

**UNIVERSIDAD NACIONAL DE INGENIERIA**

**Facultad de Ciencias**

**Escuela Profesional de Ingeniería Física**



**INFORME DE SUFICIENCIA**

**PARA OPTAR EL TITULO PROFESIONAL DE  
INGENIERO FÍSICO**

**DESARROLLO DE UN SISTEMA BASADO EN LA VISIÓN ARTIFICIAL PARA EL  
RECONOCIMIENTO DE PLACAS VEHICULARES**

**Presentado por:**

**Bach. José Antonio Suárez Pino**

**Asesor:**

**Mg. Germán Yuri Comina Bellido**

**LIMA – PERÚ**

**2011**



## INDICE

INDICE	3
LISTA DE FIGURAS	5
TÍTULO	7
RESUMEN	8
1. INTRODUCCIÓN	9
1.1. Introducción	9
1.2. Objetivos	11
1.2.1. Objetivo general	11
1.2.2. Objetivos específicos	11
2. FUNDAMENTO TEÓRICO	12
2.1 Sistema de adquisición de imágenes:	12
2.1.1 La Cámara	12
2.1.2 Sensores tipo CCD y CMOS	13
2.1.2.1 Tecnología CCD	14
2.1.2.2 Fundamentos de la tecnología CCD	14
2.1.2.3 Proceso de lectura en la tecnología CCD	16
2.1.2.4 Tecnología CMOS	18
2.1.2.5 Tipos de detectores CMOS	19
2.1.2.6 Proceso de lectura en la tecnología CMOS	20
2.1.2.7 Factor de relleno	20
2.1.2.8 Ruido térmico	20
2.1.2.9 Eficiencia cuántica	20
2.1.2.10 El efecto Fotoeléctrico	21
2.2 Teoría del Color	21
2.2.1 Filtro de Bayer	23
2.3 Representación Numérica De Las Imágenes Digitales	24
2.3.1.1 Imagen Binaria	26
2.3.1.2 Imagen en Escala de Grises	26
2.3.1.3 Imagen a Color	26

2.3.1.4 Imagen HSI	27
2.3.1.5 Conversión de Imagen RGB A HSI	28
2.3.1.6 Segmentación por Color	29
2.4 Segmentación	30
2.5 redes neuronales	30
2.5.1 El algoritmo <i>Back</i> Propagación (BP)	31
2.5.2 El proceso <i>Forward</i> Pass en el Algoritmo BP	32
2.5.3 Error de propagación en el Algoritmo BP	33
3. PROCEDIMIENTOS EXPERIMENTALES	35
3.1 Conversión de la Imagen RGB a HSI	35
3.2 Binarización de la Imagen HSI	38
3.3 Operaciones Morfológicas y detección de regiones candidata	39
3.4 Reconocimiento con los caracteres aplicando redes neuronales artificiales	44
3.4.1 Entrenamiento de la red neuronal	46
3.4.2 Ingreso de datos	46
3.4.3 Inicialización	46
3.4.4 Propagación hacia adelante	47
3.4.5 Propagación hacia atrás	47
3.4.6 Cálculo de error	47
3.4.7 Mostrar resultados	48
4. RESULTADOS EXPERIMENTALES	49
5. DISCUSIÓN	50
6. CONCLUSIONES	51
7. REFERENCIAS	52

## LISTA DE FIGURAS

Figura 1.1	Diagrama de bloque del reconocimiento de los caracteres de la placa de un automóvil.	10
Figura 2.1	Cuerpo de una cámara.	12
Figura 2.2	Ejemplos de sensores CCD, lineales y de dos dimensiones.	14
Figura. 2.3	Sección transversal de un capacitor buried channel usado en la fabricación de sensores CCD.	15
Figura 2.4	Arquitectura, proceso de lectura y registro en un sensor CCD.	17
Figura 2.5	Proceso de transferencia de carga de un pozo de potencial a otro dentro de un píxel.	17
Figura 2.6	Transferencia de información en el registro de lectura, esta se da horizontalmente.	18
Figura 2.7	Sensores CMOS de tipo pasivo (izquierda) y de tipo activo (derecha).	19
Figura 2.8	Partes del ojo humano.	22
Figura 2.9	Pigmentos y colores primarios y secundarios.	23
Figura 2.10	Esquema del Filtro de Bayer sobre un sensor de imagen.	24
Figura 2.11	Eje de coordenada utilizado y notación para la referenciar un elemento de la imagen.	25
Figura 2.1.	Espacio del color RGB.	26
Figura 2.13	El espacio de color HSI.	29
Figura 2.14	Proceso de segmentación.	30
Figura 2.15	Estructura de Red Neuronal.	32
Figura 3.1	Fotografía de automóvil.	35
Figura 3.2	Fotografía de automóvil recortada.	36
Figura 3.3	Diagrama de flujo que convierte la imagen RGB a HSI.	37
Figura 3.4	Imagen en componente HSI.	37
Figura 3.5	Diagrama de flujo que binariza la imagen HSI .	38
Figura 3.6	Imagen Binarizada.	39
Figura 3.7	Remover ruido.	39
Figura 3.8	Imagen binarizada “sin ruido”.	40

Figura 3.9	Diagrama de Flujo que detecta áreas probables.	41
Figura 3.10	Muestra los resultados de las regiones candidatas.	42
Figura 3.11	(a) Imagen dilatada (b) Imagen erosionada (c) Imagen llena agujeros con píxeles (d)	42
Figura 3.12	Muestra la región donde se encuentra la placa.	43
Figura 3.13	La región de la placa ubicada en colores RGB.	43
Figura 3.14	Segmentación de placa.	44
Figura 3.15	Algoritmo de la red neuronal.	45
Figura 3.16	Base de datos de patrones.	46
Figura 3.17	Muestra de la fotografías.	48
Figura 4.1	Tabla de resultado.	49
Figura 4.2	Tabla de resultado de errores por etapa.	49

TITULO

**DESARROLLO DE UN SISTEMA BASADO EN LA VISIÓN ARTIFICIAL PARA EL  
RECONOCIMIENTO DE PLACAS VEHICULARES**

## RESUMEN

El presente trabajo describe el desarrollo del sistema de reconocimiento de placas vehiculares basado en visión artificial. En primer lugar, la placa es detectada dentro de la imagen digital usando las características de ancho, altura y área de los objetos que hay en la imagen. Después de ubicar la placa, los caracteres son extraídos de la imagen uno a uno. Finalmente cada carácter es reconocido usando el algoritmo de redes neuronales artificial. Las pruebas de reconocimiento se realizaron tomando 50 fotografías de placas de automóviles, encontrándose que el sistema tiene una eficiencia del 70%.

**Palabras claves** Localización de placa, Segmentación de imagen, red neuronal artificial.

# 1. INTRODUCCIÓN

## 1.1 Introducción

El presente trabajo es un estudio y aplicación de variadas técnicas de procesamiento digital de imágenes orientado hacia el área de reconocimiento de caracteres, utilizando para ello un sistema conformado por una cámara digital, una computadora y el sistema de procesamiento digital de imágenes desarrollado.

La necesidad de un sistema de reconocimiento automático de placa de automóviles, ha sido cada vez mayor. En concreto, los sistemas de reconocimiento en combinación con los entes reguladores de transporte, en diferentes áreas tales como la aplicación de las normas de tránsito (por ejemplo, velocidad de aplicación límite) y usos comerciales, tal como pago automático de peajes.

Un sistema de reconocimiento práctico generalmente contiene las siguientes fases. En primer lugar, tiene que ser capaz de localizar el número de placa, incluso en entornos complicados, por lo que pre-procesamiento de la imagen de entrada se requiere para mejorar el rendimiento en las etapas posteriores (segmentación de carácter y reconocimiento de caracteres). La segunda etapa se refiere a la localización exitosa de la región de los caracteres. Por otra parte, un paso importante en esta fase es la segmentación de cada carácter, para aplicar la fase de reconocimiento. La última etapa del sistema de reconocimiento tiene como objetivo reconocer con éxito los caracteres de las placas.

En el proceso de reconocimiento de los caracteres se utiliza el algoritmo de redes neuronales

En la figura 1.1 se muestra un diagrama de bloques en el que se ilustra el esquema básico de todo el procesamiento de la imagen de entrada hasta la obtención de los caracteres. En cada etapa hubo diversos problemas que se fueron superando con el curso de la investigación y el desarrollo de los algoritmos como veremos a continuación.

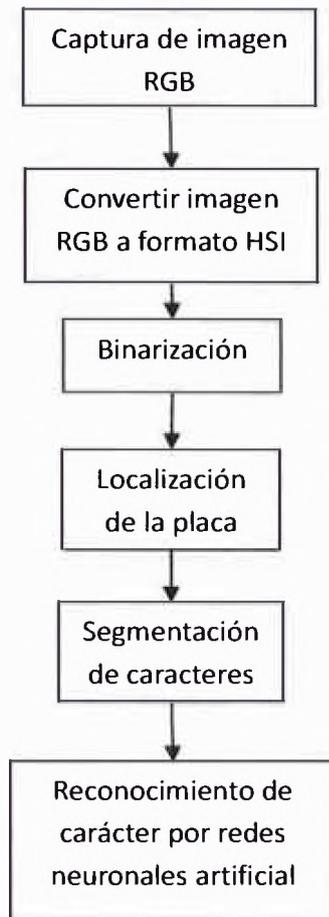


Figura 1.1 Diagrama de bloque del reconocimiento de los caracteres de la placa de un automóvil.

## **1.2 Objetivos**

### **1.2.1 Objetivo General:**

Desarrollar un sistema basado en la visión artificial para el reconocimiento de placas vehiculares.

### **1.2.2 Objetivos específicos:**

Desarrollar un algoritmo de ubicación de la placa en una imagen.

Desarrollar un algoritmo de segmentación de caracteres en la imagen donde se encuentra la placa.

Desarrollar un algoritmo de reconocimientos de caracteres a través de redes neuronales artificiales.

## 2. FUNDAMENTO TEÓRICO

### 2.1 Sistema de adquisición de imágenes:

#### 2.1.1 La Cámara

Una cámara fotográfica, de cine o de vídeo está compuesta de las siguientes partes principales: Diafragma, Objetivo, Obturador, Plano imagen:

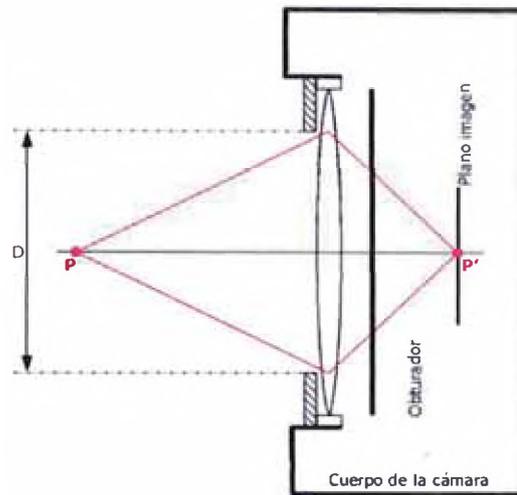


Figura 2.1 Cuerpo de una cámara.

El diafragma (o iris) es un sistema mecánico de láminas que forman un orificio circular por el que penetra la luz y cuyo tamaño puede ser modificado permitiendo controlar la cantidad de luz que entra en la cámara.

El objetivo es un conjunto de lentes por donde pasa la luz, procedente de un objeto, que ha entrado por el diafragma formando una imagen del objeto.

El obturador es un sistema electrónico que bloquea o permite el paso de la luz. Permite controlar el intervalo de tiempo durante el cual el plano imagen recibe la luz procedente del objeto.

El plano imagen es la película (fotográfica o de cine) o sensor donde se forma la imagen.

Enfocar un objeto consiste en desplazar el objetivo hacia atrás (objetos próximos) o hacia delante (objetos lejanos) hasta que la imagen del mismo aparezca nítida en el plano imagen donde está situada la película sensible (cámara fotográfica analógica o cámara de cine) o sensor digital de imagen.

### **2.1.2 Sensores digitales de imagen**

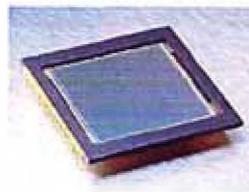
Existen dos tipos de tecnologías utilizadas para la fabricación de sensores de cámaras digitales. Se trata de los CCD (Charge Coupled Device) o CMOS (Complementary Metal Oxide Semiconductor). Ambos tipos de sensores están formados en su esencia por semiconductores de metal-óxido (MOS) y están distribuidos en forma de matriz.

Su función es la de acumular una carga eléctrica en cada una de las celdas de esta matriz. Estas celdas son los llamados píxeles. La carga eléctrica almacenada en cada píxel, dependerá en todo momento de la cantidad de luz que incida sobre el mismo. Cuanta más luz incida sobre el píxel, mayor será la carga que este adquiera.

Aunque en su esencia, los CCD y CMOS funcionan de una manera muy similar, hay algunas diferencias entre ambas tecnologías.

### 2.1.2.1 Tecnología CCD

Un sensor CCD es un circuito integrado que consiste de celdas sensibles a la luz (píxeles) que son capaces de producir una carga eléctrica proporcional a la cantidad de luz que reciben. Los píxeles están contruidos en un arreglo lineal o de dos dimensiones. Estos son fabricados usando tecnología MOS (metal oxide semiconductor) [2], donde cada píxel es con iderado como un capacitor MOS. En la figura 2.2 se pueden apreciar modelos de sensores CCD con distinta cantidad de píxeles.



Manufacturer: Kodak  
Type: area array  
Pixel count: 4096x4096  
Transfer method: full frame transfer



Manufacturer: Dalsa  
Type: area array  
Pixel count: 256x256  
Transfer method: frame transfer



Manufacturer: Kodak  
Type: linear array  
Pixel count: 5000  
Transfer method: full frame transfer

Figura 2.2 Ejemplos de sensores CCD, lineales y de dos dimensiones.

### 2.1.2.2 Fundamentos de la tecnología CCD

Cada píxel de un sensor CCD está basado en un capacitor de tipo MOS, de acuerdo a su estructura de fabricación estos son considerados como surface channel CCD y buried channel CCD. Por lo general los CCD son fabricados siguiendo la estructura buried channel debido a que presenta una baja eficiencia de la transferencia de cargas debido a irregularidades en la superficie de la interfase óxido - semiconductor. En la figura 2.3 se observa una sección transversal de un capacitor MOS con la estructura buried channel. Los dispositivos CCD con la estructura buried

channel son fabricados sobre sustratos de Silicio tipo p (aprox. de 300  $\mu\text{m}$  de espesor) con una capa de Silicio tipo n sobre su superficie (aprox. de 1  $\mu\text{m}$  de espesor), luego una capa de dióxido de silicio (aprox. De 0,1  $\mu\text{m}$  de espesor) es crecida sobre la región tipo n y finalmente se depositan electrodos metálicos (llamados gates) usando procesos de depósito químico de vapor (o CVD) [2, 3].

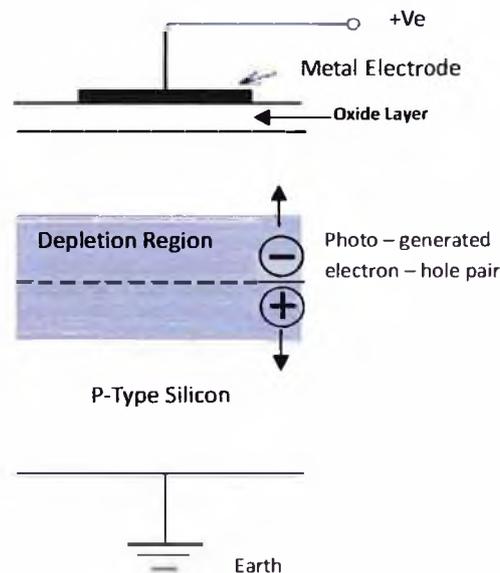


Figura 2.3 Sección transversal de un capacitor buried channel usado en la fabricación de sensores CCD.

El Silicio tipo p y tipo n generalmente son dopados con átomos de Galio y Fósforo respectivamente, en el de tipo p se tienen huecos extras mientras que en el de tipo n son los electrones que se encuentran en exceso en su estructura. Bajo la acción de un campo eléctrico se puede controlar el movimiento de las cargas positivas como de los electrones, esto es uno de los fundamentos principales en la operación de los sensores CCD.

Si se ponen en contacto un material tipo p y uno tipo n, se crea una unión p-n. Los electrones extra del material tipo n se difundirán sobre el material tipo p y llenarán algunos de los huecos extra del material tipo p. Esta difusión y recombinación de pares hueco-electrón hace que el material tipo n se cargue positivamente y el material tipo p se cargue negativamente. A medida

que la carga de estos materiales se incrementa, se genera un campo eléctrico alrededor de la superficie de contacto de los materiales tipo n y tipo p. Este campo eléctrico frenará el proceso de difusión, cuando la difusión termine, no quedarán más electrones o huecos alrededor de la unión, en esta zona donde los electrones y huecos se recombinan se forma un pozo de potencial, llamada zona de deflexión. Fuera de esta zona, aún existen electrones extra en el material tipo n y huecos extra en el material tipo p. esta zona de deflexión es muy importante para la creación dispositivos electrónicos. Aplicando un voltaje sobre la zona de deflexión, podemos incrementar o disminuir el campo eléctrico sobre esta zona. Si el campo eléctrico es disminuido por un voltaje aplicado en reversa (reverse sesgos voltage), la zona de deflexión disminuye y facilita el paso de alguna corriente aplicada a través de los dos materiales.

La importancia de aplicar un voltaje sobre la región de deflexión es que nos permite controlar el paso de una corriente aplicada a través de cualquier material p-n. Cuando se aplica un voltaje en reversa sobre la unión p-n, habrá una corriente muy pequeña, la cual para efectos prácticos puede considerarse como cero. Este caso corresponde al estado off.

Cuando esta unión es expuesta a la luz, los fotones con suficiente energía que llegan a la superficie de la unión liberarán electrones, al mismo tiempo se genera un hueco por electrón. Así los fotone incidentes crean pares hueco-electrón.

Aplicando un voltaje en reversa, los electrones son capturados por el pozo potencial generado. A mayor intensidad de luz sobre cada píxel, el número de electrones capturados por el pozo de potencial será también mayor. Así las diferencias en intensidad de la luz incidente serán registradas por diferencias en el número de electrones en cada pozo de potencial.

### **2.1.2.3 Proceso de lectura en la tecnología CCD**

El proceso de lectura está dividido en dos secciones. La primera sección involucra mover las cargas asociadas a los píxeles a través de la superficie del arreglo que forma el sensor, las cargas son transferidas verticalmente de una unidad recolectora de cargas a otra. Este método permite una buena eficiencia en la transferencia de la información, pero es necesario un alto consumo de energía, este alto consumo de energía permite mantener una buena calidad de imagen y limitar el ruido electrónico. En la segunda sección, las cargas son leídas horizontalmente por un registro

para luego digitalizar esta información. En la figura 2.4 se observa el proceso de lectura y registro en la tecnología CCD.

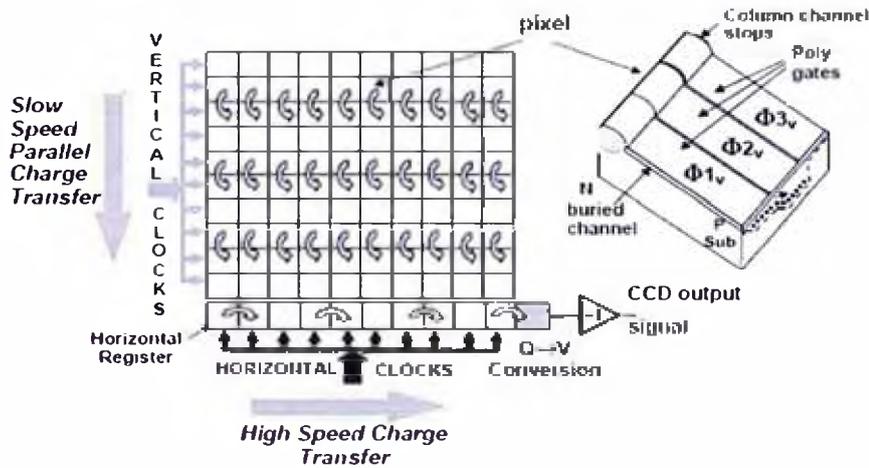


Figura 2.4. Arquitectura, proceso de lectura y registro en un sensor CCD.

En el proceso de transferencia de carga, cada píxel es dividido en distintas áreas conocidas como fases. Por lo general los sensores son diseñados con píxeles divididos en tres fases, en cada fase es posible la formación de un pozo de potencial. Para transferir la información, la carga debe transitar por las 3 fases de cada píxel. Esto se logra cambiando secuencialmente el voltaje aplicado a cada fase del píxel. En la figura 2.5 se muestra una representación del proceso de transferencia de cargas entre píxeles.

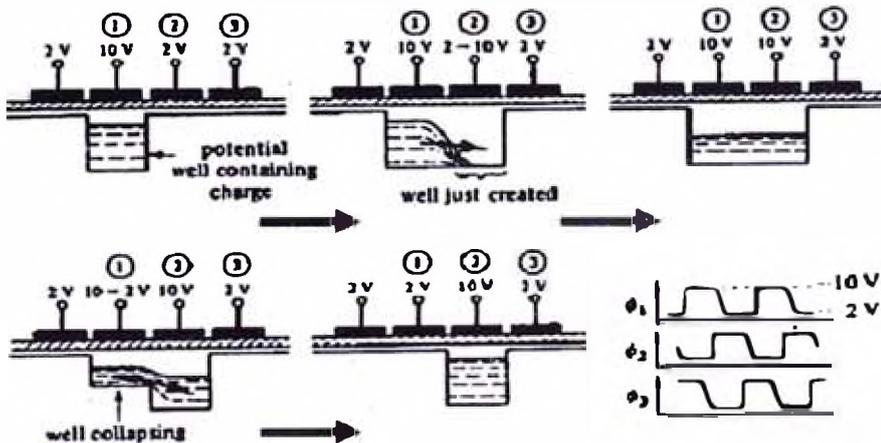


Figura 2.5. Proceso de transferencia de carga de un pozo de potencial a otro dentro de un píxel.

Primero las fases 2 y 3, cercanos al portador de la carga 1, se mantienen a un voltaje bajo (2 voltios). Luego el potencial de la fase 2 es elevado (10 voltios) sobre un intervalo finito de tiempo, de modo que ahora se tiene también un pozo de potencial en esta fase y la carga se distribuye entre el pozo 1 y 2. La carga es transferida en su totalidad a la fase 2 llevando la fase 1 a un voltaje bajo. De la misma forma se transfiere la carga a la fase 3. Este proceso es repetido en fracciones de segundo para transferir toda la carga hacia una unidad de registro.

La segunda parte del proceso de lectura ocurre luego de que la carga ha sido transferida por cada columna de píxeles. Existe una fila adicional de píxeles llamada registro de lectura. En este momento, las cargas en el registro de lectura son transferidas horizontalmente, para luego ser enviadas a un convertidor A/D antes de ser almacenadas en una memoria. Este proceso es resumido en la figura 2.6.

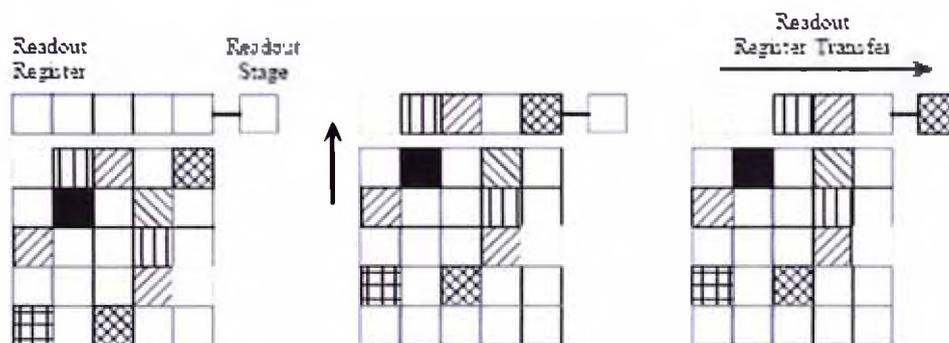


Figura 2.6 Transferencia de información en el registro de lectura, esta se da horizontalmente.

#### 2.1.2.4 Tecnología CMOS

Los sensores CMOS así como los de tipo CCD, están formados por un arreglo de elementos sensibles a la luz, donde cada uno de ellos es capaz de producir una señal eléctrica proporcional a la luz incidente. Sin embargo, se diferencian en el proceso para producir esta señal eléctrica. Mientras que un píxel en la tecnología CCD está formado por una unión p-n a un voltaje aplicado en reversa el cual crea un pozo de potencial, cada píxel en la tecnología CMOS emplea un fotodiodo, un capacitor y hasta tres transistores. Antes de ser expuestos los fotodiodos a la luz, los capacitores son cargados a un voltaje conocido. La descarga de los capacitores empieza

cuando los fotones inciden sobre los fotodiodos, la velocidad de descarga es proporcional a la intensidad de la luz sobre los fotodiodos. La carga restante en los capacitores es leída y digitalizada. Los transistores tienen la función de amplificar, transferir y digitalizar esta carga. De esta forma se puede leer cada píxel individualmente. Al tener esta tecnología mayor integración de circuitos, no todas las regiones del píxel serán sensibles a la luz, por lo que los sensores CMOS no son tan sensibles como los de tipo CCD. Los sensores CMOS presentan también mayor ruido eléctrico.

### 2.1.2.5 Tipos de detectores CMOS

Por lo general los sensores CMOS son fabricados en dos estructuras: estructura de píxel pasivo y estructura de píxel activo. Los dispositivos de píxel pasivo tienen amplificadores de carga al final de cada columna de píxeles y cada píxel tiene un sólo transistor (además de un fotodiodo y un capacitor). Este transistor es usado como una compuerta para la carga y permite la transferencia del contenido de cada capacitor al amplificador de carga. En los dispositivos de píxel activo se tiene un amplificador de carga por cada píxel. Los tipos de sensores CMOS son mostrados en la figura 2.7

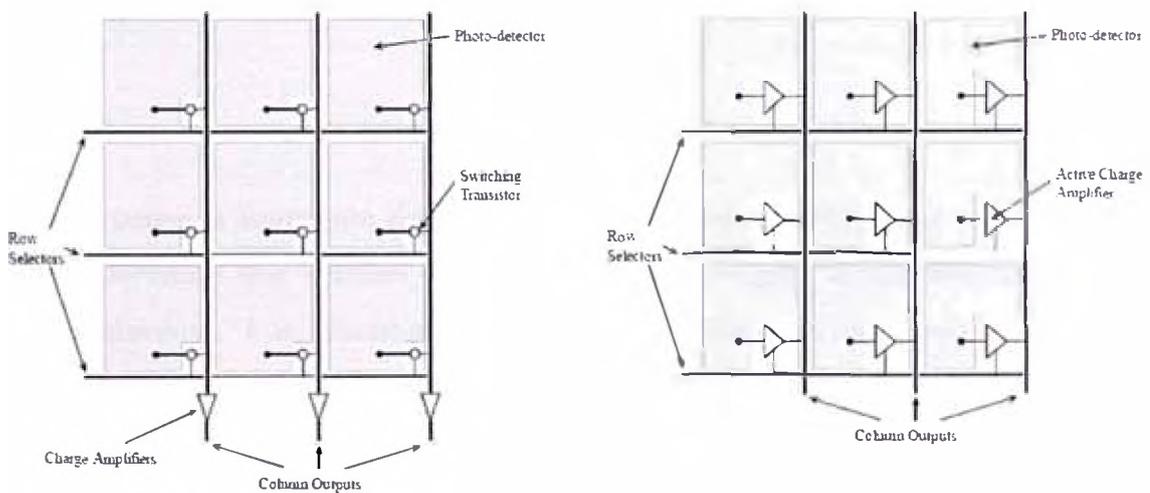


Figura 2.7 Sensores CMOS de tipo pasivo (izquierda) y de tipo activo (derecha).

### **2.1.2.6 Proceso de lectura en la tecnología CMOS**

Los sensores CMOS, tanto el de tipo pasivo como activo, usan la misma técnica para la lectura de la imagen fuera del arreglo de píxeles. Cada fila del arreglo de píxeles es leída secuencialmente, mientras que en las salidas de las columnas la lectura se da en paralelo permitiendo una lectura de cada píxel individual. De esta forma se puede realizar la lectura y formación un área seleccionada de una imagen, esto permite tener velocidades muy altas en la lectura de la información.

Para poder cuantificar la eficiencia y calidad de los sensores tanto de tipo CCD como los de tipo CMOS es necesario definir algunos factores como: factor de relleno [4], ruido térmico [4,5], eficiencia cuántica [4].

### **2.1.2.7 Factor de relleno**

Este factor indica el porcentaje de cada píxel que es sensible a la luz, en la práctica este factor es menor al 100%. En el caso de la tecnología CMOS, parte del área de cada píxel es usada para integrar su propia electrónica. Estos componentes ocupan espacio entre cada píxel y no son sensibles a la luz. El efecto neto de reducir el factor de relleno es reducir la sensibilidad del sensor.

### **2.1.2.8 Ruido térmico**

Puede ser definido como la formación de cargas indeseadas que se acumulan en los sensores debido a procesos térmicos que ocurren cuando un dispositivo opera a una temperatura por encima del cero absoluto. Los electrones generados por este tipo de ruido pueden ser recolectados por los pozos de potencial generados en los sensores CCD y aparecerán como señales indeseadas en la salida del dispositivo.

### **2.1.2.9 Eficiencia cuántica**

La eficiencia cuántica es la medida de la eficiencia con la cual los fotones incidentes son detectados. Algunos de estos fotones incidentes pueden no ser absorbidos debido a reflexiones o pueden ser absorbidos en regiones donde los electrones no pueden ser recolectados. La eficiencia

cuántica está definida como la relación entre el número de pares hueco-electrón generados y el número de fotones incidentes.

### 2.1.2.10 El efecto Fotoeléctrico

En el efecto fotoeléctrico, electrones son emitidos desde un material como consecuencia de la absorción de energía proveniente de radiación electromagnética de muy corta longitud de onda, tal como la luz visible o ultra-Violeta (UV). A los electrones emitidos de esta forma se les refiere como “fotoselectrones” [6]

Einstein en 1905, encontró una explicación, consideró que la energía contenida en un fotón estaba relacionada con la frecuencia de la luz por la siguiente relación:

$$E = hf \quad \dots(2.1)$$

Donde  $h$  es una constante universal llamada constante de Planck. El valor numérico de esta constante, con la precisión actual, es

$$h = 6,6260755(40) \times 10^{-34} \text{ J} \cdot \text{s}$$

$f$  : frecuencia de oscilación de la radiación electromagnética

Al llegar un fotón a la superficie, es absorbido por un electrón, el electrón recibe toda la energía del fotón o ninguno en absoluto. Si esta energía es mayor que la función que la función trabajo  $\phi$ , el electrón puede escapar de la superficie.

La función trabajo  $\phi$  es la energía mínima necesaria para desprender un electrón de la superficie.

Einstein aplicó entonces el principio de conservación de la energía y propuso que la energía cinética máxima del fotoselectrón es:  $K_{m\acute{a}x} = \frac{1}{2}mv_{m\acute{a}x}^2 = hf - \phi$

## 2.2 Teoría del Color

Al igual que una cámara, un ojo normal enfoca la luz y produce una imagen nítida. Sin embargo, los mecanismos mediante los cuales el ojo controla la cantidad de la luz admitida y los ajustes para producir imágenes enfocadas correctamente son mucho más complejos, intrincados y efectivos que los correspondientes incluso a la cámara más avanzada. En todos aspectos el ojo es una maravilla fisiológica [7]

En la figura 2.8 se presenta las partes esenciales del ojo.

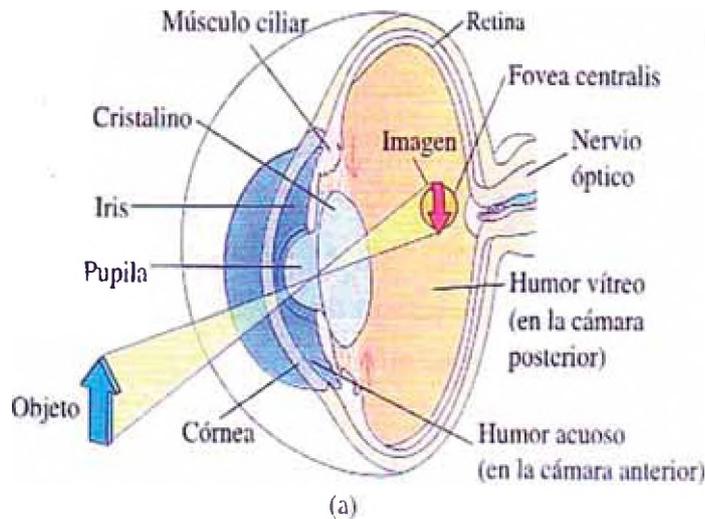


Figura 2.8 Partes del ojo humano.

El ojo es de forma esférica y tiene aproximadamente 2,5 cm de diámetro. La luz entra al ojo a través de una membrana transparente denominada córnea, detrás de la cual hay un líquido claro (el humor acuoso), una abertura variable (la pupila, que es una abertura en el iris) y el lente cristalino. La mayor parte de la refracción ocurre en la superficie exterior al ojo, donde la córnea está cubierta con una película de lágrimas. Hasta cierto punto ocurre poca refracción en el lente cristalino, ya que el humor acuoso en contacto con el lente tiene un índice de refracción promedio cercano al del lente, cuyo índice promedio es 1,437. El iris, el cual es la parte de color del ojo, es un diafragma muscular que controla el tamaño de la pupila. El iris regula la cantidad de luz que entra al ojo al dilatar la pupila en condiciones de luz baja y contraerla en condiciones de luz alta.

El sistema córnea – lente enfoca la luz hacia la parte posterior del ojo llamada retina, la cual está compuesta por millones de receptores sensibles llamados bastoncitos y conos, actúan como una serie de fotoceldas en miniatura, las cuales, cuando son estimulados por la luz, envían impulsos eléctricos al cerebro a través del nervio óptico, donde se percibe una imagen. Mediante este proceso se observa una imagen nítida de un objeto cuando la imagen llega a la retina.

El ojo enfoca a un objeto determinado variando la forma del lente cristalino plegable mediante un asombroso proceso conocido como adaptación. Un importante componente en la adaptación es el músculo ciliar, el cual está situado en un círculo alrededor del anillo del lente.

En el ojo humano hay tres tipos de conos, sensibles a luz de color azul, rojo y verde respectivamente. Cada uno de ellos absorbe la radiación de una determinada porción del espectro. Mediante las diferentes intensidades de las señales producidas por los 3 tipos de conos, podemos distinguir todos los colores que forman el espectro de luz visible.

Existen de unos 6 a 7 millones de conos en el ojo humano. El 65% de los conos son sensibles a la luz roja, 33% son sensibles a la luz verde y un 2% son sensibles a la luz azul (los conos que perciben azul son más sensibles). Debido a esta característica de absorción del ojo humano, los colores se observan debido a una combinación de colores, lo llamado color primario, rojo (R, Red), verde (G, Green) y azul (B, Blue) [8]

Los colores que observamos en los objetos es producto de la propiedad de reflexión del objeto. La suma de los colores primarios producen los colores secundarios: magenta (rojo más azul), cian (verde más azul) y amarillo (rojo más verde). Mezclando los tres colores primarios se obtiene blanco o sumando un color primario con su opuesto secundario, también, se obtiene blanco. Los pares color primario y su opuesto secundario son el rojo-cian, verde-magenta y azul-amarillo.

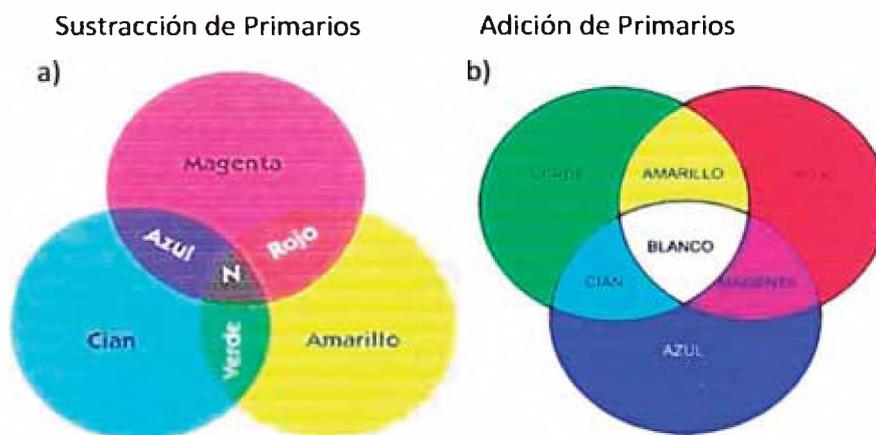


Figura 2.9 Pigmentos y colores primarios y secundarios.

### 2.2.1 Filtro de Bayer

El filtro de Bayer es una malla cuadrículada de filtros rojos, verdes y azul que se sitúa sobre un sensor digital de imagen (CCD) para hacer llegar a cada fotodiodo una tonalidad de los distintos colores primarios. Interpolando las muestras de varios fotodiodos se obtiene un píxel de color

El filtro de Bayer está formado por un 50% de filtros verdes (534 nm – 545 nm), un 25% de rojos (564 nm – 580 nm) y un 25% de azules (420 – 440 nm). [9,10]

Como se puede apreciar en la figura 2.10, este filtro Bayer no tiene una proporción equivalente de cada uno de los tres colores. El hecho de que la presencia de píxeles verdes sea mayor ( $1/2G + 1/4B + 1/4R$ ) tiene una explicación muy sencilla: el ojo humano es más sensible a este color y las cámaras tratan de reproducir las imágenes tal y como nosotros las percibimos.

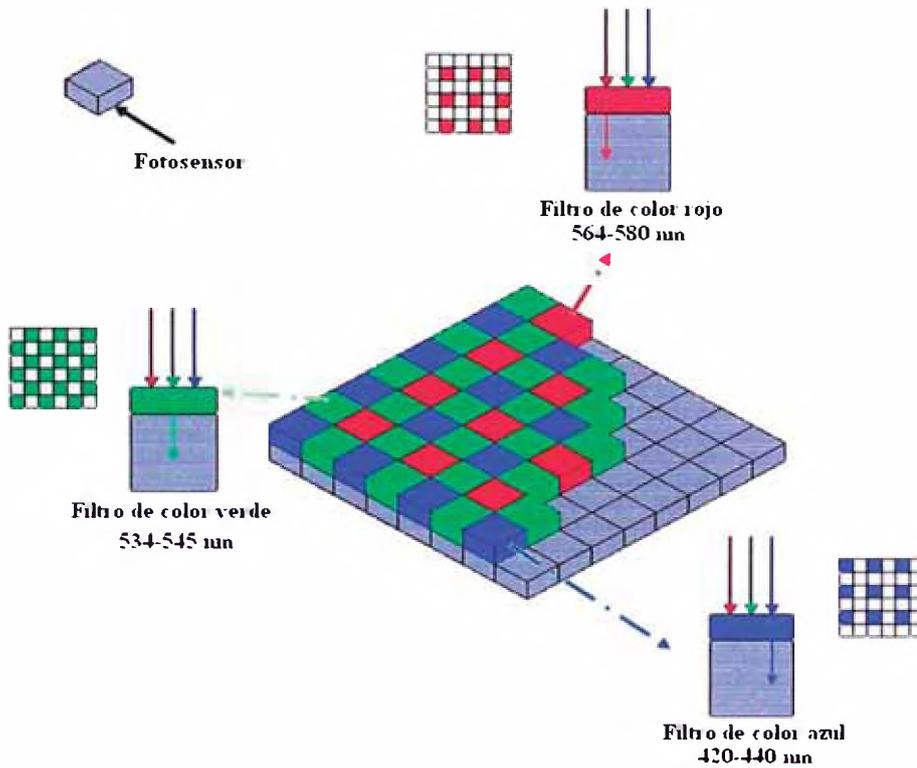


Figura 2.10 Esquema del Filtro de Bayer sobre un sensor de imagen.

### 2.3 Representación Numérica De Las Imágenes Digitales

Una imagen digital es un arreglo bidimensional de  $M \times N$  elementos, donde cada uno de estos elementos se denomina "píxel" (*picture element*) o elemento de la imagen. Estos elementos pueden ser de tipo binario, escala de grises o a color.

En el presente trabajo se utilizó la representación de las imágenes a color mediante los componentes rojo, verde y azul debido a que es la forma en la que las cámaras digitales utilizadas almacenan las imágenes; aunque también existen otras formas de asignar los componentes de color (por ejemplo cian, amarillo y magenta o componentes HSI)

El ordenamiento de los elementos que conforman la imagen se detalla en la figura 2.11 Una imagen de  $N$  píxeles de ancho por  $M$  píxeles de alto se representa por medio de una matriz de  $M$  filas por  $N$  columnas, y debido a que usamos imágenes a color cada uno de estos elementos esta compuesto por 3 componentes correspondientes a los colores rojo, verde y azul como se mencionó anteriormente. Cada uno de estos componentes puede tomar un valor entre 0 a 255 debido a que se emplean 8 bits por componente de color, correspondiendo el color negro cuando los tres componentes tienen un valor 0, y al color blanco cuando los tres componentes tienen el máximo valor de intensidad, 255.

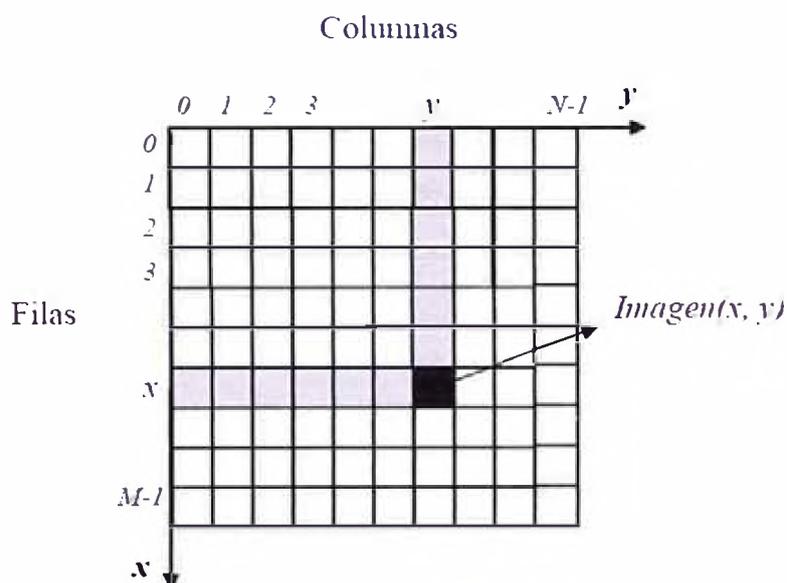


Figura 2.11 Eje de coordenada utilizado y notación para la referenciar un elemento de la imagen

Para la ubicación de un píxel o elemento específico dentro de la matriz se ha utilizado la notación:  $(x, y)$ , donde  $x$  corresponde al número de fila de arriba hacia abajo e  $y$  corresponde al número de columna de izquierda a derecha. De esta manera el píxel de la primera fila y primera columna tiene la posición  $(0,0)$ , y el píxel de la última fila y última columna tiene la ubicación  $(M-1, N-1)$ .

### 2.3.1 Tipos De Imagen Digital

**2.3.1.1 Imagen Binaria:** La intensidad que cada uno de los píxeles puede tomar solamente uno de dos valores: un valor bajo (0 en la mayoría de los casos) o un valor alto (1 o 255, según la cantidad de bits que se esta utilizando). Se asigna el color negro para un valor de 0, y el color blanco para el valor alto.

**2.3.1.2 Imagen en Escala de Grises:** Cada uno de los componentes de la imagen puede tomar un valor dentro de un rango de valores, de 0 a K. Ahora, si por ejemplo, estas componentes tienen el tamaño de 1 *byte* (8 *bits*), es decir que podrá tomar un valor entero entre 0 a 255, correspondiendo el 0 al color Negro y el 255 al color blanco o de mayor intensidad.

**2.3.1.3 Imagen a Color:** Existen diversas formas para representar el color de un píxel. En este caso se utiliza el formato RGB. Cada uno de los píxeles tiene un tamaño en memoria de 3 bytes (24 bits) debido a que esta compuesto por 3 componentes, cada uno correspondiente a los colores rojo, verde y azul, de tamaño de 1 byte cada uno (8 bits), por lo tanto cada componente de color puede tomar un valor de intensidad de 0 a 255, generando mas de 16 millones de colores los cuales se encuentran en el espacio de color mostrado en la figura 2.12, en donde cada color esta representado por un vector (R,G,B) cuyos componentes representan el valor de intensidad de los colores rojo, verde y azul respectivamente.

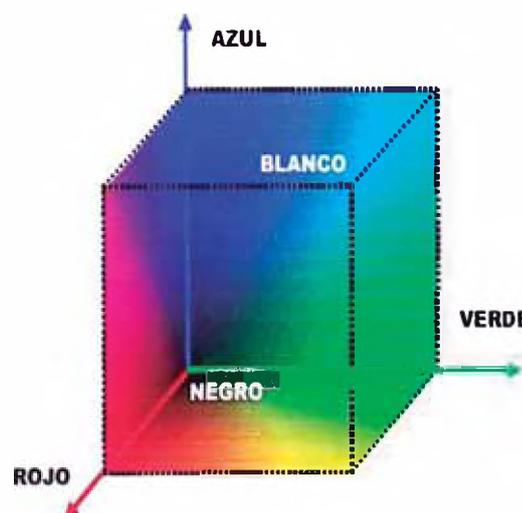


Figura 2.12 , Espacio del color RGB

La estructura utilizada para la representación numérica de las imágenes a color se muestra en la matriz (2.2). Aquí se ha utilizado matrices de  $M \times N$  elementos, de 3 componentes cada uno, correspondientes a la intensidad de los colores rojo, azul y verde. Los procesamientos se realizan de manera independiente por cada componente de color por lo que puede considerarse también a cada imagen como la combinación de 3 matrices independientes de  $M \times N$  elementos unidimensionales.

$$\text{Imagen RGB} = \begin{bmatrix} \begin{bmatrix} R(0,0) \\ G(0,0) \\ B(0,0) \end{bmatrix} & \begin{bmatrix} R(0,1) \\ G(0,1) \\ B(0,1) \end{bmatrix} & \dots & \begin{bmatrix} R(0,M-1) \\ G(0,M-1) \\ B(0,M-1) \end{bmatrix} \\ \begin{bmatrix} R(1,0) \\ G(1,0) \\ B(1,0) \end{bmatrix} & \begin{bmatrix} R(1,1) \\ G(1,1) \\ B(1,1) \end{bmatrix} & \dots & \dots \\ \vdots & \vdots & \dots & \vdots \\ \begin{bmatrix} R(N-1,0) \\ G(N-1,0) \\ B(N-1,0) \end{bmatrix} & \dots & \dots & \begin{bmatrix} R(N-1,M-1) \\ G(N-1,M-1) \\ B(N-1,M-1) \end{bmatrix} \end{bmatrix} \quad \dots(2.2)$$

Los diferentes colores se forman por la combinación de intensidades de cada uno de estos 3 componentes, de esta forma un píxel de color blanco se representa como (255, 255, 255), el color Negro: (0, 0, 0), el Rojo (255, 0, 0) y el Amarillo (255, 255, 0).

### 2.3.1.4 Imagen HSI

El tono es un atributo cromático que describe un color puro (amarillo puro, naranja puro, rojo puro), mientras que la saturación proporciona una medida del grado en que un color puro está diluido en la luz blanca. El modelo HSI debe su utilidad a dos hechos básicos. Primero, la

componente de la intensidad, **I**, está desacoplada de la información cromática contenida en la imagen. Segundo, las componentes de tono y saturación están íntimamente relacionadas con la forma en que los seres humanos percibimos el color. Estas características hacen que el modelo HSI sea una herramienta ideal para desarrollar algoritmos de procesamiento de imágenes basados en alguna de las propiedades de la percepción del color del sistema visual humano. Los ejemplos sobre la utilidad del modelo HSI van desde el diseño de sistemas automáticos para determinar el grado de maduración de frutas y otros vegetales, hasta sistemas para comparar muestras de color o inspeccionar la calidad de productos coloreados. En estas aplicaciones, y en otras similares, la clave consiste en que la operación del sistema se fundamenta en las propiedades del color de forma que una persona pueda utilizarlas para realizar la labor en cuestión.

### 2.3.1.5 Conversión de Imagen RGB a HSI

La transformada de (R, G, B) a (H, S, I) es:

$$H = \cos^{-1} \frac{\frac{1}{2} [(R - G) + (R - B)]}{\left[ (R - G)^2 + (R - B)(G - B) \right]^{\frac{1}{2}}} \quad \dots 2.3$$

$$S = 1 - \frac{3}{R + G + B} \min(R, G, B) \quad \dots 2.4$$

$$I = \frac{R + G + B}{3} \quad \dots 2.5$$

El espacio HSI es representado en la figura 2.13, donde H es la tonalidad y es representado por el ángulo H variando de 0° a 360°, la tonalidad H varía desde rojo cuando H= 0°, a través del amarillo H=60° , verde H= 120°, azul H= 240° y regresando al rojo que es 360°. La saturación corresponde al radio que varía de 0 a 1. Cuando S es 0, el color es un gris de intensidad I. Cuando S es 1, el color está en el límite de la base del cono superior. La intensidad I varía a lo largo del eje z cuando z=0 es negro y z=1 es blanco.

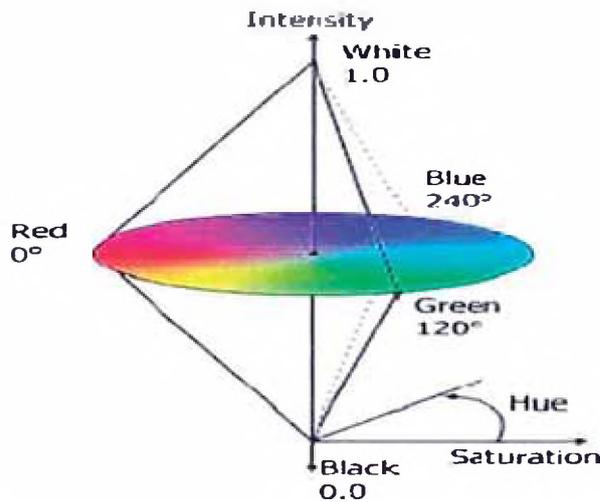


Figura 2.13 El espacio de color HSI

### 2.3.1.6 Segmentación por Color

Para hallar la ubicación de la placa en la imagen se plantea utilizar la conversión de imagen RGB a HSI de acuerdo a las ecuaciones 2.3 al 2.5

Debido a que la placa se encuentra en un fondo blanco, entonces en el proceso de binarización es probable que se pierda la región de interés (la placa) debido a la iluminación cambiante. Ahora el método que se propone es utilizar tres ecuaciones; para casos de pobre, normal y alta iluminación. [11]

Iluminación Normal 
$$b_{white(n)} = \begin{cases} 1, & \{S(x,y) \leq (\mu_s + \delta_s)\} \& \{I(x,y) \geq \mu_I + 0.25 \cdot \delta_I\} \\ 0, & \text{otro caso} \end{cases} \dots 2.6$$

Iluminación Baja 
$$b_{white(l)} = \begin{cases} 1, & \{S(x,y) \leq (\mu_s + \delta_s)\} \& \{I(x,y) \geq \mu_I - 0.33 \cdot \delta_I\} \\ 0, & \text{otro caso} \end{cases} \dots 2.7$$

Iluminación Alta 
$$b_{white(h)} = \begin{cases} 1, & \{S(x,y) \leq (\mu_s + \delta_s)\} \& \{I(x,y) \geq \mu_I + 0.5 \cdot \delta_I\} \\ 0, & \text{otro caso} \end{cases} \dots 2.8$$

## 2.4 Segmentación

Los algoritmos de segmentación de imágenes generalmente se basan en dos propiedades básicas de los niveles de gris de la imagen: Discontinuidad y Similitud. Dentro de la primera categoría se intenta dividir la imagen basándonos en los cambios bruscos en el nivel de gris. Las áreas de interés en esta categoría son la detección de puntos, líneas y bordes en la imagen. Las áreas dentro de la segunda categoría están basadas en las técnicas de umbrales, crecimiento de regiones, y técnica de división y fusión. La segmentación subdivide una imagen en regiones u objetos, dependiendo del problema a resolver, de manera que los objetos de interés se encuentren de forma aislada [8]. En primer lugar es necesario determinar cuales son los píxeles que pertenecen a los objetos de interés y cuales pertenecen al fondo de la imagen, por lo tanto cada píxel tomara solo uno de dos valores posibles, a este proceso se le denomina binarización, el cual se explica en la siguiente sección. La segmentación termina con el proceso de etiquetado, es decir con una numeración a cada objeto que permita su posterior identificación tal como se muestra en la figura 2.14

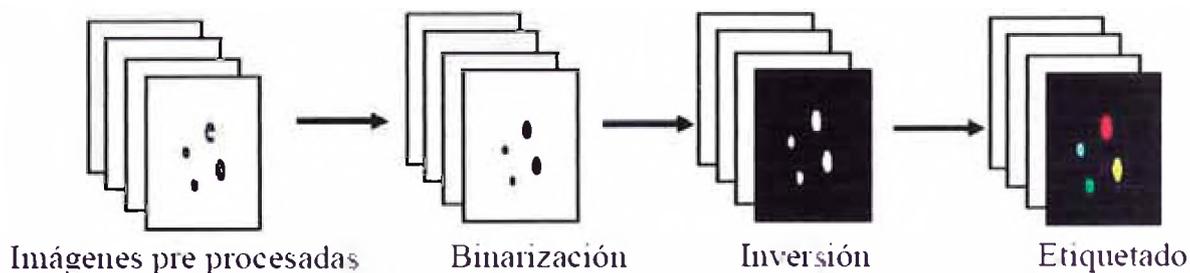


Figura 2.14 Proceso de segmentación

## 2.5 Redes Neuronales Artificiales

Una red neuronal artificial (ANN) es un esquema de computación distribuida inspirada en la estructura del sistema nervioso de los seres humanos. La arquitectura de una red neuronal se forma conectando múltiples procesadores elementales, siendo éste un sistema adaptivo que posee un algoritmo para ajustar sus pesos (parámetros libres) para alcanzar los requerimientos de desempeño del problema basado en muestras representativas.

Por lo tanto podemos señalar que una ANN es un sistema de computación distribuida caracterizada por:

- Un conjunto de unidades elementales, cada una de las cuales posee bajas capacidades de procesamiento.
- Una densa estructura interconectada usando enlaces ponderados.
- Parámetros libres que deben ser ajustados para satisfacer los requerimientos de desempeño.
- Un alto grado de paralelismo.

Es importante señalar que la propiedad más revelante de las redes neuronales artificiales es su capacidad de aprender a partir de un conjunto de patrones de entrenamiento, es decir, es capaz de encontrar un modelo que ajuste los datos. El proceso de aprendizaje también conocido como entrenamiento de la red puede ser supervisado o no supervisado.

El aprendizaje supervisado consiste en entrenar la red a partir de un conjunto de datos o patrones de entrenamiento compuesto por patrones de entrada y salida. El objetivo del algoritmo de aprendizaje es ajustar los pesos de la red “w” de manera tal que la salida generada por la ANN sea lo más cercanamente posible a la verdadera salida dada una cierta entrada. Es decir, la red neuronal trata de encontrar un modelo al proceso desconocido que generó la salida. Este aprendizaje se llama supervisado pues se conoce el patrón de salida el cual hace el papel de supervisor de la red.

En cambio en el aprendizaje no supervisado se presenta sólo un conjunto de patrones a la ANN, y el objetivo del algoritmo de aprendizaje es ajustar los pesos de la red de manera tal que la red encuentre alguna estructura o configuración presente en los dato

### **2.5.1 El algoritmo *Back Propagación* (BP)**

BP es un algoritmo muy utilizado, y puede asignar procesos no lineales. Es una red *feedforward* con una o más capas ocultas. La arquitectura elemental de la red *back* propagación tiene tres capas. No hay restricciones sobre el número de las capas ocultas. BP es un método sistemático para el entrenamiento redes neuronales artificiales multicapa. En la figura 2.15, se muestra una

estructura red de *feedforward* multicapa, y todos los parámetros que se indican son de acuerdo con esta figura.

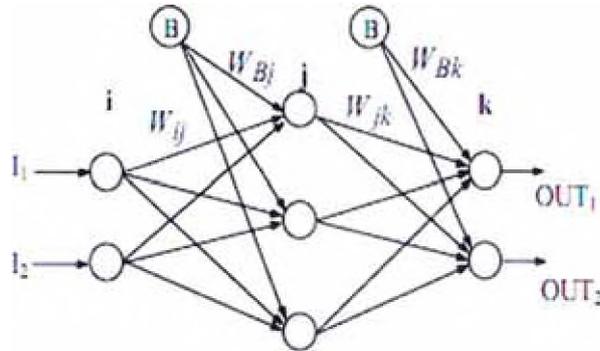


Figura 2.15 Estructura de Red Neuronal

### 2.5.2 El proceso *Forward Pass* en el Algoritmo BP

Los valores netos (NET) se obtienen multiplicando las entradas y los pesos relacionados. El cálculo de los valores NET de los perceptrones en la capa oculta está dada por la fórmula:

$$NET_j = \sum I_i W_{ij} + W_{Bj}^{NEW} \quad \dots 2.9$$

$$OUT_j = \frac{1}{1 + e^{-NET}} \quad \dots 2.10$$

Para calcular los valores de salida (OUT), a los valores NET se les aplica una función de activación. De manera similar, las salidas de las unidades ocultas son las entradas de las unidades próximas de las capas ocultas o capas de salidas. Los valores  $W_{Bj}^{NEW}$  y  $W_{Bk}^{NEW}$  presentan sesgos. OUT y NET son calculados usando las fórmulas:

$$NET_k = \sum I_j W_{jk} + W_{Bk}^{NEW} \quad \dots 2.11$$

$$OUT_k = \frac{1}{1 + e^{-NET_k}} \quad \dots 2.12$$

### 2.5.3 Error de propagación en el Algoritmo BP

Las salidas del modelo de red neuronal se obtienen de las unidades de capa de salida. La diferencia entre los valores objetivo y los valores actuales son considerados como error del sistema. Los valores de error obtenidos se propagan hacia atrás, al peso de la conexión. Este proceso se aplica mediante las ecuaciones siguientes;

En primer lugar, desde la capa de salida a la última capa oculta;

$$\delta_k = f'(NET_k)(TARGET_k - OUT_k) \quad \dots 2.13$$

$$\delta_k = OUT_k(1 - OUT_k)(TARGET_k - OUT_k) \quad \dots 2.14$$

$$\Delta W_{kj}(n+1) = \eta \delta_k OUT_k + \alpha [\Delta W_{kj}(n)] \quad \dots 2.15$$

$$W_{kj}^{NEW} = W_{kj}^{OLD} + \Delta W_{kj}(n+1) \quad \dots 2.16$$

Donde  $TARGET_k$  representa el valor de salida deseado.  $\eta$  es la tasa de aprendizaje,  $\alpha$  es el coeficiente de momentum,  $f(NET_k)$  presenta la función de activación,  $n$  es número de iteraciones y  $\Delta W$  es el cambio de peso relacionado. Este término se agrega al antiguo peso de la conexión relacionada para obtener una nueva.

En segundo lugar, desde la capa oculta a la entrada de la capa;

$$\delta_j = f'(NET_j) \cdot \sum \delta_k W_{kj} \quad \dots 2.17$$

$$\delta_j = (OUT_j(1 - OUT_j)) \sum \delta_k W_{kj} \quad \dots 2.18$$

$$\Delta W_{\mu j}(n+1) = \eta \delta_j OUT_j + \alpha [\Delta W_{\mu j}(n)] \quad \dots 2.19$$

$$W_{\mu j}^{NEW} = W_{\mu j}^{OLD} + \Delta W_{\mu j}(n+1) \quad \dots 2.20$$

El sesgo afecta la función de activación para obligar al proceso de aprendizaje, por lo tanto. la velocidad de aprendizaje aumenta el proceso. Se vuelven a calcular los sesgos como siguen, para los sesgos de la capa de salida, donde las letras y los símbolos tienen significados similares;

$$\Delta W_{BK}(n+1) = \eta \delta_k + \alpha [\Delta W_{BK}(n)] \quad \dots 2.21$$

$$W_{Bj}^{NEW} = W_{Bj}^{OLD} + \Delta W_{Bk}(n+1) \quad \dots 2.22$$

y para los sesgos de la capa oculta.

$$\Delta W_{Bj}(n+1) = \eta \delta_j + \alpha [\Delta W_{Bj}(n)] \quad \dots 2.23$$

$$W_{Bj}^{NEW} = W_{Bj}^{OLD} + \Delta W_{Bk}(n+1) \quad \dots 2.24$$

El entrenamiento del modelo de red neuronal, puede ser entendido desde el proceso anterior es llevado en dos pasos. El primer paso se llama *Forward Pass*, es decir, compuesta de cálculo para *NET* y *OUT*. El segundo paso se llama *Backward Pass* que se compone de propagación de errores a lo largo de pesos de conexión. El proceso iterativo se repite hasta un aprendizaje satisfactorio, es decir, las diferencias entre el *TARGET* y *OUT* sean mínimas [12]

### 3. PROCEDIMIENTOS EXPERIMENTALES

Para el reconocimiento de la placa vehicular de una imagen. Primero tenemos que adquirir imágenes para la cual se utilizó una cámara fotográfica de teléfono móvil marca Nokia de 2 Mega píxel de resolución.

Para el procesamiento de los datos se utilizó una computadora portátil Marca Compaq, sistema Operativo Windows Seven, Procesador AMD, 2 MB de memoria RAM.

#### 3.1 Conversión de la Imagen RGB a HSI

La figura 3.1 muestra la imagen en formato jpg de tamaño 600 x 1200 de ancho y alto respectivamente, donde objetivo es obtener los caracteres de la placa.

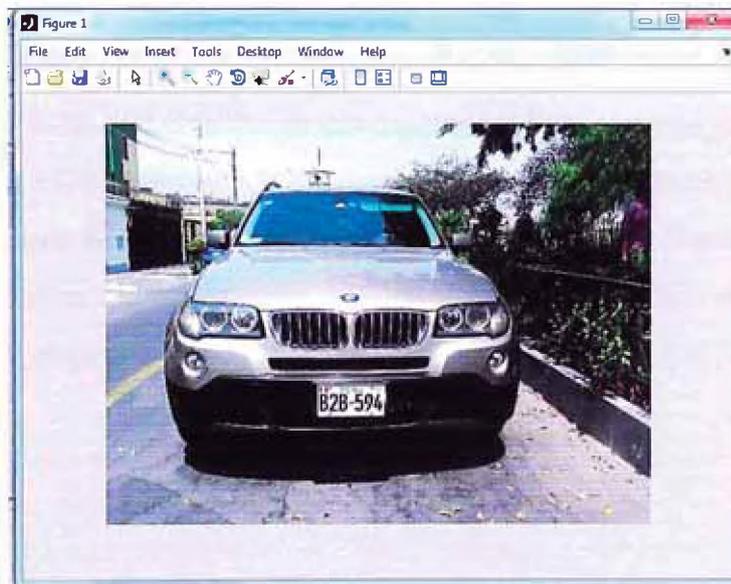


Figura 3.1 fotografía de automóvil

En la figura 3.2 se muestra la imagen recortada, la región donde hay más probabilidad que esté ubicada la placa, esto se hace con intención de mejorar el tiempo de detección de la placa.

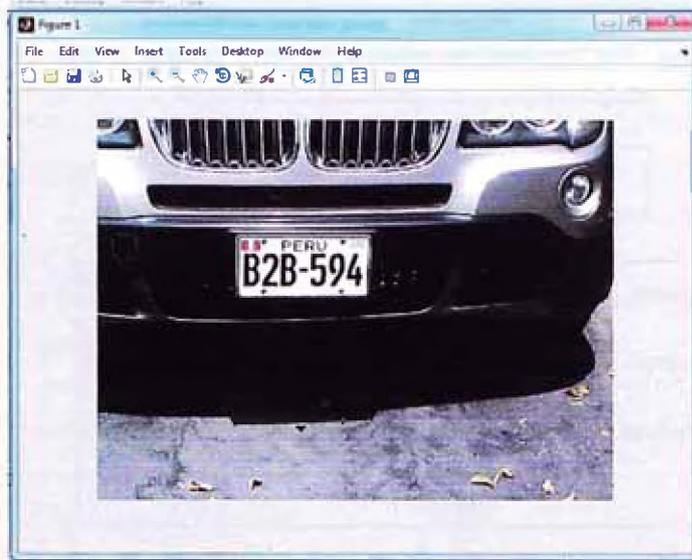


Figura 3.2 fotografía de automóvil recortada

Adquirida la imagen RGB se procede a convertirla a imagen HSI de acuerdo al algoritmo de la figura 3.3. La Imagen RGB está constituida por 3 componentes como se mencionó en el capítulo 2. La variable Imagen RGB representa la imagen descargada,  $n$  representa la ubicación del píxel en la matriz, cada imagen RGB está compuesta de 3 matrices R, G y B que son las componentes (comp) que la constituyen. De esta manera, la variable ImagenHSI está compuesta de 3 matrices H, S e I. Para realizar estas conversiones se utilizan las ecuaciones 2.3 al 2.5 del capítulo 2

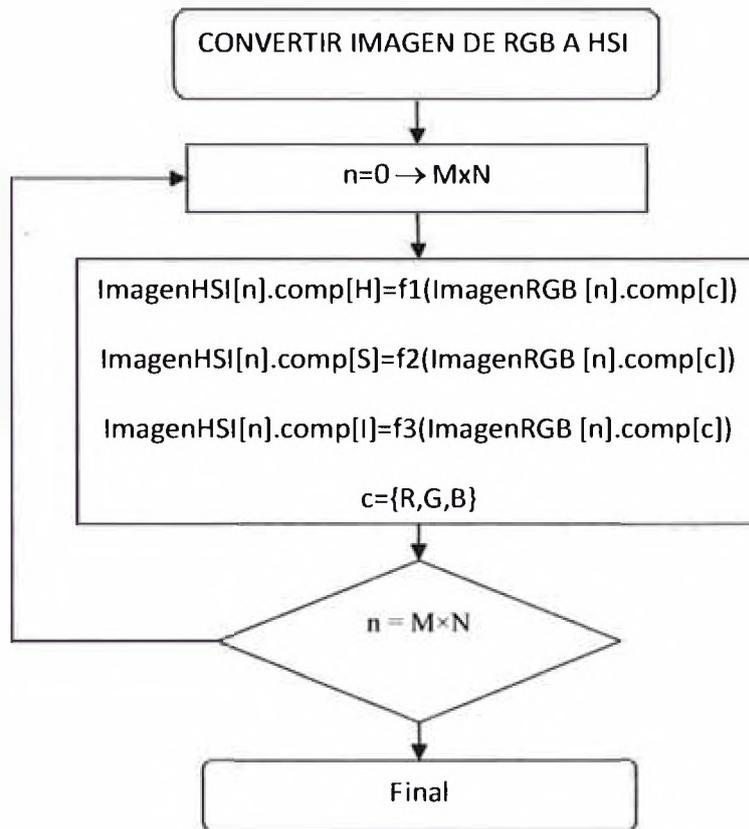


Figura 3.3 Diagrama de flujo que convierte la imagen RGB a HSI

En la figura 3.4 se observa la imagen HSI

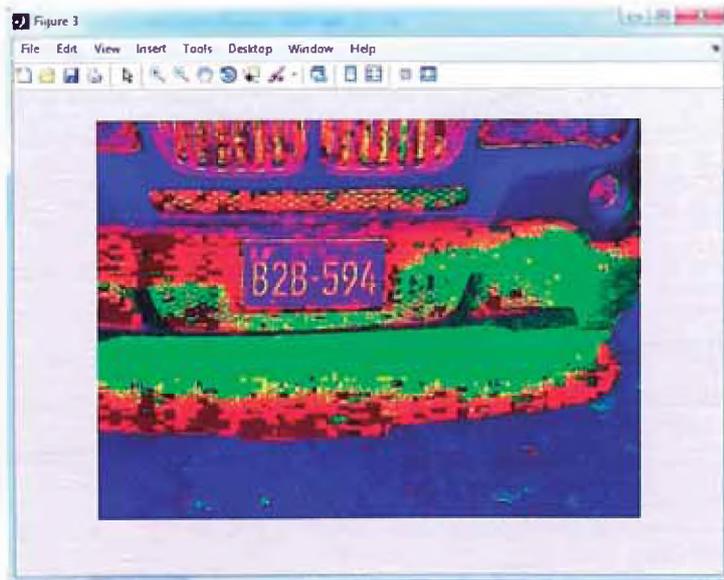


Figura 3.4 Imagen en componente HSI

### 3.2 Binarización de la Imagen HSI

Luego de obtenida la imagen HSI se procede a binarizar la imagen con la intención de buscar regiones claras, ya que el fondo de la placa es blanco. La figura 3.5 muestra el algoritmo de como se procede a binarizar la imagen HSI, para ello utilizamos las ecuaciones 2.6 al 2.8 mencionadas en el capítulo 2

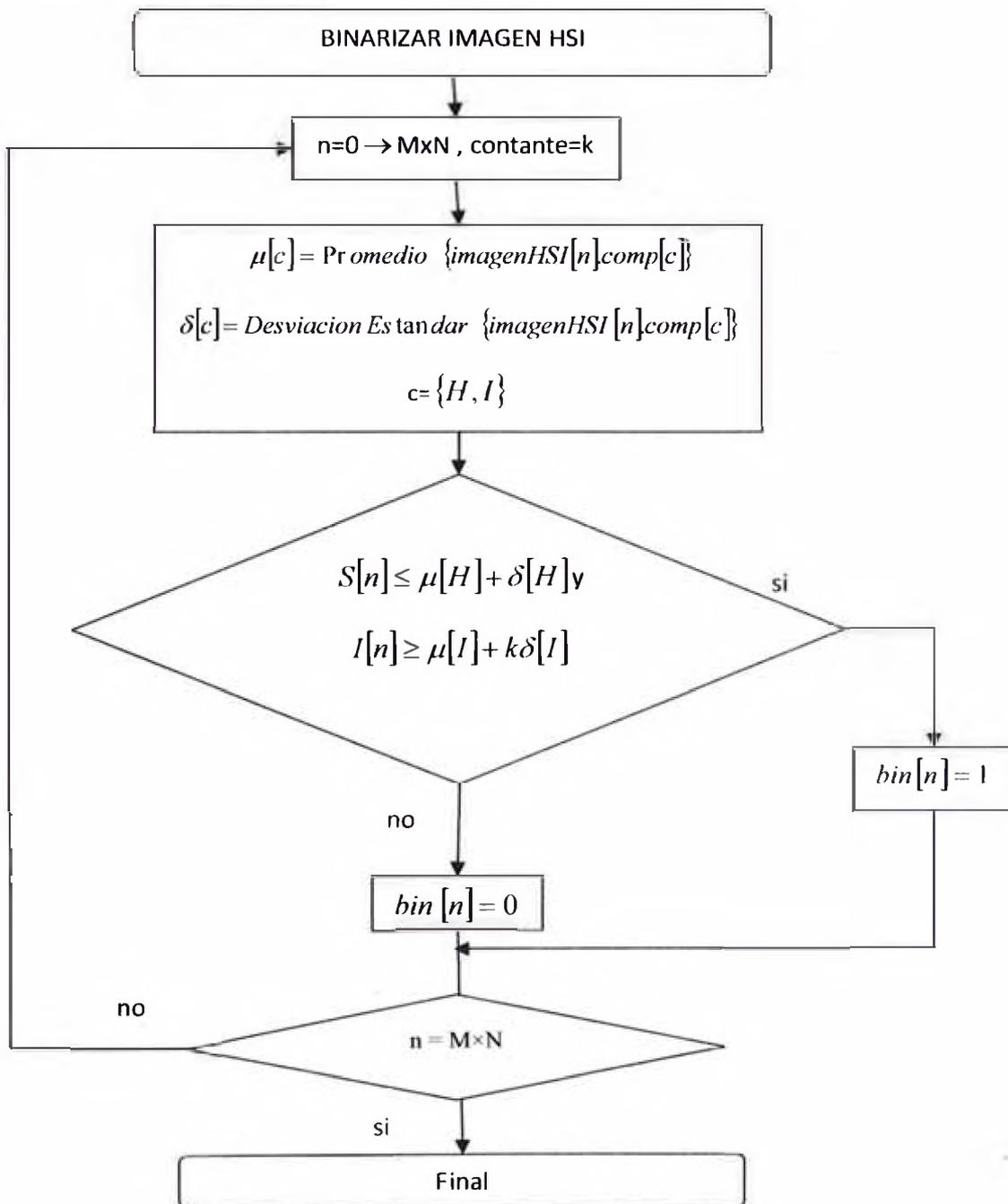


Figura 3.5 Diagrama de flujo que binariza la imagen HSI

En la figura 3.6 se observa la imagen binarizada, esto quiere decir que la imagen está compuesta por ceros y unos, donde cero es blanco y uno es negro. Se puede apreciar que el fondo de la placa donde se ve claramente los caracteres es blanco.

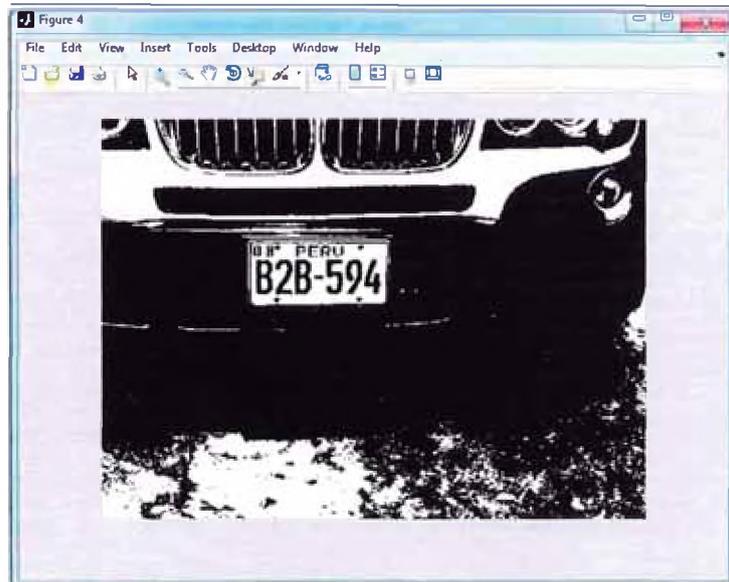


Figura 3.6 Imagen Binarizada

### 3.3 Operaciones Morfológicas y detección de regiones candidatas

Realizada la conversión se procede a hacer las operaciones morfológicas de la imagen binarizada en la Figura 3.7 se aprecia el paso remover ruido de la imagen para localizar la placa.

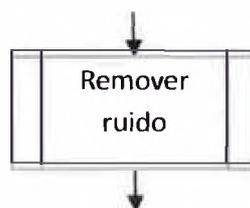


Figura 3.7 Remover ruido

La operación morfológica “Remover ruido”, es decir que se remueve regiones aisladas cuya área es pequeña comparable al área o región donde se encuentra la placa. En la Figura 3.8 se ve más “limpia” comparada con la figura 3.6

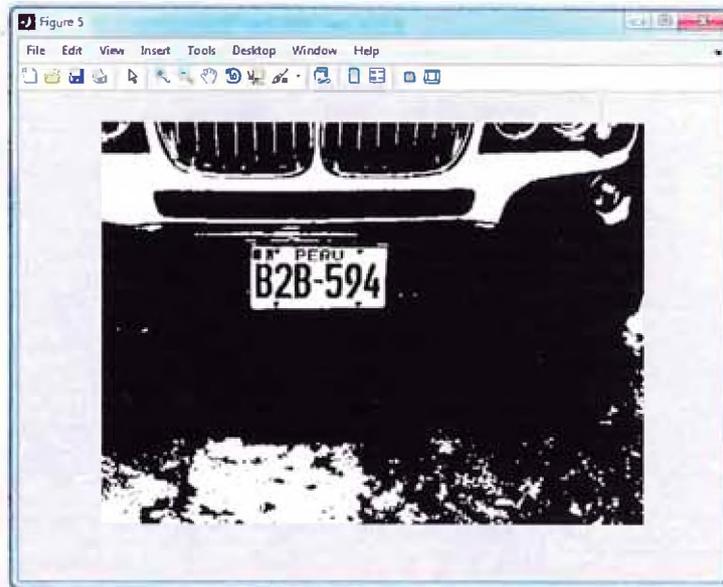


Figura 3.8 imagen binarizada “sin ruido”

La imagen es segmentada en regiones, cada región tiene características como por ejemplo: “area\_real”, que es la cantidad de píxeles que hay en esa región, “perimetro\_real” que es el contorno de esa región. Ahora toda región puede ser contenida por rectángulos y cuando hablamos del ancho, altura, área o perímetro nos referimos a esa figura geométrica que contiene a dicha región en estudio. En la Figura 3.9 se muestra el diagrama de flujo que permite detectar áreas más probables o candidato para ubicar la región donde se encuentra la placa.

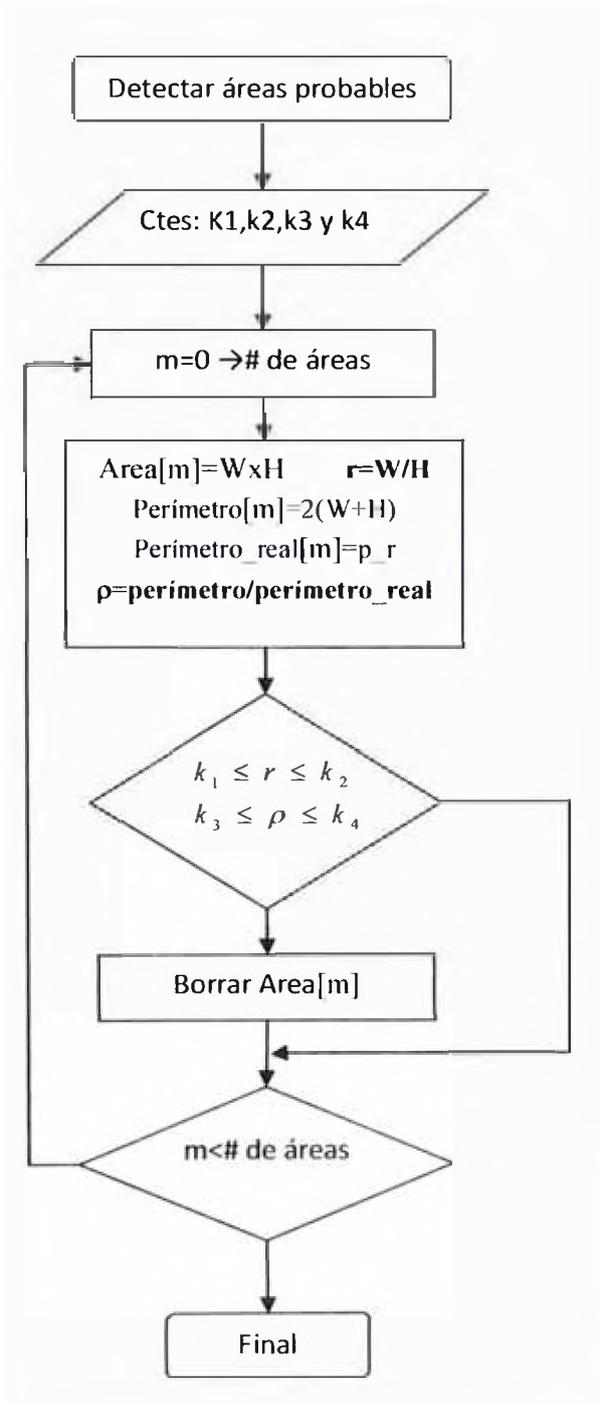


Figura 3.9 Diagrama de flujo que detecta áreas probables.

En la figura 3.10 se muestra la imagen con las regiones candidatas que cumplen con las condiciones del algoritmo mostrado en la figura 3.9.

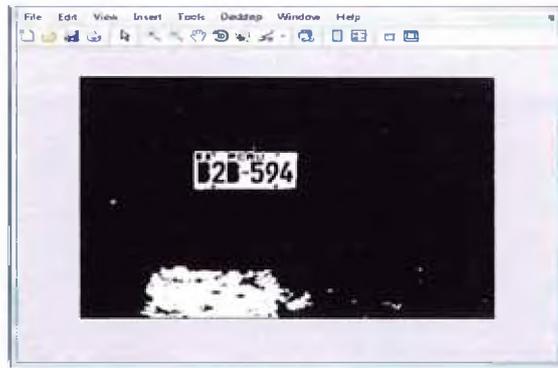
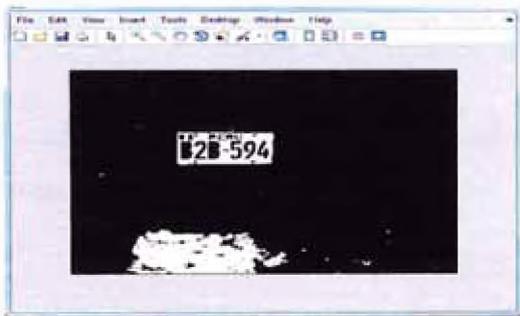
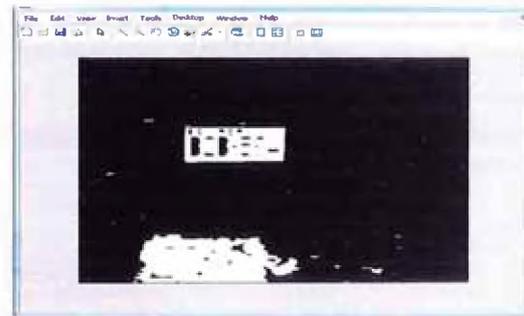


Figura 3.10 muestra los resultados de las regiones candidatas

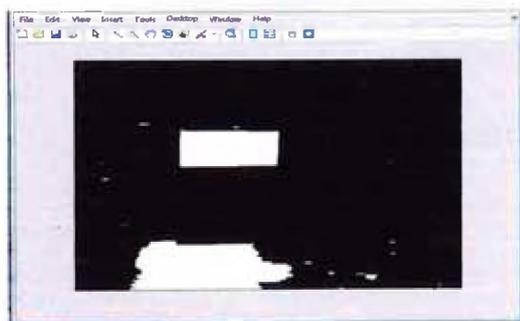
En la figura 3.11 (a), (b), (c) y (d) se muestra las operaciones morfológicas a la imagen



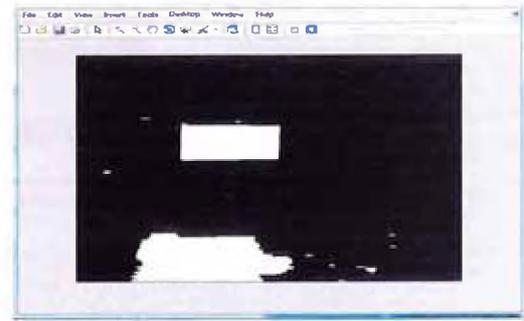
(a)



(b)



(c)



(d)

Figura 3.11 (a) Imagen dilatada (b) Imagen erosionada (c) Imagen llena agujeros con pixeles (d) Imagen donde se ha eliminado pequeños píxeles aislados

Luego de realizar las operaciones morfológicas, repetimos el algoritmo mostrado en la figura 3.9 y obtenemos la imagen mostrada en la figura 3.12

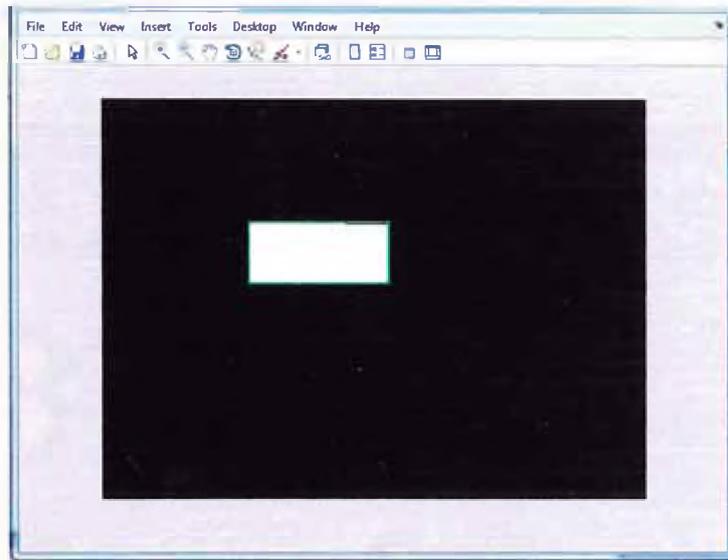


Figura 3.12 Muestra la región donde se encuentra la placa

De lo anterior obtenemos las coordenadas donde se encuentra la placa en la imagen recortada como se muestra en la figura 3.13



Figura 3.13 la región de la placa ubicada en colores RGB

La figura 3.14 muestra la imagen de la placa binarizada, a la cual se le ha removido las áreas pequeñas y luego se ha segmentado, obteniendo los seis caracteres.

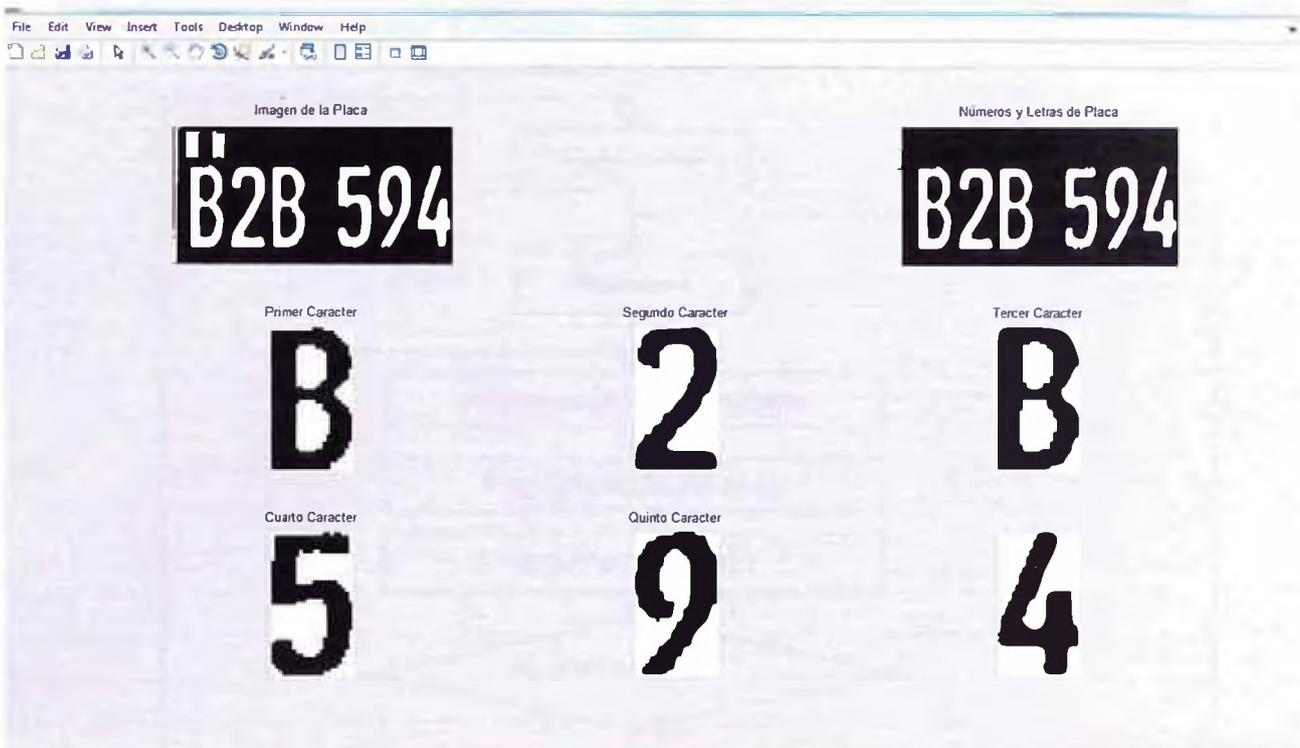


Figura 3.14 Segmentación de placa

### 3.4 Reconocimiento de los caracteres aplicando redes neuronales artificiales

Primero, para realizar el reconocimiento de cada carácter, tenemos que entrenar la red neuronal para ello con los prototipos o patrones de los caracteres de 0 - 9 y A - Z. La figura 3.15 muestra el algoritmo de entrenamiento de la red neuronal artificial:

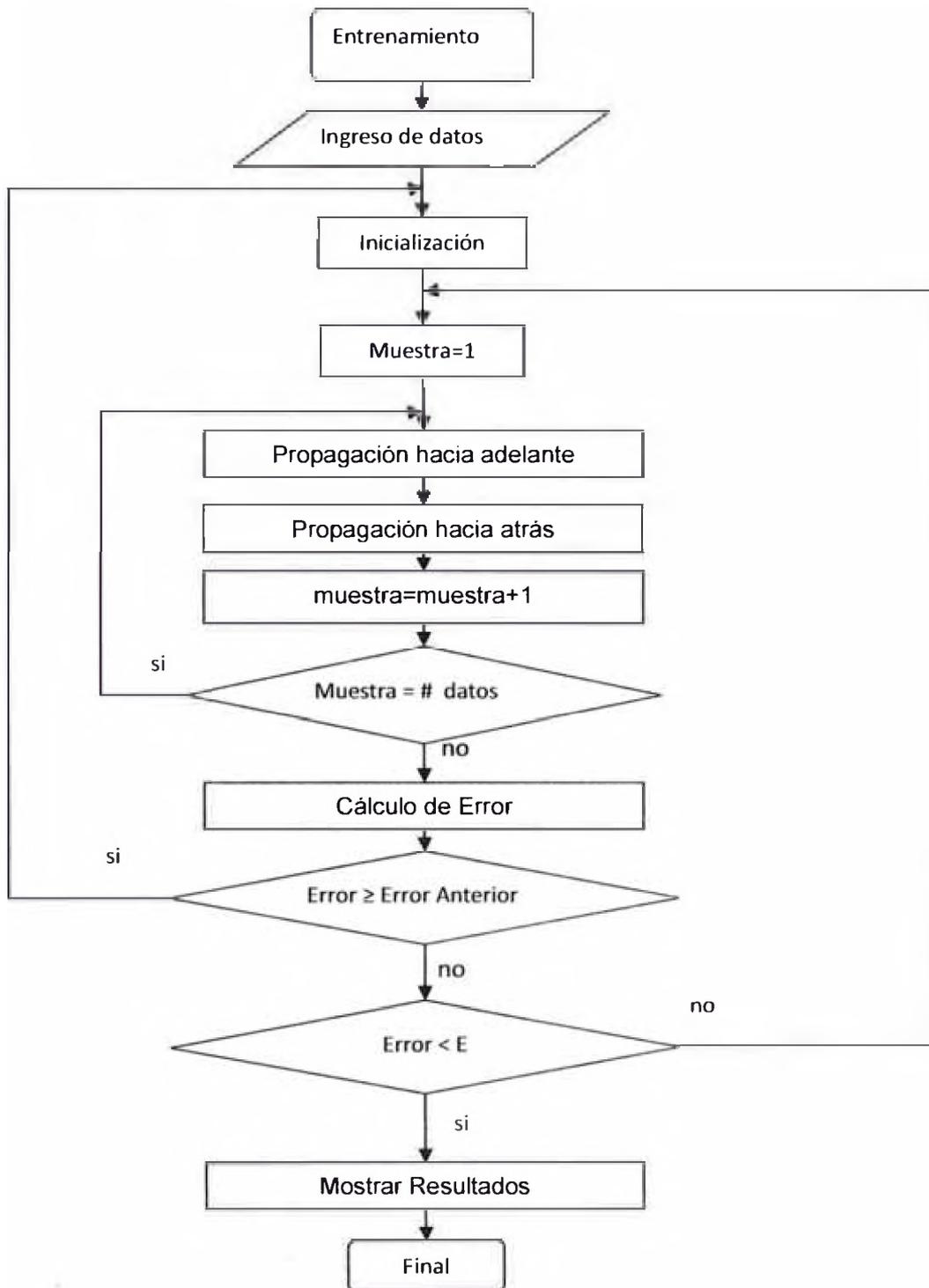


Figura 3.15 algoritmo de la red neuronal

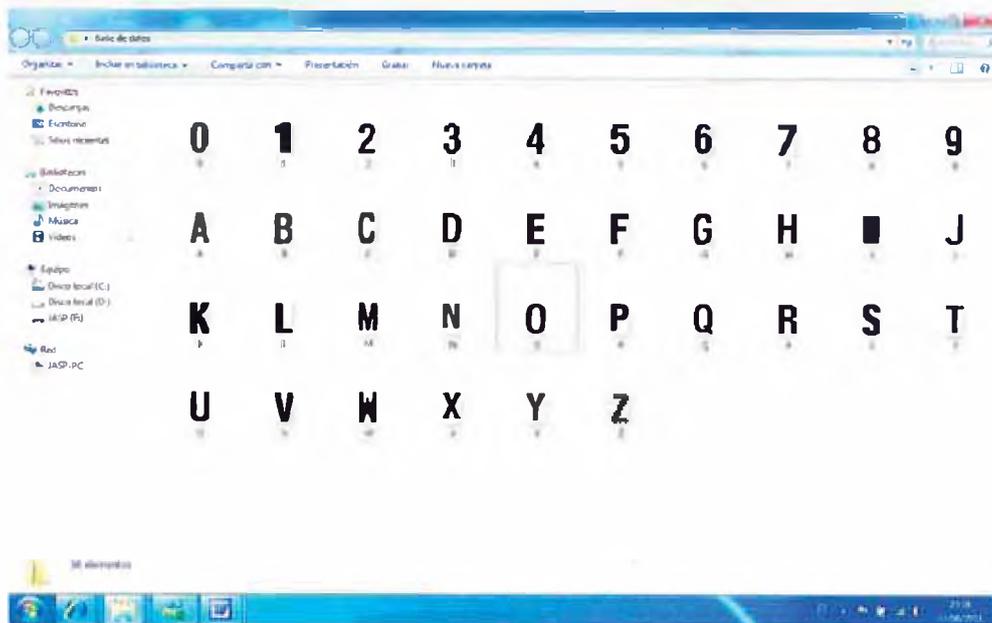


Figura 3.16 Base de datos de patrones

### 3.4.1 Entrenamiento de la red neuronal

La Figura 3.15, describe el diagrama de flujo del programa de entrenamiento *Back* propagación (BP) de la red neuronal multicapa que ha sido codificado en matlab.

### 3.4.2 Ingreso de datos

Se introducen los datos que definen el tamaño de la red neuronal como son el número de entrada, salidas, capas y neuronas. Por cada capa, estas últimas se registran en un vector de tamaño variable dependiendo del número de neuronales ingresadas.

Luego se ingresan el número de muestras para el entrenamiento a continuación se piden ingresar los valores de las entradas y salidas de cada muestra, que se guarda como una fila de una matriz de dos dimensiones.

### 3.4.3 Inicialización

Los pesos y los sesgos para cada neurona obtienen su valor inicial en forma aleatoria con un límite establecido. El parámetro de entrenamiento empieza con un valor pequeño y será de utilidad para la dinámica de aprendizaje de la red. Los sesgos son almacenados en una matriz de

dos dimensiones, donde el número de fila indica la capa en la que se encuentra la neurona y el número de columna indica el número de neurona en esa capa. Los pesos son almacenados en una matriz de tres dimensiones, los dos primeros índices, expresan a que neurona pertenece cada peso, análogamente al caso de los sesgos, con la diferencia que el tercer índice hace referencia al número de la neurona de la capa anterior a la cual está conectada. Tanto para la asignación de pesos y sesgos. Los valores iniciales son configurados con signo intercambiado esto con la finalidad de que la media sea cercana a cero y acelerar la convergencia. La tasa de aprendizaje va disminuyendo de valor hasta la última, para hacer que los valores finales de entrenamiento sean cercanos.

#### **3.4.4 Propagación hacia adelante**

Los valores de las entradas de la muestra actual se ingresan como salidas para la primera capa entonces utilizando los pesos y sesgos de las neuronas se consigue la respuesta de ellas ( que se , almacena en una matriz de dos dimensiones con formato de índice igual a la matriz sesgos), las cuales son llevadas como entradas para la siguiente capa y se forma un proceso iterativo hasta llegar a la última capa.

#### **3.4.5 Propagación hacia atrás**

Se obtiene las señales del error en cada salida (diferencias entre la señal deseada y obtenida) para calcular el error de la muestra, este valor es almacenado como elemento de un vector, para obtener el error cuadrático medio de un vector cuando se revisen todas las muestras. Se obtienen los gradientes de las neuronas en la capa de salida y luego se forma un proceso iterativo para obtener los gradientes en las neuronas de la capa anteriores. Seguidamente se modifican los pesos con los valores de los gradientes (almacenados en una matriz de dos dimensiones, con el formato de los índices igual que los sesgos), la respuesta de la neurona en la capa anterior, la tasa de aprendizaje de la capa y valor actual del peso a ser modificado.

#### **3.4.6 Cálculo de error**

Con el vector de error completo se procede a calcular el error cuadrático medio, fundamental para decidir la convergencia de los pesos en el entrenamiento

### 3.4.7 Mostrar resultados

Se realiza una impresión de los valores de los pesos y los sesgos resultante del entrenamiento, así como el error cuadrático medio al se que se llegó al terminar el entrenamiento.



Figura 3.17 Muestra de las fotografías utilizadas.

Utilizando las fotografías como las que se muestran en la figura 3.17 de cada carácter y los resultados del algoritmo de la figura 3.15, obtenemos los resultados.

#### 4. RESULTADOS EXPERIMENTALES

Las imágenes de prueba fueron tomadas en un entorno no controlado con diferentes tipos de iluminación del día. Toda las placas fotografiadas pertenecen a autos particulares, cuyas placas cumplen los requisitos de la categoría M, fondo blanco, de acuerdo a lo establecido en el Reglamento de Placa Única Nacional de Rodaje [13], aprobado por Decreto Supremo N°017-2008-MTC y modificada por los Decretos Supremos N°015-2009-MTC.

Las fotografías fueron tomadas frontalmente de tal manera que no se realizó corrección por ángulo. En la figura 4.1 se presentan los resultados obtenidos:

Número de fotografías	Placas detectadas	Porcentaje de éxito
50	35	70 %

Figura 4.1 Tabla de resultado

Ya que el algoritmo es un proceso que depende 3 etapas, hemos realizado el seguimiento para observar en que etapa se originó el error, obteniéndose lo siguiente resultados figura 4.2.

	Placas no detectadas	Porcentaje
Número de fotografías	15	30 %
Ubicación de la placa	7	14 %
Segmentación de Caracteres	4	8 %
Reconocimiento de caracteres	4	8 %

Figura 4.2 Tabla de resultado de errores por etapa

## 5. DISCUSIÓN

En la figura 4.1 se muestra los resultados para de la detección de la placa, detectándose exitosamente 35 de una población de 50 fotografía, lo equivale a 70 % de efectividad.

La efectividad de la primera etapa (localización de la placa) es relevante para la etapas siguientes, es decir, si no se detecta adecuadamente la placa, los resultados de las otras dos etapas, no repercute en el resultado final, simplemente, no será exitoso. De igual manera el error que hubiera en la segunda etapa (segmentación de caracteres), esto se debe principalmente al ruido, repercute en la última etapa de reconocimiento. En la última, la efectividad netamente es del algoritmo de reconocimiento por redes neuronales artificiales, en la cual detectamos algunos casos de falsos positivos que son los resultados de la detección de una placa vehicular que no es correcto. Ya que hay casos donde no se detecta adecuadamente el carácter, por ejemplo el algoritmo detecta D como 0 (cero), la S como 5 o viceversa.

De acuerdo a la tabla de resultados de la figura 4.2, se puede observar que el error mayor es en la etapa de localización de la placa, 14 %, esto se debe a la baja o alta iluminación en la imagen de la placa, y cuando se realiza la binarización, la región de interés (donde se encuentra la placa) se pierde. También están los problemas de ruido de la foto, que son pequeñas regiones aisladas, que se anexan y modifican la región de interés y por lo tanto, el algoritmo no detecta ninguna región que cumpla con las características geométricas que tiene una placa. En caso de la etapa de segmentación de los caracteres, el más frecuente es el ruido, el porcentaje de error de la muestra es 8 %. En la última etapa que es la de reconocimiento de caracteres el error representa el 8 % de total de muestra, esto se debe a la falta de entrenamiento de la red.

## **6. CONCLUSIONES**

Se logró desarrollar un sistema con un algoritmo eficiente que detecta la placa en una imagen con 70 % de efectividad.

Se comprobó que la aplicación de redes neuronales a reconocimiento de caracteres es adecuado.

Se comprobó que el método de ubicación de la placa por el método de segmentación por color es apropiado.

## 7. REFERENCIAS

- [1] B. H. Bransden and C. J. Joachain. Introduction to quantum mechanics, Longman Scientific & Technical, New York, 1989.
- [2] J. Uymura. Introduction to VLSI circuits and systems, John Wiley and sons, 2002.
- [3] Guozhong Cao. Nanostructures & nanomaterials, synthesis, properties & applications, Imperial College Press, London, 2004.
- [4] Stuart A. Taylor. CCD and CMOS Imaging Array Technologies: Technology Review. Xerox Research Centre Europe, 1998.
- [5] James J. Brophy. Electrónica fundamental para científicos. Editorial Reverté, S.A, Nueva York, 1969.
- [6] Sears Francis y otros, "Física Universitaria" Addison Wesley Longman, Novena Edición, 1999, México.
- [7] Serway Raymond y Jewett John, "Física para Ciencias e Ingeniería" Cengage Learning Editores, Séptima Edición, 2008, México.
- [8] R. Gonzalez, R. Woods, "Digital Image Processing". Prentice Hall, Segunda Edición, 2002
- [9] [http://en.wikipedia.org/wiki/Bayer\\_filter](http://en.wikipedia.org/wiki/Bayer_filter) consultado el 2011-02-05
- [10] <http://www.quesabesde.com/camdig/articulos.asp?articulo=119> consultado el 2011-02-14
- [11] Kaushik Deband Kang-Hyun Jo, "Segmenting the License Plate Region Using a Color Model", Peng-Yeng Yin Ed., Pattern Recognition, Capítulo 22, pp. 401-418. Austria 2009.
- [12] Cemil Oz y F. Ercal, "Automatic Vehicle License Plate Recognition Using Artificial Neural Networks". Conference on Intelligent Systems Design and Applications (ISDA 2003), Advances in soft Computing, Springer Verlag, pp. 23-32.
- [13] <http://placas.pe/Documentos/DS-017-2008-MTC.doc> consultado el 2011-02-25.