4 1 3 7	Z K N 6 4 8 3
Z K M 2 3 0 0	Z K V 2 5 1 7	Z K L 1 3 4 3
Z M H 9 2 7 7	Z J M 5 2 5 3	Z K M 2 5 5 0
Z K T 4 8 8 6	Z K U 3 7 3 3	Z K N 6 8 2 5
Z K L 3 0 8 8	Z K V 6 1 3 7	Z K S 8 9 6 9
Z K L 2 4 9 9	Z K L 2 9 1 4	Z K K 8 9 7 9
Z K N 7 0 7 1	Z K L 1 1 6 4	Z K N 5 0 5 9
Z K N 5 0 0 2	Z K T 4 2 8 5	Z K N 6 1 6 7
Z K L 3 2 6 5	Z K W 2 4 8 0	Z K N 5 8 5 5
Z L D 4 9 4 0	Z K S 8 4 6 1	Z K L 4 5 3 7
Z K W 5 7 6 0	Z K M 1 2 0 2	Z K N 7 9 0 0
Z K M 2 2 4 4	Z K M 1 3 6 2	Z K K 8 5 4 0
Z K V 6 0 8 1	Z K T 4 3 1 1	Z J W 3 9 6 0
Z K L 6 9 6 9	Z K T 4 2 8 5	Z K T 4 5 2 9
Z K L 1 8 8 8	Z K G 3 2 3 3	Z K K 6 5 7 3
Z K L 5 9 1 9	Z V Y 1 1 3 6	Z J W 7 2 4 5
Z K V 7 3 6 9	Z J M 4 9 8 2	Z K M 1 5 4 0
Z J Y 8 0 2 7	Z K T 4 2 7 4	Z K L 6 9 5 0
Z K N 5 9 4 3	Z K L 2 9 3 8	Z K K 6 1 9 5
Z K L 5 3 3 4	Z J K 7 3 3 3	Z K X 2 4 5 0
Z K V 6 1 4 3	Z K X 3 3 3 0	Z K X 7 4 8 6
Z J M 5 9 9 7	Z K S 8 3 5 2	Z K P 8 9 2 3
Z K N 7 2 5 2	Z L P 3 1 9 8	Z K U 3 4 7 3
Z K B 7 6 5 0	Z K T 4 6 4 1	Z K L 5 0 1 7
Z K L 5 7 5 0	Z K M 1 2 4 7	Z K P 7 4 6 0
Z K V 6 1 7 0	Z K N 5 2 3 3	Z K T 4 0 0 9
Z K G 1 7 1 8	Z K W 6 0 6 3	Z K P 7 6 0 5
Z K A 1 5 5 6	Z K K 5 9 9 0	
Z K L 5 3 0 1	Z K V 6 0 9 8	
Z K N 7 8 0 9	Z L D 4 6 0 2	

Z M H 7 4 2 2	Z K S 7 9 7 9
Z K P 7 1 9 0	Z K U 3 0 1 8
Z K K 5 4 3 9	Z K K 5 1 9 5
Z K K 5 3 5 6	Z K C 1 8 8 2
Z K T 4 1 9 1	Z K V 6 9 0 8
Z J M 4 0 0 9	Z J A 8 3 5 9
Z K M 2 5 0 7	Z K M 2 3 6 5
Z K S 5 6 2 7	Z K K 6 4 1 0
Z J S 6 2 0 6	Z K L 1 9 3 7
Z J N 2 7 5 1	Z K S 8 9 0 7
Z K M 2 8 5 2	Z K W 6 4 1 3
Z K W 4 9 6 6	Z K U 3 8 2 6
Z J M 4 8 4 7	Z K L 3 1 1 2
Z K M 2 8 3 8	Z K V 6 8 4 7
Z K W 9 8 9 7	Z K B 5 2 1 0
Z K T 4 9 4 4	Z K K 5 4 3 8
Z K L 1 5 5 0	Z K U 3 5 5 7
Z L P 5 4 8 9	Z K P 7 9 8 7
Z K L 3 8 8 9	Z K L 4 6 4 4
Z K N 7 4 7 5	Z K W 6 1 2 3
Z K U 9 2 5 7	Z M M 9 4 1 8
Z K L 3 7 3 2	Z K L 3 1 9 9
Z K F 9 7 7 4	Z K L 4 0 3 8
Z K N 6 4 9 6	Z K X 2 2 8 1
Z K K 8 7 8 8	Z M B 2 4 1 9
Z K X 2 2 5 8	Z K L 6 7 3 4
Z K W 6 2 6 7	Z K U 3 3 0 2
Z K K 7 2 0 9	
Z K L 4 5 1 7	
Z K U 4 0 7 2	
Z K P 7 5 5 9	
Z K S 8 2 5 9	
Z K M 2 8 7 8	
Z K T 4 1 8 2	
Z K N 5 4 5 6	
Z K T 4 5 8 7	
Z K K 5 5 6 1	
Z K P 8 3 5 8	
Z K N 6 7 1 3	
Z K M 2 8 5 7	
Z K W 6 0 0 5	
Z L M 3 6 3 5	
Z K M 2 9 0 2	
Z K L 1 9 2 8	
Z K L 6 0 8 5	
Z K M 2 1 4 0	

A	B	C	D	E	F	G	H	I	J	K	L	M
3	3	1	2	0	1	2	3	0	16	216	48	30
N	P	Q	R	S	T	U	V	W	X	Y	Z	
22	15	0	0	0	17	9	13	15	8	2	220	
1	2	3	4	5	6	7	8	9	0			
77	96	84	93	109	84	93	90	80	74			
ZK	ZJ	ZL	ZM									
192	16	5	4									

Tabla IV. Matrículas correspondientes a vehículos de carga. Muestra la ocurrencia de cada letra y número

correspondían a datos de una Z como entrada, y un valor de 1 como salida, y los 18 restantes a patrones de dígitos y letras donde la salida deseada es 0.

En la tabla V aparecen los resultados obtenidos al experimentar con cuatro redes con una sola capa oculta, en cada caso se tienen 11, 13, 15 y 18 unidades ocultas.

Se escogió la red con 11 nodos ocultos ya que a menos cantidad de nodos, menos recursos de la computadora ocupa, además que mostró una efectividad similar a las redes con mayor número de nodos. Las redes fueron entrenadas con un conjunto de 24 pares entrada-salida deseada, 15 correspondientes a las letras que está destinada a clasificar la red, y los 9 pares restantes a patrones donde no debe reconocer.

La arquitectura de la última red tiene una capa oculta con 20 unidades. Se hicieron pruebas con redes de 11, 13 y 15 unidades ocultas, pero su aprendizaje fue demasiado lento (a los 34.3 minutos después de haber iniciado el entrenamiento aún no se habían encontrado los valores adecuados de los pesos), por lo cual fueron descartadas. En las pruebas realizadas a la red seleccionada, ésta clasificó erróneamente solo dos de los patrones, de un conjunto de 44. Al presentarle patrones de letras que no debía reconocer, las asignó a una de sus clases, lo cual obviamente no debe suceder, una cantidad de 23 de un total de 54 patrones. Esto fue a causa de que durante el entrenamiento no se le indicó a la red aquellos patrones los cuales no debía clasificar. En la siguiente prueba se aumentó el conjunto de entrenamiento de 45 a 60. De estos 60 pares 18 correspondían a los patrones que no debe reconocer. Disminuyendo de 23 patrones mal clasificados a sólo 5. La red tomó un tiempo de 4.6 minutos para encontrar los valores de sus pesos. Como se puede observar el incremento en el porcentaje de éxito fue muy bueno al indicarle a la red aquellos patrones que no debe clasificar.

#Nodos	11	13	15	18
Ciclos	2491	2376	2394	2173
Tiempo	30 seg	34 seg	39 seg	45 seg
% éxito	92.13%	88.8%	93.3%	93.3%

Tabla V. Resultados del análisis de desempeño de redes con una sola capa oculta con 11, 13, 15, y 18 nodos, para la clasificación de letras

#### 5.7.7.4. Redes neuronales que utilizan la información de hoyos de los caracteres

Es necesario contar con redes adicionales, las cuales utilizan la información de los hoyos que pueden tener los caracteres (el esquema secundario de descripción). Para esto son utilizados un multiperceptron y un perceptrón para dígitos y para letras; el multiperceptrón para el caso donde se detecte un solo hoyo en el caracter a reconocer, y el perceptrón para aquellos caracteres donde se detecten dos hoyos.

##### *Arquitectura*

La arquitectura de las redes que utilizan el esquema de descripción secundario en el primer caso, y como cualquier perceptrón solamente cuenta con las capas de entrada y salida, la capa de entrada tiene solamente dos nodos, uno para cada descriptor, y que son la posición vertical del hoyo en el caracter y el área que cubre éste, ambos valores normalizados (se encuentran en el rango  $[0, 1]$ ). En la capa de salida se tienen cuatro nodos, tanto en el conjunto de dígitos como en el conjunto de letras se tienen un total de cuatro caracteres con un solo hoyo (0, 4, 6, 9 y A, D, P, R).

Para el caso de los caracteres con dos hoyos (8 y B), el perceptrón utilizado tiene 3 nodos en la capa de entrada, ya que se utilizan la distancia que hay entre los hoyos, y el área de cada hoyo. Como clasifica solamente un caracter, la capa de salida cuenta con un sólo nodo.

### *Análisis del desempeño de las redes*

Se decidió utilizar perceptrones en este caso debido a que la información a procesar es poca y por que mostraron una mayor rapidez en la etapa de aprendizaje que la que mostraron las redes de retropropagación. El tiempo de aprendizaje de cada par de redes fue de aproximadamente un minuto, para cada caso, utilizándose un conjunto de aprendizaje de seis pares de entrenamiento.

Las redes mostraron un buen desempeño al realizarse pruebas con un conjunto de 60 dígitos. De éste conjunto solamente clasificó erróneamente o no reconoció 8 dígitos, un porcentaje de éxito de aproximadamente el 87%.

Utilizando éste conjunto de redes adicionales disminuye la posibilidad de un mala clasificación de caracteres en el reconocimiento general, ya que si por algún motivo las redes principales no reconocen algún caracter con hoyos, estas redes llevaran a cabo ese trabajo de forma exitosa.

## **6. Resultados y conclusiones**

En este capítulo se presenta un análisis del desempeño de las técnicas utilizadas durante todo el procesamiento de imágenes partiendo de la segmentación, hasta llegar a la etapa de reconocimiento, de forma individual. Así como el análisis global del desempeño del sistema en general.

### **6.1 Etapa de segmentación**

Debido a que se trabajó con imágenes a color, la técnica de segmentación de umbralización de varias variables, donde se utilizan los componentes RGB del color de cada pixel de las regiones de interés, en conjunto con la condición de área de la placa para la eliminación de regiones no deseadas, resultó ser el apropiado para nuestro caso de estudio. Durante el paso de segmentación de la placa se tuvo éxito en aproximadamente el 92% de imágenes del conjunto muestra, con un total de 37 imágenes. El 8% donde no se obtuvieron resultados satisfactorios, se debió a causa de deformaciones en la placa del automóvil, algunas estaban dobladas y otras recortadas, reduciendo de esta forma su área, siendo eliminadas al comparar éste contra los valores preestablecidos, afectando de ésta forma la segmentación, y por lo tanto los pasos posteriores.

En la segmentación de los caracteres impresos en la placa, también se obtuvo un buen porcentaje de éxito utilizando la técnica anterior, junto con el emparejamiento de plantillas para encontrar un caracter, y apartir de ahí calcular las posiciones de los caracteres restantes; aproximadamente el 84% de las imágenes obtenidas durante la segmentación de la placa. Se tuvieron fallas en aquellas placas donde a causa de la binarización se introduce ruido, este problema es a causa de una iluminación muy alta o muy baja, provocando una localización de

partes de la placa que tenían correlación con las plantillas utilizadas, sin embargo, estas partes no correspondían a caracteres.

## **6.2. Descripción**

Gracias a la naturaleza del dominio del problema en que se trabajó, donde es raro encontrar un caracter rotado, el esquema de descripción basado en una malla resultó ser el apropiado a utilizar en nuestro trabajo ayudando al módulo de reconocimiento a tener un buen desempeño. Se realizaron pruebas con un conjunto de dígitos obtenido durante la segmentación, utilizando un módulo de reconocimiento de pruebas, donde se implementa un multiperceptrón de una sola capa, con una arquitectura de 21 unidades de entrada (las correspondientes a cada componente en el vector de descriptores), y 10 salidas (una para cada dígito). También se utilizó un segundo esquema de descripción, donde las características que se toman en cuenta son la posición vertical (arriba, abajo, en medio) del hoyo y el área que éste cubre, en el caso donde el caracter cuente con un hoyo. Si el caracter tiene dos hoyos, se utiliza la distancia entre éstos, así como el área de cada uno.

Antes de tomar la decisión de la utilización de estos esquemas de descripción se realizaron pruebas con otros esquemas. Primeramente se hicieron pruebas enviando al módulo de reconocimiento todo el conjunto de pixeles binarios que definen a cada caracter, pero la etapa de entrenamiento de las redes se vio afectada, siendo demasiado lenta, ya que era mucha información a procesar, se tenían 189 nodos de entrada en la red (los caracteres se encuentran en matrices de  $9 \times 21$ ), una vez entrenada la red, ésta no mostró un mejor desempeño que la que utilizó el esquema de malla. También se hicieron pruebas utilizando un conjunto de siete momentos invariantes, pero éstos no cumplieron con la condición de discriminación de los descriptores, tomando valores muy similares entre caracteres distintos, lo que dificultaría la clasificación, debido a esto se decidió descartarlos.

### **6.3. Reconocimiento**

En la etapa de reconocimiento se implementaron 4 redes neuronales de retropropagación, cada una con una arquitectura apropiada que utiliza información del esquema de descripción principal. Para encontrar las arquitecturas más apropiadas se experimentaron con varias redes, se analizó su tiempo de entrenamiento y su desempeño. Fueron seleccionadas aquellas que obtuvieron un mayor porcentaje de éxito en la clasificación, la red que contara con el menor número de capas y donde estas capas tuvieran el menor número de nodos posible.

En la etapa de pruebas, cada red mostró un porcentaje de aproximadamente el 90% de caracteres clasificados correctamente, enfatizando el hecho que el caracter debe estar segmentado perfectamente, con un total de 142 dígitos, y 89 letras.

Ademas de las redes mencionadas, se utilizó otro conjunto también de cuatro redes que reciben los datos del esquema secundario de descripción, la información de los hoyos que puede tener un caracter. Dos redes para los dígitos y dos para las letras. En cada caso una es para el reconocimiento de caracteres con un sólo hoyo, y otra para clasificar caracteres con dos hoyos. Estas redes son multiperceptrones, que fueron seleccionados ya que mostraron mayor rapidez en el entrenamiento que la red de retropropagación, así como un buen trabajo de clasificación.

Se analizó el desempeño de estas redes con un conjunto de 60 dígitos, de los cuales solamente se clasificaron mal o no se reconocieron 8, siendo el porcentaje de éxito de aproximadamente 87%.

### **6.4. Proceso automático de reconocimiento.**

Durante las primeras pruebas realizadas al sistema, la segmentación de caracteres se realizó calculando las posiciones de los caracteres a partir del caracter encontrado durante el

de la mala clasificación en la mayoría de las matriculas es debido a una mala segmentación. En la figura 32 se ejemplifica todo el proceso de reconocimiento.

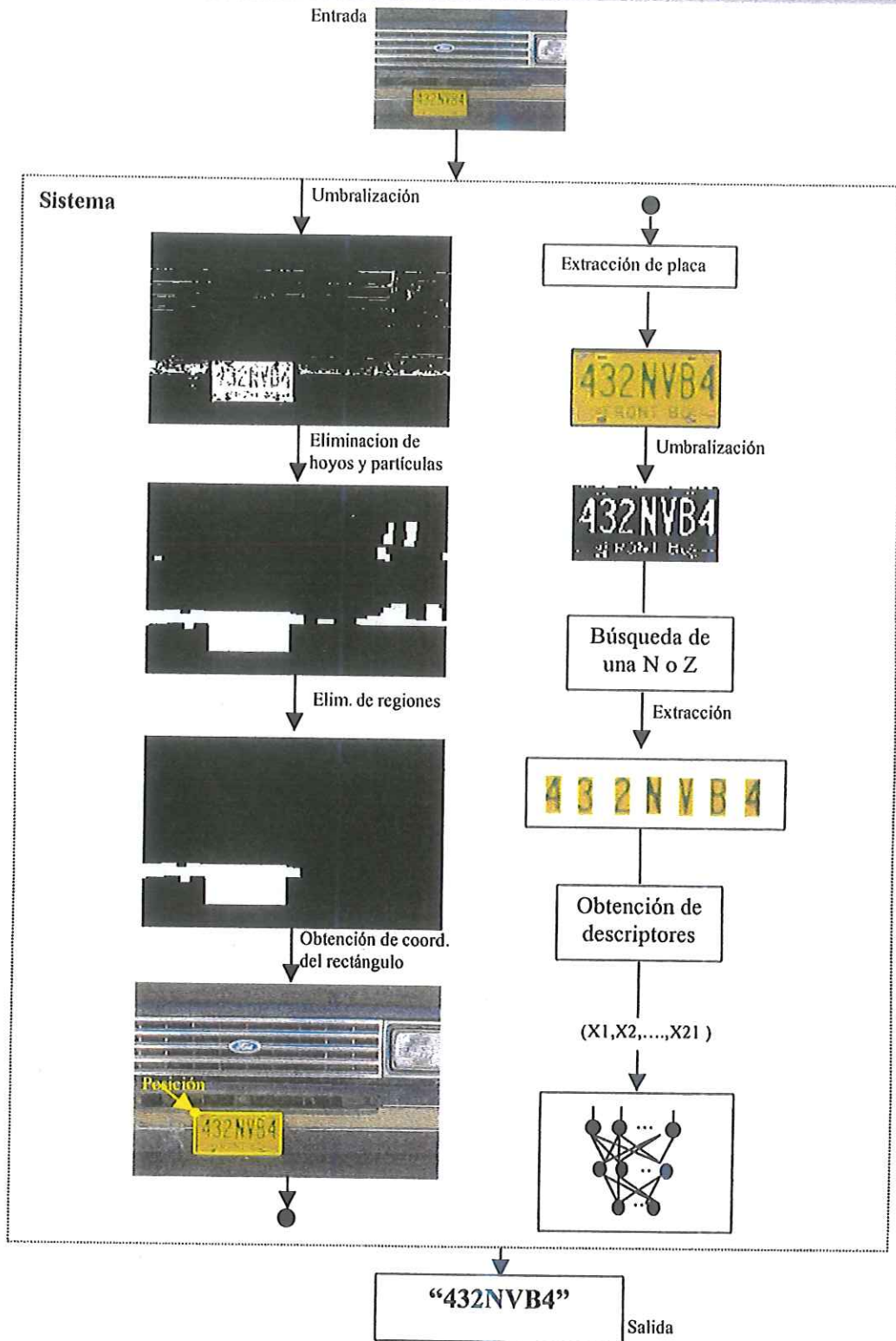


























Fig 32. Proceso de reconocimiento automático.

*Las Redes Neuronales Artificiales en el Reconocimiento de Imágenes*

Placa	Matrícula reconocida	Placa	Matrícula reconocida
	434 BVB 4		578 NVB 4
	ZKP 715 3		ZKW 5212
	ZKG 74# #		499 NUS 9
	25# NKG 6		438 NUE 9
	### ### #		518 NVN 5
	497 NTN 6		ZKP 732 1
	### ### #		363 NVK 4
	386 #UJ 2		209 NUW 3
	ZKU 751 3		339 NTY 7
	276 N#J 1		129 NUL 2
	788 #V# 8		858 NVD 7
	### ### #		ZKX 490 4














	### ## #		794 NVJ 3
	668 NVM 2		900 NUH 5
	ZKL 444 4		ZKT 483 9
	166 NVB 3		ZKK 407 8
	ZKS 313 5		222 NUJ 8
	029 NUY 4		229 NUW 6
	386 NUJ 2		

Tabla VI. Placas segmentadas y matrículas correspondientes reconocidas

## 6.5. Conclusiones

Realizando un análisis de todo el proceso de investigación realizado durante este trabajo se puede concluir lo siguiente:

En el reconocimiento de caracteres impresos por máquinas, las redes neuronales artificiales son una buena opción a utilizar, debido a su marcada habilidad para derivar de datos complicados o imprecisos, además de su capacidad de extraer patrones y detectar tramas que son muy difíciles de apreciar por otras técnicas computacionales.

Entre mas información participe en el procesamiento mas recursos de decisión podrán considerarse, en nuestro caso de estudio la información de color fue muy importante en la etapa de segmentación, así como la información acerca del área de la lámina de la placa.

Es de importancia la utilización de la información que se puede obtener del dominio del problema. En nuestro caso la información obtenida acerca de las características de las matrículas fue de gran ayuda tanto en la segmentación (el saber que existen dos tipos matrículas fronterizas, que en cada matrícula aparece siempre una N o una Z) como en el reconocimiento (la información acerca de las posiciones de conjuntos de letras y conjuntos de números), facilitando dichas tareas.

Es importante el utilizar un conjunto de varias redes neuronales en el módulo de reconocimiento para agilizar el trabajo si las figuras a reconocer son muchas. En el caso del reconocimiento de caracteres, existen caracteres con forma similar, lo que dificulta su distinción, afectando tanto la etapa de entrenamiento de la red como su etapa de implementación.

La implementación de dos conjuntos (o más) de redes neuronales que utilicen diferente información para el reconocimiento es de gran utilidad para el incremento de la tasa de éxito, si cierto conjunto de redes no reconocen un caracter, es probable que otro conjunto lo haga.

Las bibliotecas de procesamiento de imágenes Matrox Imaging Library son una herramienta muy poderosa, ya que cuentan con una gran cantidad de rutinas para llevar a cabo tareas tales como la captura, despliegue y manipulación de imágenes (filtrado, binarización, selección de regiones, operaciones morfológicas, detección de bordes, etc.).

## **6.5. Aportaciones y trabajo futuro.**

En este trabajo se presentaron los módulos principales de un sistema de reconocimiento automático de matrículas o LPR (por sus siglas en inglés). A partir de lo realizado en nuestro trabajo de investigación se pueden desarrollar diferentes aplicaciones que impliquen ésta tarea.

Los programas de cómputo desarrollados son el “middleware” de un sistema LPR completo, que se encuentra entre el módulo de adquisición y la aplicación. El sistema actual carece de módulo de adquisición; una cámara y una tarjeta de adquisición conectadas a la computadora donde se ejecute el sistema de reconocimiento, así como la interfaz que comunique a estos dos sistemas. El desarrollo de la interfaz de manipulación del sistema de adquisición puede ser la siguiente etapa del presente trabajo.

Otra etapa a desarrollar es una aplicación en particular del sistema LPR, donde se ve involucrada un manejador de base de datos. Un objetivo en particular de un sistema de este tipo es, por ejemplo, detectar autos robados del estado. Se puede instalar el sistema en una caseta de cobro de Caminos y Puentes Federales, y donde al pagar los automovilistas la cuota se les tome una foto frontal de su vehículo, la cual será la entrada a nuestro sistema. Una vez reconocida la matrícula, ésta puede ser enviada (via Internet), a la computadora central del departamento vehicular, donde se cuenta con una base de datos de todos los vehículos del estado, encuentre los datos del auto al que le corresponde la matrícula, los envíe a la computadora que los solicitó, y ésta los despliegue, para saber la situación del auto (robado, ilegal, etc.).

## Referencias

- [Brown, 1998] Brown, Eric W. 1998. *Optical Character Recognition*. Sitio en internet.  
URL: <http://www.ccs.neu.edu/home/feneric/charrecnn.html>
- [Castleman, 1979] Castleman, Kenneth R. 1979. *Digital Image Processing*.  
Prentice Hall, Englewood Cliffs, New Jersey.
- [Churchland y Sejnowski, 1992] Churchland, Patricia S. y Terrence J. Sejnowski. 1992. *The Computational Brain*. The MIT Press. Cambridge, Massachusetts.
- [Cummins y Dellarosa, 2000] Cummins, Robert y Denise Dellarosa Cummins. 2000. *Minds, Brains and Computers, The Foundations of Cognitive Science*. Ed. Blackwell Publishers.
- [González y Woods, 1992] González, Rafael C. y Richard E. Woods. 1992. *Digital Image Processing*. Addison-Wesley Publishing Company, Inc. Estados Unidos.
- [Hall, 1979] Hall, Ernest L., 1979. *Computer Image Processing and Recognition*. Academic Press. New York.
- [Hilera y Martínez, 1995] Hilera, José R. y Víctor J. Martínez. 1995. *Redes Neuronales Artificiales, Fundamentos, Modelos y Aplicaciones*. Addison-Wesley Iberoamericana, S. A. Wilmington, Delaware, E.U.A.
- [Jahne, 1997] Jahne, Bernd. 1997. *Digital Image Processing: Concepts, Algorithms and Scientific Applications*. 4a Edición. Ed. Springer. Alemania
- [Kartalopoulos, 1996] Kartalopoulos, Stamatios V. 1996. *Understanding Neural Networks and Fuzzy Logic; Basic Concepts and Applications*. IEEE Press. Piscataway, New Jersey.
- [Kohonen, 1988] Kohonen, Teuvo. 1988. *An Introduction to Neural Computing*. Neural Networks, Vol 1, No 1. p. 4. Pergamont Press.

- [Maravall, 1994] Maravall, Darío. 1994. *Renocimiento de Formas y Visión Artificial*. Addison-Wesley Iberoamericana, S. A. Wilmington, Delaware, E.U.A.
- [Nelson, 1998] Nelson, Lee J. 1998. *License Plate Recognition Systems*. Sitio en Internet. URL: <http://www.ettm.com/news/lpr.html>
- [Papoulis, 1965] Papoulis, A. 1965. *Probability, Random Variables and Stochastic Processes*. McGraw Hill. Nueva York.
- [Rosenzweig y Leiman, 1989] Rosenzweig, Mark R. y Arnold I. Leiman. 1989. *Psicología Fisiológica*. McGraw Hill. España.
- [Sinha, 1999] Sinha, Anshu. 1999. *An Improved Recognition Module for the Identification of Handwritten Digits*. Tesis de Licenciatura, MIT. Sitio en Internet. URL: <http://scanner-group.mit.edu/htdocs/thesis/anshuthesis.htm>