

870116

1
Ej.

UNIVERSIDAD AUTONOMA DE GUADALAJARA

INCORPORADA A LA UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO

ESCUELA DE INGENIERIA EN COMPUTACION



RECONOCIMIENTO DE FORMAS E IMAGENES POR COMPUTADORA

TESIS PROFESIONAL

QUE PARA OBTENER EL TITULO DE
INGENIERO EN COMPUTACION

P R E S E N T A

DUSTANO PINEDA TIZNADO

GUADALAJARA, JAL. NOVIEMBRE DE 1994

TESIS CON
FALLA DE ORIGEN



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas Tesis Digitales Restricciones de uso

DERECHOS RESERVADOS © PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis está protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.



" INGENIERÍA EN COMPUTACION "

Guadalajara, Jalisco a 3 de noviembre de 1994.

SR. DUSTANO PINEDA TIZNADO
PASANTE DE: INGENIERO EN COMPUTACION
P R E S E N T E:

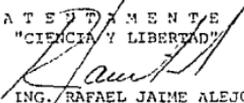
Me es grato informarle que la Comisión de Tesis que me honro en presidir, despues de haber revisado detalladamente el trabajo intitulado:

" RECONOCIMIENTO DE FORMAS E IMAGENES POR COMPUTADORA "

El cual nos ha presentado como requisito para sustentar su examen re
cepcional de INGENIERO EN COMPUTACION, ha recomendado que habiendo cubierto los requi
sitos formales y de contenido se aprueba la impresión del mismo.

Ruego a usted tomar nota que la copia del presente oficio deberá ser-
incluido en los preliminares en todo ejemplar de su trabajo.

A T E N T A M E N T E
"CIENCIA Y LIBERTAD"


ING. RAFAEL JAIME ALEJO
DIRECTOR
ESC. DE ING. EN COMPUTACION.

III

DEDICATORIAS

A DIOS : Por permitir la realizacion de este objetivo

A MIS PADRES : Por su gran responsabilidad, amor y apoyo

A MI ESPOSA Y MI HIJO : Por ser lo que mas quiero

V
INDICE GENERAL

	No. PAG.
DEDICATORIAS	iii
INDICE	v
INDICE DE TABLAS	viii
INDICE DE FIGURAS	ix
INTRODUCCION	xiv
<i>CAPITULO I . DETECCION Y DIGITALIZACION DE IMAGENES</i>	
1.0 INTRODUCCION A LA DETECCION Y DIGITALIZACION DE IMAGENES.	1
1.1 ADQUISICION DE LA IMAGEN (CAPTACION)	3
1.2 CONVERSION DE SEÑALES	4
1.2.1 MUESTREO	6
1.2.2 CUANTIFICACION	8
1.2.3 CODIFICACION	9
1.3 ALMACENAMIENTO DE LA IMAGEN	11
1.4 TECNICAS DE ILUMINACION	13
1.5 LA INTERFASE (EL DIGITALIZADOR)	16
1.6 CONCLUSIONES	18

VI

CAPITULO II. *SEGMENTACION Y PREPROCESAMIENTO.*

2.0	INTRODUCCION A LA SEGMENTACION Y PREPROCESAMIENTO	19
2.1	PREPROCESO	20
2.1.1	ELIMINACION DEL RUIDO (SUAVIZADO)	20
2.1.2	PERFILADO	24
2.2	SEGMENTACION	27
2.2.1	FIJACION DE UMBRALES	28
2.2.2	CRECIMIENTO DE REGION	31
2.2.3	DETECCION DE BORDES	34

CAPITULO III. *EXTRACCION DE CARACTERISTICAS (DESCRIPCION)*

3.0	INTRODUCCION A LA EXTRACCION DE CARACTERISTICAS	37
3.1	DESCRIPTORES DE FRONTERA	40
3.1.1	CODIGOS DE CADENA	40
3.1.2	SIGNATURAS	43
3.1.3	APROXIMACIONES POLIGONALES	44
3.1.4	NUMERO DE CONTORNOS	46
3.2	DESCRIPTORES DE REGION	49

VII

CAPITULO IV. RECONOCIMIENTO DE FORMAS.

4.0	INTRODUCCION AL RECONOCIMIENTO DE FORMAS	50
4.1	METODOS	52
4.2	PROBLEMAS COMUNES DEL RECONOCIMIENTO	57
4.3	APLICACIONES	60

CAPITULO V. PROGRAMACION E INTERPRETACION

5.0	INTRODUCCION A LA PROGRAMACION Y LA INTERPRETACION DE RESULTADOS	61
5.1	PROGRAMAS DE PROCESAMIENTO	62
5.1.1	SUAVIZADO	62
5.2	PROGRAMAS DE RECONOCIMIENTO	74
5.2.1	RECONOCIMIENTO POR ANGULOS	74
5.2.1.1	ANALISIS	77
5.2.2	RECONOCIMIENTO POR PUNTOS CLAVE	80
5.2.2.1	ANALISIS	82
5.2.3	RECONOCIMIENTO DELTA-D	85
5.2.3.1	ANALISIS	87
5.2.4	RECONOCIMIENTO SINTACTICO	91
5.2.4.1	ANALISIS	95
	CONCLUSIONES	XVI
	BIBLIOGRAFIA	XVII

VIII

INDICE DE TABLAS

	No. PAG.
CAPITULO I	
TABLA 1 CODIFICACION PARA LA ESCALA DE GRISES.	10
TABLA 2 TECNICAS DE ILUMINACION	14
CAPITULO III	
TABLA 3 CARACTERISTICAS Y MEDIDAS PARA LA IDENTIFICACION DE OBJETOS BIDIMENSIONALES	39
CAPITULO V	
TABLA 4 CRITERIOS UTILIZADOS PARA EL SUAVIZADO DE IMAGENES	62-63
TABLA 5 BITACORA DE EVALUACION	64
TABLA 6 CASOS DEL RECONOCIMIENTO SINTACTICO	94

IX

INDICE DE FIGURAS

	No. PAG.
CAPITULO I	
FIG. 1 FASE DE DETECCION Y DIGITALIZACION DE IMAGENES	3
FIG. 2 CONEXION MICROCOMPUTADORA-ADC	4
FIG. 3 CONVERSION A/D	5
FIG. 4 MUESTREO Y DIGITALIZACION DE UNA FORMA DE ONDA ANALOGICA	6
FIG. 5 MEMORIA RAM PROPUESTA	12
FIG. 6 ILUMINACION FRONTAL Y POSTERIOR	15
FIG. 7 METODO DE ILUMINACION	15
FIG. 8 DIAGRAMA DE BLOQUES DEL DIGITALIZADOR PROPUESTO	17
CAPITULO II	
FIG. 9 MASCARA DE 3*3	21
FIG. 10 SUAVIZADO (IMAGEN ORIGINAL)	22
FIG. 11 SUAVIZADO B1	23
FIG. 12 SUAVIZADO B2	23
FIG. 13 HISTOGRAMA DE INTENSIDAD Y APROXIMACION COMO SUMA DE FUNCION DE DENSIDAD DE PROBABILIDAD	30
FIG. 14 CRECIMIENTO DE REGION	31

X

FIG. 15 CRECIMIENTO DE REGION	32
FIG. 16 DETECCION DE ARISTA POR OPERADORES DE DERIVADA	35
CAPITULO III	
FIG. 17 CODIGOS DE CADENA	40
FIG. 18 PASOS PARA OBTENER EL CODIGO DE CADENA	41
FIG. 19 EJEMPLO DE CODIGO DE CADENA	42
FIG. 20 SIGNATURAS DE DISTANCIA CON RESPECTO AL ANGULO	43
FIG. 21 APROXIMACIONES POLIGONALES	44
FIG. 22 NUMERO DE CONTORNOS	46
FIG. 23 PASOS PARA GENERAR EL NUMERO DE CONTORNOS	48
CAPITULO IV	
FIG. 24 SISTEMA DE RECONOCIMIENTO DE FORMAS	51
FIG. 25 DESCOMPOSICION DE UN OBJETO EN PATRONES BASICOS	53
FIG. 26 DESCRIPCION SINTACTICA DEL CARACTER "A"	55
FIG. 27 DESCRIPCION ARBORECENTE DE UNA IMAGEN	56
FIG. 28 ILUSION OPTICA	58
CAPITULO V	
FIG. 29 EFECTO DE ESCALERA EN EL SUAIVIZADO B2	64

XI

FIG. 30 DIAGRAMA DE BLOQUES DEL PROGRAMA DE SUAVIZADO	65
FIG. 31 DIAGRAMA DE FLUJO DEL BLOQUE DE SUAVIZADO	66
FIG. 32 DIAGRAMA DE FLUJO DEL SUAVIZADO B1	67
FIG. 33 DIAGRAMA DE FLUJO DEL SUAVIZADO B2	68
FIG. 34 FIGURA A DE LA SECUENCIA DEL PROGRAMA DE SUAVIZADO	69
FIG. 35 FIGURA B DE LA SECUENCIA DEL PROGRAMA DE SUAVIZADO	69
FIG. 36 FIGURA C DE LA SECUENCIA DEL PROGRAMA DE SUAVIZADO	70
FIG. 37 FIGURA D DE LA SECUENCIA DEL PROGRAMA DE SUAVIZADO	71
FIG. 38 FIGURA E DE LA SECUENCIA DEL PROGRAMA DE SUAVIZADO	71
FIG. 39 FIGURA F DE LA SECUENCIA DEL PROGRAMA DE SUAVIZADO	72
FIG. 40 FIGURA G DE LA SECUENCIA DEL PROGRAMA DE SUAVIZADO	72
FIG. 41 FIGURA H DE LA SECUENCIA DEL PROGRAMA DE SUAVIZADO	73
FIG. 42 FIGURA I DE LA SECUENCIA DEL PROGRAMA DE SUAVIZADO	73
FIG. 43 DIAGRAMA DEL RECONOCIMIENTO POR ANGULOS	76

XII

FIG. 44 EJEMPLO DEL PROGRAMA DE RECONOCIMIENTO POR ANGULOS	77
FIG. 45 TRASLAPAMIENTO DE FIGURAS EN EL REC. POR ANGULOS	77
FIG. 46 TRASLAPAMIENTO DE FIGURAS EN EL REC. POR ANGULOS	78
FIG. 47 TRASLAPAMIENTO DE FIGURAS EN EL REC. POR ANGULOS	78
FIG. 48 PUNTOS CLAVES EN LAS FIGURAS A ANALIZAR	80
FIG. 49 DIAGRAMA DEL PROG. DE RECONOCIMIENTO POR PUNTOS CLAVE.	81
FIG. 50 EJEMPLO DEL PROG. DE RECONOCIMIENTO POR PUNTOS CLAVE	82
FIG. 51 TRASLAPAMIENTO EN EL PROG. DE REC. POR PUNTOS CLAVE	82
FIG. 52 TRASLAPAMIENTO EN EL PROG. DE REC. POR PUNTOS CLAVE	83
FIG. 53 DIF. TIPOS DE TRASLAPAMIENTOS EN REC. POR PUNTOS CLAVE.	84
FIG. 54 DIF. TIPOS DE TRASLAPAMIENTOS EN REC. POR PUNTOS CLAVE.	84
FIG. 55 DIAGRAMA DEL RECONOCEDOR DELTA_D	86
FIG. 56 PRESENTACION DEL PROGRAMA DELTA_D	87
FIG. 57 TRASLAPAMIENTO DE FIGURAS EN EL REC. DELTA_D	88

XIII

FIG. 58 DIF. TIPOS DE TRASLAPAMIENTOS EN EL REC. DELTA_D	89
FIG. 59 DIF. TIPOS DE TRASLAPAMIENTOS EN EL REC. DELTA_D	89
FIG. 60 DIF. TIPOS DE TRASLAPAMIENTOS EN EL REC. DELTA_D	90
FIG. 61 DIAGRAMA DEL PROGRAMA DE RECONOCIMIENTO SINTACTICO	92
FIG. 62 DIAGRAMA DE FLUJO DEL RECONOCIMIENTO SINTACTICO	93
FIG. 63 PRESENTACION DEL RECONOCIMIENTO SINTACTICO	96
FIG. 64 EJEMPLO DE UNA FIGURA DURANTE EL REC. SINTACTICO	96
FIG. 65 EJEMPLO DE UNA FIGURA YA RECONOCIDA	97

XIV

INTRODUCCION:

En los últimos años se ha venido observando un notable y desmedido crecimiento en el área de la inteligencia artificial; tan rápido es este avance que aún no nos acostumbramos a algo cuando ya está encima una versión más sofisticada de lo mismo. El tiempo cambia, y el hombre junto con él debe de acoplarse, por lo que es una necesidad encontrar una manera de adaptarse al entorno lo mejor posible, es ahí donde se fundamenta el rápido crecimiento de las áreas de automatización en los últimos años, y probablemente sea por esto, que la inteligencia artificial ha pasado de ser un pequeño aspecto de la ciencia informática a ser quizá la aportación más importante a la informática desde la invención del transistor.

De las distintas ramas de la inteligencia artificial se ha escogido el área de visión y reconocimiento de modelos para desarrollar este trabajo de tesis, en el cual se ofrece un estudio de esta extensa rama que es el procesamiento de imágenes y reconocimiento de formas.

El propósito de este trabajo es proporcionar a la gente aficionada a la IA de este país, las bases y conocimientos básicos necesarios, así como los principales problemas que se toparán al incursionar en esta área.

En el capítulo I, se propone un digitalizador de imágenes económico y eficiente, se explican las diferentes fases de detección y digitalización de imágenes tales como la captación, la conversión de señales, el almacenamiento de la imagen y las diversas técnicas de iluminación aplicadas a los variados modelos de reconocimiento.

El capítulo II se mueve en el campo del preprocesamiento y segmentación de las imágenes, detallando diversos métodos, los cuales nos proporcionarán las condiciones necesarias para proceder a la siguiente fase que es la extracción de características (capítulo III).

XV

En el capítulo IV se explican a grandes rasgos los dos principales métodos utilizados en el reconocimiento de formas así como las posibles aplicaciones de los sistemas de visión, en el capítulo V se implementan diversos métodos de preprocesamiento y reconocimiento, mostrando las ventajas y deficiencias de los algoritmos probados así como las conclusiones generales.

El sistema desarrollado trabaja únicamente sobre imágenes bidimensionales de escenas específicas, en algunos casos se probó traslapamiento e imágenes ruidosas; todas las pruebas fueron simuladas mediante software, por el hecho de no contar físicamente con el digitalizador de imágenes, solo se trabajó con imágenes binarias por el hecho de la facilidad y simplicidad de su manipulación. Solo en algunos de los programas de reconocimiento se trabajó en modo texto.

CAPITULO I

"DETECCION Y DIGITALIZACION DE IMAGENES"

1.0 INTRODUCCION:

A través de toda su historia, el hombre ha sido un individuo investigador, inquieto por modificar su entorno para beneficio y comodidad propia adaptando su medio lo mejor posible de acuerdo a sus posibilidades, y utilizando su ingenio como principal herramienta; en los tiempos modernos la ciencia avanza a pasos agigantados: economía, medicina, botánica..... electrónica o inteligencia artificial son solo algunas de las muchísimas disciplinas en las cuales se investiga constantemente en todo el mundo, y es precisamente en estas dos últimas disciplinas, donde centraremos nuestro foco de estudio.

El potencial de aplicación de un sistema de visión artificial sólo queda limitado por nuestra imaginación y la tecnología disponible. Inspirado en una antigua inquietud dentro de esta área de la inteligencia artificial que es la robótica, expongo ante ustedes quizá de manera primitiva, pero de acuerdo a nuestras posibilidades un pequeño sistema de detección y digitalización de imágenes el cual les expondré en el transcurso del presente capítulo.

Para lograr nuestro fin me he auxiliado de la herramienta número uno del hombre de los tiempos modernos: "LA COMPUTADORA".

Los dispositivos clásicos estaban limitados pero ayudaban a su operador proporcionándole información en bloques de datos binarios, los cuales era necesario almacenarlos en grandes masas de memoria. Un poco tiempo despues surgieron los "transductores ópticos" que se encargan de captar la imagen analógica y digitalizarla y en sentido contrario convertir los datos digitales que se reciben en señales de video compuesto la cual es posible visualizarse en una pantalla.

Luego, con el desarrollo de la ciencia y la tecnología se ha reducido aún más esta compleja maraña de problemas, al aparecer las memorias de semiconductores y discos magnéticos de alta densidad.

El gran desarrollo alcanzado por estos sistemas ha impulsado simultáneamente al desarrollo de nuevos métodos en el mundo de los datos resultantes.

En nuestro estudio partimos del hecho de que contamos con un ordenador de propósito general al cual adaptaremos para lograr nuestros fines y por lo tanto nuestro sistema no será tan eficiente como uno con características funcionales y estructurales como los computadores de propósito especial para el tratamiento de imágenes y reconocimiento de formas.

1.1 ADQUISICION DE LA IMAGEN (CAPTACION)

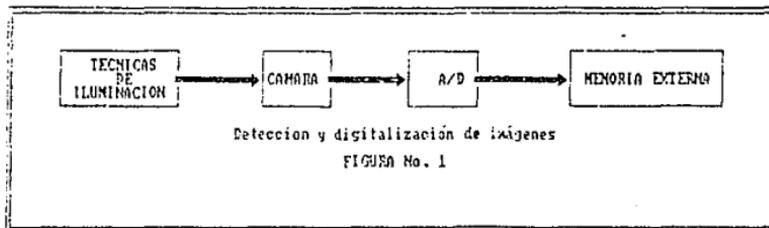
En cualquier sistema de visión típico es posible identificar tres aspectos básicos: a) Detección y Digitalización, b) Procesamiento, y c) Análisis y Aplicación.

La detección o captación de las imágenes nos exige indispensablemente algún tipo de dispositivo que nos permita formar imágenes extrayéndolas de una escena, este dispositivo puede ser una televisión, una videocasetera, una cámara de vídeo, etc. Para nuestro diseño sólo deberá cumplir con el estándar NTSC ya sea a color o blanco y negro.

Este paso inicial consiste en captar la imagen de una escena con la cámara de visión. "La imagen está compuesta por intensidades de luz relativas correspondientes a las diversas partes de la escena en cuestión" [9]. Estas diferentes intensidades de luz son los valores analógicos continuos que nos servirán de muestra y posteriormente convertirse a su correspondiente valor digital.

La digitalización viene siendo el siguiente paso y se consigue por medio de un convertidor analógico/digital (A/D) y posteriormente se pasa al almacenamiento temporal de la imagen formada por una memoria intermedia donde se almacenan temporalmente los datos extraídos de la imagen de la escena, este tercer paso nos sirve para desahogar relativamente la memoria interna de nuestro computador que se podría aprovechar para otras tareas.

A continuación se muestra un diagrama que nos expone a groso-modo la fase de detección y digitalización de imágenes.

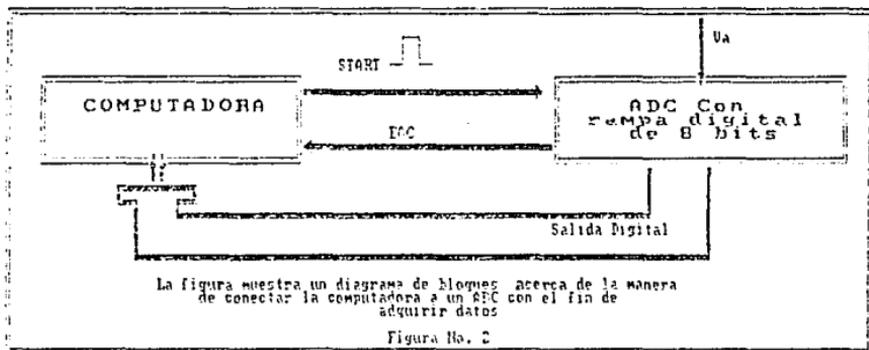


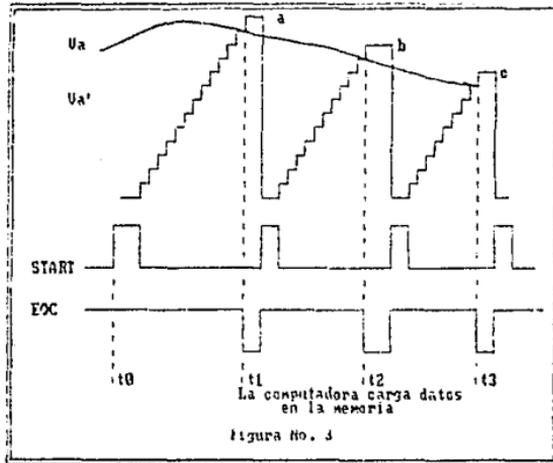
1.2 CONVERSION DE SEÑALES

El proceso de conversión analógico/digital (A/D) implica la trama de una señal de entrada analógica y la generación de una salida que represente la señal de tensión en la memoria digital de una computadora.

La conversión A/D está constituida por tres fases básicas: muestreo, cuantificación y codificación.

Un convertidor A/D toma un voltaje de entrada analógica y después de cierto tiempo produce un código de salida digital que representa la entrada analógica.



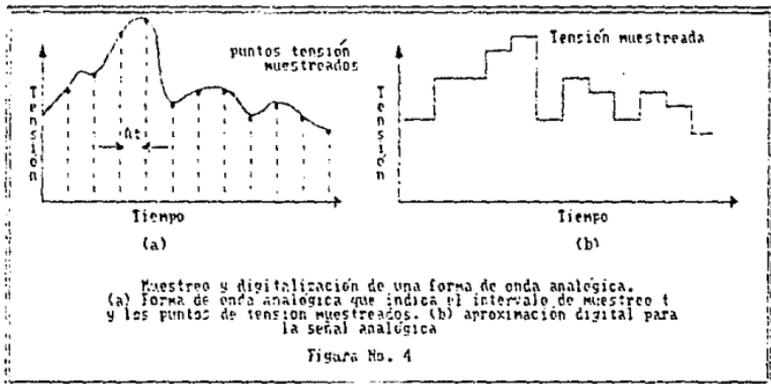


CONVERSION A/D.

La computadora genera las pulsaciones "start" que inician con una conversión A/D. La señal EOC (fin de la conversión) del ADC se alimenta a la computadora. La computadora examina esta señal EOC para así saber cuando se completa la conversión de corriente A/D; después se transfieren los datos digitales de la salida del ADC a la memoria.[54][55].

1.2.1. MUESTREO:

Una señal analógica se muestra periódicamente para obtener una serie de señales analógicas de tiempos discretos.



Estableciendo una frecuencia de muestreo especificada, la señal analógica puede aproximarse mediante las salidas digitales muestreadas. El grado de aproximación de la señal analógica se determina por la frecuencia de muestreo del convertidor A/D. La frecuencia de muestreo debe ser al menos el doble que la más alta frecuencia de señal de video, si se desea reconstruir con exactitud dicha señal.

Para nuestro sistema de visión la señal de video analógica se genera por cada línea de las 256 que comprende la resolución. La capacidad de muestreo del convertidor A/D es 200 nanosegundos (200×10^{-9} s). Esta es la duración del ciclo requerida para completar el proceso de conversión A/D para un elemento de imagen. Utilizando la norma norteamericana de 18.66 milisegundos ($1/60$ s) para explorar la placa de video completa constituida por 256 líneas, determinar la frecuencia de muestreo y el número de pixels que pueden procesarse por línea.

La duración del ciclo por pixel está limitada por el proceso de conversión A/D a 200×10^{-9} s : = 0.2×10^{-6} seg./pixel.

La velocidad de exploración para líneas en la placa frontal de video es de $1/60$ s. = 16.66×10^{-3} s.

En consecuencia, la velocidad de exploración para cada línea (VEPL) es

$$VEPL = (16.66 \times 10^{-3} \text{ s}) / 256 \text{ líneas.} = 65.1 \times 10^{-6} \text{ s / línea.}$$

El número de pixels que pueden procesarse por línea (PPL) por lo tanto:

$$PPL = \frac{65.1 \times 10^{-6} \text{ s/línea}}{0.2 \times 10^{-6} \text{ s/línea}} = 325.5 \text{ pixe ,línea}$$

Ya puesto en práctica esta interfase tendríamos que considerar el tiempo requerido para que el haz electrónico se detenga durante su trama desde una línea a la siguiente. Este tiempo muerto disminuye el número de pixeles utilizados en el sistema de captación de la imagen y de esta manera se ajusta a los 256 pixeles por línea que deseamos.

Para la imagen que esperamos obtener que es de 256×256 pixeles hay un total de 65,536 pixeles a considerar durante el periodo de exploración del monitor (cada pixel se debe procesar mediante alguna técnica de procesamiento de imágenes que estudiará en capitulos posteriores) que se realizará simultáneamente con el periodo de muestreo. Esto exige de una computadora que sea lo suficientemente rápida para que facilite el proceso.

Para una señal de video dada que representa una línea, el número de muestras tomadas determina la resolución horizontal del sistema formador de imágenes. El número total de líneas determina la resolución vertical.

1.2.2 CUANTIFICACION:

Cada nivel de tensión de tiempo muestreado es asignado a un número finito de niveles de amplitud definido. Estos niveles de amplitud predefinidos son característicos para un convertidor A/D particular y están constituidos por un conjunto de valores discretos de niveles de tensión, el número de niveles de cuantificación está definido por:

$$\text{Número de niveles de cuantificación} = 2^n$$

En donde "n" es el número de Bits del convertidor A/D .

Con más Bits la señal es más precisa, el convertidor que proponemos nos permita cuantificar en $2^6 = 64$ niveles de cuantificación diferentes.

1.2.3. CODIFICACION

Le llamamos codificación al proceso de cambiar los niveles de amplitud que se van cuantificando al código digital. Este proceso implica la representación de un nivel de amplitud mediante una secuencia de dígitos binarios. La capacidad del proceso de codificación para distinguir entre los diferentes niveles de amplitud es una función del espacio que exista entre cada nivel de cuantificación. Dado el rango de escala completa de una señal de video analógico, el espaciado de cada nivel está definido por:

$$\text{Espaciado del nivel de cuantificación} = \text{rango escala completa} / 2^n$$

El error de cuantificación resultante del proceso de cuantificación se define como:

$$\text{Error de cuantificación} = \pm 1/2 (\text{espaciado del nivel de cuantificación})$$

[5][7][8]

Una señal de tensión de video continua se ha de convertir en señal discreta. El rango de la señal después de la ampliación es teóricamente de 0 a 5v. El convertidor A/D que recomendamos tiene una capacidad de 6 bits. Trataremos de determinar el número de niveles de cuantificación, la resolución, el espaciado del nivel de cuantificación y el error de cuantificación.

$$\text{Capacidad del convertidor A/D} = 6 \text{ bits.}$$

$$\text{Número de niveles de cuantificación} = 2^6 = 64$$

$$\text{La resolución del convertidor A/D es } 1/64 = 0.015625 \text{ ó } 1.56 \%$$

Para el rango de 5v se tiene :

$$\text{Espaciamiento del nivel de cuantificación} = (5 \text{ V}) / 2^6 = 0.078125 \text{ V.}$$

$$\text{Error de cuantificación} = 1/2 (0.078125 \text{ V}) = 0.0390625 \text{ V}$$

La representación de la señal de tensión en forma binaria implica el proceso de codificación, el cual se realiza asignando la secuencia de dígitos binarios para representar niveles de cuantificación crecientes.

CODIFICACION PARA LA ESCALA DE GRISES

TABLA No.1		
Rango Tensión, V	Número binario	Escala de gris
0	000 000	0 (Negro)
0.0390625	000 001	1 (Gris oscuro)
0.078125	000 010	2
0.15625	000 011	3
:	:	:
:	:	:
4.92187	111 110	62 (Gris Claro)
4.9609375	111 111	63 (Blanco)

1.3 ALMACENAMIENTO DE LA IMAGEN:

Después de la conversión A/D, la imagen se almacena en una memoria RAM estática la cual cubre un doble papel; ser memoria de refresco a los efectos de presentación visual y ser memoria de almacenamiento de información en el proceso de adquisición de imágenes.

Posteriormente la imagen se pasa a la memoria de la computadora, algunos autores le llaman Buffer de cuadro.

Se han desarrollado varias técnicas para adquirir y acceder a las imágenes digitales. En condiciones ideales, es deseable adquirir un cuadro único de datos en tiempo real. Lo ideal sería cuantificar en 8 bits por pixel los cuadros digitales, sin embargo un buffer de 6 bits como el que proponemos es apropiado puesto que el sistema de cámara medio no pueda producir 8 bits de datos exentos de ruido, aunque existen cámaras sofisticadas que lo logran a cambio de un alto precio. Además, el ojo humano solo puede separar eficazmente unos $2^6 \approx 64$ niveles de gris. Para el captador de trama se utiliza una combinación de contadores de filas y columnas que se sincronizan con la exploración del haz electrónico en la cámara. De esta manera es posible direccionar cada posición en la pantalla. Para leer la información almacenada en el buffer, los datos se captan mediante una señal que envía la computadora a la dirección correspondiente a una combinación fila-columna.[34]

En cuanto a la arquitectura de esta memoria RAM, está organizada en 64K celdas de 8 bits cada una, las cuales tienen un corto tiempo de lecturas y escritura.

Dicha memoria es posible realizarla con 2 Integrados 43256-15 (32K x 8 static ram) cuyo tiempo de acceso es de 150 ns. [40]

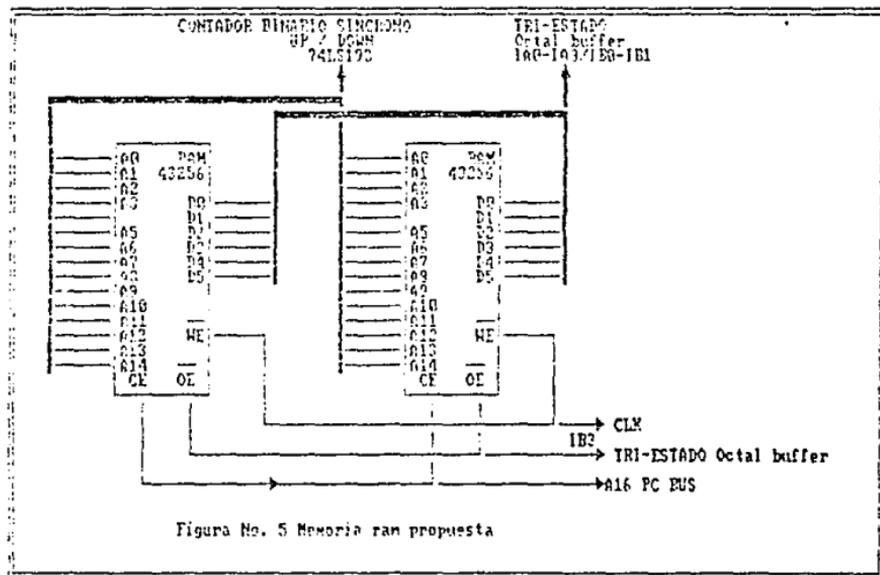


Figura No. 5 Memoria ram propuesta

1.4 TECNICAS DE ILUMINACION:

Es de singular importancia adaptar a todo sistema de visión una técnica de iluminación adecuada a la escena en cuestión, puesto que una buena iluminación de esta, repercute en el efecto que tendrá sobre el nivel de complejidad de los algoritmos de procesamiento de imágenes requeridos.

Una mala iluminación entorpece la tarea de interpretación de una escena, es por esto que debemos de cuidar desde un principio la elección de la técnica de iluminación adecuada para evitarnos cuellos de botella que requerirán análisis más complejos de interpretación y por lo tanto mas tiempo de procesamiento.

"Las técnicas de iluminación adecuadas deben proporcionar un alto contraste y reducir al mínimo las reflexiones espectaculares y las sombras, a no ser que se diseñe específicamente en el sistema". [8]

Existen algunos dispositivos básicos de iluminación utilizados en robótica para la "visión de máquina" y estos se agrupan en las categorías siguientes:

- 1.-Dispositivos de superficie difusa. Ejemplo de iluminadores de superficies difusas son los tubos fluorescentes y mesas de luz.
- 2.-Proyectores de condensador. Un proyector de condensador transforma una fuente de luz en expansión en una fuente de luz en condensación. Se utiliza en la óptica de formación de imágenes.
- 3.-Proyectores puntuales o difusos. Se utilizan para iluminar áreas superficiales.
- 4.-Collimadores. Se utiliza para proporcionar un haz de luz paralelo sobre el suelo.
- 5.-Formadores de imágenes. Tales como proyectores de diapositivas y ampliadoras ópticas, forman una imagen de "Blanco" en el plano del objeto.

Se han desarrollado muchas técnicas para utilizar estos dispositivos de iluminación, la finalidad de las técnicas de iluminación es dirigir el recorrido de la luz desde el dispositivo de iluminación a la cámara, de manera que la escena sea captada lo más nitidamente posible por la cámara.

A continuación se muestra una tabla con diferentes técnicas de iluminación:

Técnica	Función-uso
A. Fuente de luz frontal	
1. Iluminación frontal	Área iluminada de modo que la superficie defina la característica de imagen
2. Iluminación especular (campo oscuro)	Utilizada para el reconocimiento de defectos superficiales (fondo oscuro)
3. Iluminación especular (campo de luz)	Utilizada para el reconocimiento de defectos superficiales; cámara en línea con los rayos reflejados (luz de fondo)
4. Dispositivo formador de imágenes frontal	Aplicaciones de luz estructurada; la luz de las imágenes se superpone sobre la superficie del objeto; haz de luz desplazado como función del espesor
B. Fuente de luz posterior	
1. Iluminación posterior (campo iluminado)	Utiliza un difusor superficial para las características de siluetas; se emplea en la inspección de piezas y en medidas básicas.
2. Iluminación posterior (campo oscuro)	Produce imágenes de alto contraste de utilidad para aplicaciones de gran ampliación
3. Iluminación posterior (colimador)	Produce una fuente de rayos luminosos paralelos tal que las características del objeto no estén en el mismo plano
4. Iluminación desplazada posterior	De utilidad para obtener imágenes destacadas cuando la característica está en un medio transparente.
C. Otros dispositivos diversos	
1. Divisor de haz	Transmite luz a lo largo del mismo eje óptico que el sensor, la ventaja es que puede iluminar objetos de difícil visión
2. Espejo dividido	Similar al divisor de haz, pero más eficaz con exigencias de intensidades más bajas
3. Rodillos de luz y reflexión	La fuente de luz es reflejada por un par de rodillos para

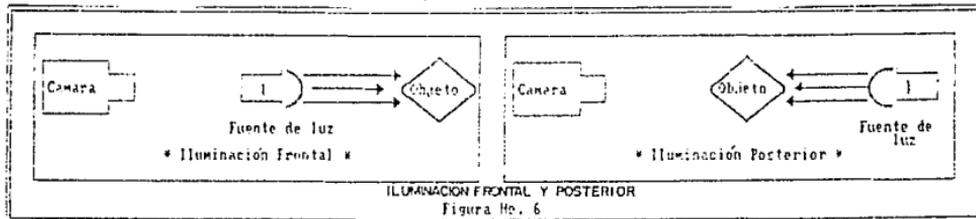
Tabla No.2

TABLERO DE ILUMINACION

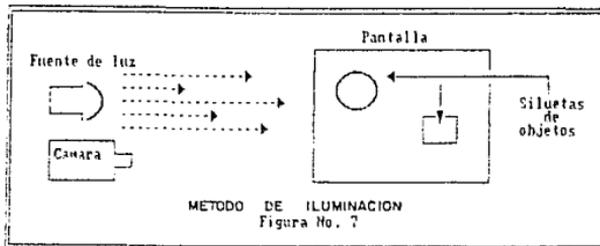
Como se puede observar existen dos técnicas básicas de iluminación utilizadas en visión artificial: iluminación Frontal e Iluminación Posterior.

La iluminación frontal es aquella que se lleva a cabo cuando se sitúa la fuente de luz al mismo lado de la escena que la cámara. Por lo tanto la luz reflejada es la que utiliza la cámara para crear la imagen.

Obtenemos una iluminación posterior cuando la fuente de luz esta situada detrás del objeto de interés y dirigida a la cámara. La imagen captada por la cámara es una silueta del objeto, esta técnica de iluminación se utiliza en aplicaciones donde la silueta del objeto proporciona los datos suficientes para su reconocimiento o cuando existe necesidad de obtener medidas importantes del objeto.



En nuestro sistema utilizamos una técnica de iluminación frontal con una lámpara que proyecta luz blanca sobre la figura que se encuentra en la pantalla, como estas figuras carecen de profundidad evitamos el problema de sombras indeseadas en nuestra escena. Las figuras están hechas de papel negro opaco para evitar reflejos que nos podrían ocasionar problemas a la hora de efectuar el reconocimiento. La malla que utilizamos como pantalla es una superficie blanca y lisa similar a las utilizadas en los estudios fotográficos, de esta manera, la figura hace el contraste que nosotros deseamos con el fondo; Blanco y negro.



1.5 LA INTERFASE (EL DIGITALIZADOR)

A Continuación se plantea la recomendación para su desarrollo de un digitalizador de imágenes adaptable a un slot de expansión de cualquier computadora IBM PC o compatible. La señal de video puede ser extraída de alguna videocasetera ó cámara de video que cumpla con el standar NTSC con salida a color o blanco y negro.

Este digitalizador tiene una resolución de 256 x 256 pixels con 64 niveles de gris, con una velocidad de conversión de 60 cuadros por segundo, esta tarjeta puede estar conectada a un slot de 8 ó 16 bits y se recomienda aplicarla en un sistema con monitor VGA para obtener los mejores resultados puesto que puede desplegar sin inconvenientes los 64 niveles de gris que proporciona la interfase. Sin embargo, es adaptable a cualquier sistema capaz de resolución gráfica, (CGA, HERCULES, EGA) con la condición de posible disminución en la calidad de la imagen (Resolución y tonos de gris). La comunicación a la interfase conectada al slot es por medio de Software.

El digitalizador está compuesto por varios bloques de circuitos íntimamente relacionados que enumeramos a continuación.

I).-CIRCUITO AMPLIFICADOR: Este circuito se encarga de adecuar la señal recibida por el digitalizador a los niveles requeridos por el circuito convertidor de la señal analógica a digital para su posterior procesamiento.

II).-CIRCUITO SINCRONIZADOR: En este circuito se van detectando los pulsos de sincronía de la señal de video que sirven para la identificación posterior del inicio y fin del cuadro de barrido.

III).-CIRCUITO CONVERTIDOR: En este circuito se convierte la señal analógica en digital y a la salida se podrán procesar los datos. Este circuito convertidor está en sincronía con los circuitos de sincronización y memoria. (Ver el punto 1.2 "Conversión de señales").

1.6 CONCLUSIONES

A lo largo del presente capítulo se expone teóricamente la forma de detectar y digitalizar imágenes, muestra diferentes alternativas de dispositivos de captación (1.1) y recomienda técnicas de iluminación para facilitar el procesamiento posterior de la imagen (1.4).

La sección 1.5 propone un dispositivo digitalizador de imágenes que se puede desarrollar a bajo costo, con algunos pequeños inconvenientes técnicos, pero suficiente para proporcionar una imagen de calidad para aplicar técnicas de procesamiento (Capítulo 2); El estudio detallado y la elaboración de este dispositivo queda fuera del objetivo de esta tesis,* por lo que se limitó este capítulo a exponer de manera meramente informativa el desarrollo de este.

* "Dispositivo digitalizador de imágenes" : Tesis complementaria propuesta por el Ing. Rafael Flores Balarezo, egresado de la carrera de Ingeniería en Computación generación 87-91.

CAPITULO II.

SEGMENTACION Y PREPROCESAMIENTO.

2.0.-INTRODUCCION A LA SEGMENTACION Y PREPROCESAMIENTO.

Como ya lo hemos aclarado anteriormente, el objetivo principal de un sistema de visión, es la correcta interpretación y comprensión por parte de la computadora (en nuestro caso) de la imagen obtenida (escena), para poder tomar decisiones apropiadas en situaciones distintas, dependiendo de lo que ocurra a su alrededor.

Puesto que es demasiado fructible, que en muchas ocasiones la imagen no esté en las condiciones propicias como para que la computadora pueda tomar una decisión, es decir que nuestra escena contenga imperfecciones tales como distorsión, ruido, mala o desigual iluminación, partes ocultas o incompletas, sombras indeseadas, desvanecimientos locales, etc., se hace necesario hacer un preprocesamiento de nuestra imagen y así corregir algunos de los inconvenientes antes citados. Para esto existen varias técnicas especiales, algunas demasiado complejas y otras no tanto; por lo que se implementaron algunos algoritmos, quizá algo rudimentarios pero suficientes para alcanzar nuestros propósitos.

2.1 PREPROCESO

Existen básicamente dos técnicas para el preproceso de la imagen,[9] estas son:

- a) **Técnicas de dominio espacial:** Es precisamente en esta técnica donde desarrollamos nuestro preproceso y se refiere al estudio del conjunto de píxeles que componen una imagen y operan directamente sobre ella.
- b) **Técnicas de dominio de frecuencias:** En estas técnicas se trabaja con conjuntos de píxeles complejos, resultado de aplicar la transformada de Fourier a una imagen. Debido a que se requiere gran cantidad de procesamiento no se han popularizado mucho, sin embargo son de suma importancia en análisis de áreas y de objetos en movimiento y es factible aplicarlas cuando se cuenta con computadoras sofisticadas.

2.1.1. ELIMINACION DEL RUIDO (SUAVIZADO)

El suavizado se utiliza para reducir el ruido que puede estar presente en una imagen y puede ser ocasionado por el resultado del muestreo, transmisión, cuantización o perturbaciones en el entorno durante el proceso de obtención de imágenes; nosotros simulamos estos desperfectos y luego tratamos de corregirlos mediante esta técnica.

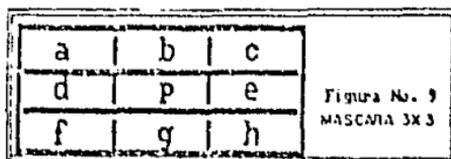
Se puede hacer el suavizado de una imagen utilizando una técnica directa en el dominio espacial llamada "Promediado del Entorno de Vecindad" que consiste en generar una imagen suavizada $g(x,y)$ de nuestra imagen base $f(x,y)$, cuya intensidad para cada punto (x,y) de esta imagen suavizada se obtiene promediando los valores de intensidad de los píxeles de f incluidos en el entorno de vecindad predeterminado de (x,y) . Para nuestro caso esta técnica no es aplicable pues solo estamos trabajando con dos colores: blanco y negro (imagen binaria), y no con niveles de gris.

Las imágenes binarias resultan cuando utilizamos un método de iluminación posterior o luz estructurada como lo vimos en el capítulo uno de ésta tesis, o también de procesamientos tales como la detección de contornos o umbrales, como nosotros trabajamos con imágenes binarias utilizamos un método que algunos autores llaman Suavizado Binario de Imágenes y para explicarlo expondré mi convenio:

- Etiquetar los píxeles oscuros con un 1.
- Etiquetar los píxeles claros con un 0.

En este caso el ruido producirá efectos tales como contornos irregulares, pequeños huecos, esquinas perdidas y puntos aislados.

Para simplificar aún mas este desarrollo nos limitaremos a utilizar una máscara de 3 x 3 que tomaremos como referencia. Observar la figura 9.



Vecinos de p utilizados para suavizar imágenes binarias. Los píxeles oscuros se indican por 1 y los claros por 0.

El procedimiento de suavizado incluye los siguientes puntos:

- 1).- Rellenar los huecos (de un píxel) en zonas oscuras.
- 2).- Rellenar pequeños cortes y muescas en segmentos de lados rectos
- 3).- Elimina los unos aislados.
- 4).- Elimina pequeñas protuberancias en segmentos de lados rectos.
- 5).- Repone los puntos perdidos de las esquinas.

Tomando como base la figura anterior utilizamos la expresión booleana: $B1 = p + b \cdot g \cdot (d + e) + d \cdot e \cdot (b + g)$ que nos corregirá los primeros dos problemas que se señalan en los puntos 1 y 2 de la página anterior.

< . > Representa la función AND.

< + > Representa la función OR

Para alcanzar nuestros fines a un píxel oscuro dentro del área de la máscara se le asigna un 1 lógico y a un píxel claro un 0 lógico. De esta manera si $B1 = 1$, se le asigna un 1 a p y en caso contrario este píxel toma el valor de 0.

La expresión B1 se aplica a todos los píxeles simultáneamente, esto es que se determinará el siguiente valor de cada píxel de la máscara antes de que se actualice cualquiera de ellos.

Ejemplo:

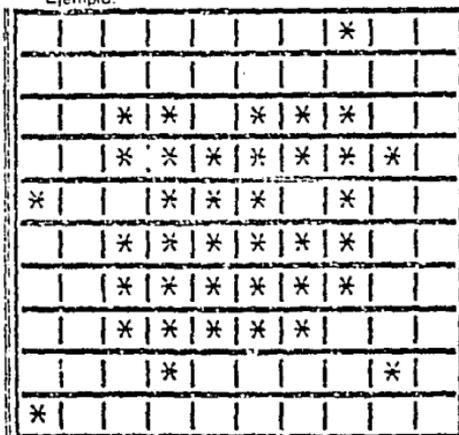


IMAGEN ORIGINAL

Figura No. 18

2.1.2 PERFILADO

Esta técnica de procesamiento de imagen también se le llama realce y consiste precisamente en eso, realzar los contornos de los objetos que aparecen en la escena.

Al contrario que el suavizado, donde se pretendía igualar los niveles de píxeles adyacentes, el perfilado, aumenta las diferencias donde se detecta un cambio brusco de intensidad entre píxeles continuos, que por lo general sucede en los contornos de las imágenes. Estas técnicas de realce de imágenes se diseñan para mejorar la calidad de las imágenes desde el punto de vista de un observador humano.

Se pueden utilizar multitud de técnicas en los problemas de realce, que va desde ampliar los niveles de gris presentes en la imagen, hasta incrementar su contraste separando más esos niveles, pasando por el realce de los bordes de los objetos o la pseudo-coloración que consiste en asignar un color distinto a cada nivel de gris detectado en la imagen, de esta manera es más sencillo reconocer las características de la imagen ya que para el hombre es más fácil reconocer los diferentes colores que los diferentes niveles de gris.

Por otra parte hay técnicas que se utilizan para recuperar información cuando se aplican a imágenes en mal estado o deterioradas. A esto generalmente se le llama restauración de imágenes. "El propósito general de las técnicas de restauración de imágenes consiste en intentar reconstruir o recuperar una imagen degradada usando el reconocimiento que se pueda tener del fenómeno que ha conducido a dicha degradación"[24]. Las técnicas que se usan en el proceso de restauración depende del tipo de degradación que haya sufrido la imagen.

A continuación listamos algunas de las técnicas de realce más comúnmente utilizadas:

1).- Igualación de Histogramas: Trata de equilibrar la distribución de los distintas intensidades de gris; cuando se aplica este proceso suele tener efectos realmente espectaculares sobre el aspecto de la imagen. [9],[10],[24],[25], [34],[35].

Esta técnica está basada en una función de transformación que viene determinada únicamente por el histograma de la imagen de entrada, y está limitado a causa de que su única función es la linealización de histogramas, un proceso que no es aplicable cuando la información que se conoce a priori es la forma del histograma de salida deseado.

2).-Especificación de Histogramas: Consiste en el procesamiento de histogramas desarrollando una solución capaz de generar una imagen tomando un histograma de intensidades especificado con anterioridad y se resume en los siguientes puntos:

a). Igualar los niveles de la imagen original.

b). Especificar la función densidad de probabilidad (FDP) de intensidad deseada y obtener la función de transformación de los niveles de intensidad de la imagen.

c). Aplicar la transformada inversa a los niveles de intensidad de la imagen del histograma igualado que se obtuvo en el paso "a".

Este procedimiento tiene como salida una imagen con la función densidad de probabilidad de intensidad especificada. [9],[10],[24],[25],[34],[35].

3).- Realce Local: Si bien el desarrollo global es adecuado para un realzado completo, a menudo es necesario realzar detalles en áreas pequeñas entonces se crean funciones de transformación que se basan en la distribución de intensidad, u otras propiedades, sobre el entorno de vecindad de cada píxel en una imagen dada.

Es posible aplicar las técnicas de proceso de histogramas al realce local, definiendo un entorno de vecindad de $n \times m$ (máscara) e ir desplazando el centro de la máscara pixel por pixel y calculando el histograma en cada posición y así obtener una función de transformación de igualación de histograma o de especificación, de esta manera se encuentra la intensidad del pixel centrado en el entorno de sus vecinos. Sólo se cambia una fila o columna en cada cambio de región y se actualiza el histograma obtenido en la posición anterior con los nuevos datos introducidos en cada movimiento. Si se utilizaran regiones no solapadas se produce el desagradable efecto de tablero de ajedrez. [9],[10],[24],[25],[34],[35].

4).- La Pseudocoloración: Este procedimiento consiste en asignar en una tabla un color distinto a cada nivel de gris que pueda ser detectado por nuestro sistema de visión, de esta manera se le va asignado a cada pixel analizado su correspondiente color y así se reduce la dificultad que tiene el hombre de diferenciar entre dos niveles de grises asimple vista iguales.[24],[25],[32],[33].

2.2 SEGMENTACION:

Se le llama segmentación a los distintos métodos utilizados para reducir datos.

"La segmentación es el proceso que divide a una escena percibida en sus partes constituyentes u objetos" [10],[11].

La segmentación es uno de los pasos más importantes del procesamiento de imágenes ya que es aquí donde se extraen los objetos de nuestra escena (imagen) para su posterior reconocimiento y análisis.

Su objetivo principal es "agrupar áreas de una imagen con características similares dentro de entidades distintas, que representan partes de la imagen" [8].

En nuestro análisis coincidimos con los estudiosos[8] en agrupar las técnicas de segmentación en la siguiente manera:

- 1) Fijación de umbrales.
- 2) Crecimientos de región.
- 3) Detección de bordes.

La discontinuidad y similitud de imágenes son los principios básicos en que se basan estas técnicas de segmentación.

2.2.1 FIJACION DE UMBRALES:

Esta operación se utiliza para extraer de una imagen en varios niveles de gris, una imagen binaria 0 y 1, en sólo dos niveles: blanco o negro. Esta imagen obtenida se utiliza como entrada a el sistema de reconocimiento.

La umbralización es una operación muy sencilla y se define como:

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

De tal manera que los píxeles en $g(x, y)$ etiquetados como 1 pertenezcan a los objetos y el resto al fondo.

En sí, el gran problema de esta técnica es la elección apropiada de T para que sea cierta la separación entre objetos y fondo.

Para esto se puede considerar a la fijación de umbral como una operación que implica pruebas con respecto a una función T de la siguiente forma:

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

Donde $f(x, y)$ indica la intensidad puntual, $p(x, y)$ indica alguna propiedad local medida en una vecindad definida del punto en consideración, por ejemplo, la intensidad media de un entorno de vecindad centrado en (x, y) .

Cuando T depende sólo de $f(x, y)$, al umbral se le llama global y cuando depende de $f(x, y)$ y $p(x, y)$ se le llama local, si T depende de las coordenadas espaciales x e y , se le llama umbral dinámico.

Si consideramos que un histograma está formado por la suma de dos funciones de densidad de probabilidad, correspondiendo cada una de estas a objetos y fondo respectivamente. Es obvio que este planteamiento solo será de interés cuando los objetos y el fondo son más o menos uniformes de manera que el histograma sea claramente diferenciable en dos modos (bimodal).

En estos casos se puede suponer que:

$$p(z) = P1 \cdot p1(z) + P2 \cdot p2(z)$$

En donde z es una variable aleatoria que representa la intensidad, $P1$ y $P2$ son las probabilidades a priori de que un punto sea objeto o fondo, y $p1()$ y $p2()$ son las densidades de probabilidad de estas dos clases.

El problema así planteado es idéntico al de clasificación de muestras en dos clases. Entonces, utilizando el criterio de clasificación de Bayes con el que se obtiene el menor error de clasificación:

Asignar z a clase 1 si $P1(z) > P2(z)$

Asignar z a clase 2 en caso contrario.

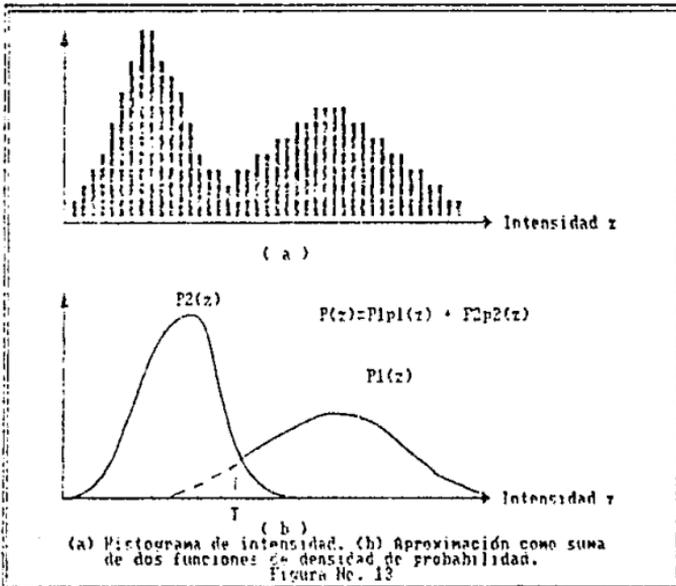
Por lo tanto el umbral T con el que se conseguirá el mínimo error será el valor de z que satisfaga $P1(z) = P2(z)$.

Podemos ahora asignar unas probabilidades a priori a cada clase (si conocemos el problema) o bien parametrizar las funciones $p1()$ y $p2()$ como gaussianas, por ejemplo, con lo que plantearía:

$$P1 p1(T) = P2 p2(T)$$

$$T = T(m1, m2, s1, s2, P1, P2)$$

El procedimiento resultante será reducido a comprobar que el histograma es bimodal para así ajustar cada una de las partes a sendas gaussianas. Una vez hecho esto conoceremos el mejor valor para T .

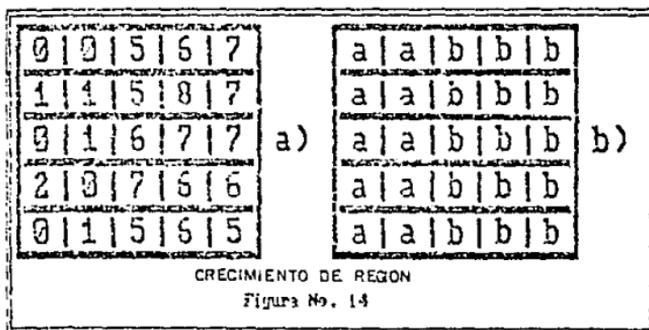


Como hemos visto con anterioridad en nuestro sistema de visión no será necesario hacer este tratamiento puesto que nuestro monitor ya nos da la imagen binaria.[9], [10], [24],[25].

2.2.2 CRECIMIENTO DE REGION:

El crecimiento de regiones es una técnica en la cual se agrupan píxeles o regiones de la escena para formar subregiones más grandes. La forma más sencilla es la agregación o adición de píxeles, que consiste en tomar un grupo de puntos a los cuales se le llama "semilla", los píxeles del entorno que tienen propiedades similares (intensidad, textura o color, por citar algunas).

Para ilustrar este procedimiento más claramente veamos la siguiente figura:



Supongamos que estas celdas son una parte de nuestro monitor y que los números que están dentro de cada una de estas celdas representan el valor de la intensidad del píxel. Si tomamos los puntos con coordenadas (3,2) y (3,4) como semillas, y serán nuestros puntos de inicio y así se obtendrá una segmentación en dos regiones: R1 asociada con la semilla con coordenadas (3,2) y R2 asociada con la semilla de coordenadas (3,4).

La propiedad que se utiliza para incluir un píxel en alguna de las regiones es que la diferencia absoluta entre la intensidad del píxel y la intensidad de la semilla sea menor que un umbral T (los píxels que satisfacen esta propiedad simultáneamente para ambas semillas se asignan arbitrariamente a la región R1).

El resultado que se obtiene utilizando $T = 3$ es el que se observa en la figura 14(b).

En este caso, el crecimiento de región da como resultado 2 regiones, donde cada punto de R1 está representado por una "a" y los puntos de R2 con "b".

• Si hubiésemos tomado cualquier punto como inicial de cada una de las dos regiones resultantes obtendríamos el mismo resultado. Si se elige $T = 8$, se obtiene una región única, como lo ilustra la siguiente figura:



Los problemas que se presentan con el ejemplo anterior es la selección de las semillas iniciales de manera que representen adecuadamente las regiones de interés y el otro es la selección de las propiedades adecuadas para asignar a los puntos a las diversas regiones durante el proceso de crecimiento.

Cuando no tenemos información a priori, se suele evaluar en cada pixel el mismo conjunto de propiedades que se usan al final para asignar los pixeles a regiones durante el proceso de crecimiento. Si al terminar este cálculo tenemos como resultado grupos de valores, tomamos como semillas a los pixeles cuyas propiedades les hagan situarse cerca del centro de estos grupos; si hicieramos un histograma de intensidades del ejemplo anterior veríamos que los puntos de intensidades entre 1 y 7 son los más predominantes.

Para seleccionar un criterio apropiado se debe de tomar en cuenta tanto el problema a consideración como el tipo de datos de imagen disponible, es por eso que para nuestro caso este problema se incrementa puesto que sólo contamos con dos colores y no podemos sacar niveles de intensidad puesto que son fijos, sin embargo utilizamos un proceso de crecimiento de región agregando pixels en los bordes de las imágenes. Normalmente en el análisis de regiones se debe usar un conjunto de descriptores que se basen en propiedades especiales (textura y momentos por ejemplo) e intensidad de una sola imagen.

Si sólo se usan descriptores se puede llevar a resultados equivocados si no se usa la información de conectividad en el proceso de crecimiento de regiones. Supongamos una disposición aleatoria de pixeles con sólo 3 valores de intensidad. La agrupación de pixeles con el mismo valor de intensidad para formar una región, sin tomar en cuenta la conectividad nos llevaría a una segmentación sin significado en este tratamiento.

2.2.3 DETECCION DE BORDES.

La detección de bordes es uno de los pasos iniciales del procesamiento de imágenes y se hace antes de muchos algoritmos de identificación de objetos.

La idea central en un proceso de extracción de bordes, es la siguiente: "Un borde se manifiesta en la imagen, como una variación brusca del nivel de grises en un entorno determinado"[19].

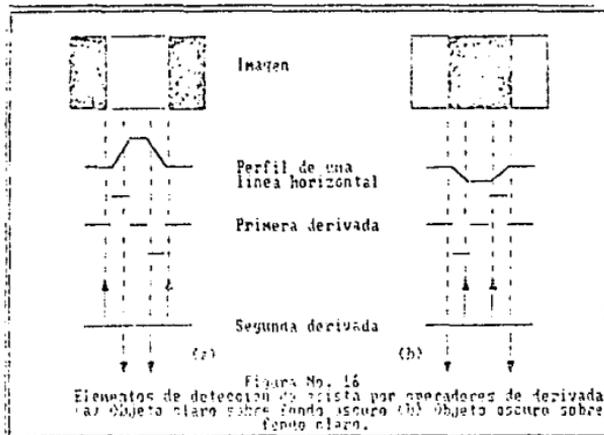
Con esta definición se presentan los siguientes problemas:

- 1). ¿ Qué tan brusca debe de ser esta variación para decidir la existencia o no de un borde ?
- 2). ¿ Se deberá considerar igual este umbral de variación para toda la imagen, o adaptarse dinámicamente ?
- 3). ¿ De qué tamaño deberá ser el entorno a considerar ?

Los estudios anteriores realizados en esta técnica dicen que se pueden realizar en dos dominios distintos: en el de la frecuencia y en el de espacio. En el dominio de la frecuencia, básicamente se limita a un filtro pasa altas, que hace desaparecer las zonas de la imagen con variaciones suaves y realiza donde hay variaciones mas bruscas. El inconveniente de éste método es que no se obtienen los resultados esperados cuando hay presencia de ruido en la imagen; esto se explica por el hecho de que los bordes son un fenómeno muy local, y cualquier transformación realizada en dominio de la frecuencia afecta la imagen de forma global, por lo que no se tienen resultados satisfactorios.

En el dominio del espacio el análisis es de la siguiente manera: Se sitúa sobre el pixel a estudiar, y se obtiene información sobre su nivel de gris y los pixels de su entorno, de esta manera lo clasificamos como pixel perteneciente o no a un borde.

Por lo general la mayoría de las técnicas de detección de bordes se basa en el cálculo de la derivada local.



La parte a) de esta figura muestra una imagen de un objeto claro sobre fondo oscuro, el perfil de intensidad que se obtiene mediante una exploración horizontal de la imagen y la primera y segunda derivada del perfil.

En el perfil se observa que los bordes (transición de oscuro a claro) se representa como una rampa en lugar de un cambio brusco de intensidad. Esto se hace así por el hecho de que los bordes de una imagen digital suelen estar ligeramente desdibujados después del muestreo.

Cuando se representa un borde de esta manera su primera derivada es cero en todas las regiones de intensidad constante y conserva un valor constante en toda la transición de intensidad. La segunda derivada, es cero en todos los puntos, con la excepción del inicio y final de una transición de intensidad.

El valor de la primera derivada se utiliza para detectar la presencia de un borde y el signo de la segunda derivada nos determina si un pixel de borde se encuentra en lado oscuro (fondo) o en el lado claro (objeto) del borde. Por ejemplo, el signo de la segunda derivada en la figura "a", es positivo para los pixels que se encuentran en el lado oscuro de las partes anteriores y posteriores de los objetos, mientras que el signo es negativo para los pixels que se encuentran en los lados iluminados de estos bordes. La explicación es similar para el caso "b": objeto oscuro sobre fondo iluminado.

Existen otros métodos para la detección de bordes tales como los de operadores gradiente y operadores Laplacianos, así como los métodos de Roberts, Sobel y Kirsch [19],[20],[24],[25],[27].

CAPITULO III

EXTRACCION DE CARACTERISTICAS (DESCRIPCION)

3.0 .-INTRODUCCION A LA EXTRAC- CION DE CARACTERISTICAS

Al igual que el ser humano, la computadora, para poder decidir entre una alternativa y otra a la hora de emitir un juicio tiene que ser previamente alimentada con cierta información especial acerca del objeto a reconocer (en nuestro caso), y de esta manera "soplarle" o darle "tips" que la auxiliarán a emitir una decisión. Para poder proporcionar esta "información", es necesario utilizar técnicas independientes para cada fin, de extracción de características.

Una característica, en sistemas de visión, "Es un parámetro único que permite la facilidad de comparación e identificación" [8].

Como se mencionó anteriormente, el problema de "describir" consiste en extraer características de un objeto para reconocerlo. Por lo general los descriptores deben de ser independientes del tamaño, la localización y orientación del objeto y deben contener suficiente información de discriminación para distinguir un objeto de otro. Los descriptores que se toman afectarán tanto a la complejidad de los algoritmos de reconocimiento como al rendimiento.

Las técnicas disponibles para extraer valores de características en imágenes bidimensionales se clasifican por lo general en las que se refieren a las características de contornos y las que trabajan con características de área.

La variedad de características se pueden utilizar para identificar un objeto o parte del mismo y también para determinar la localización y/u orientación.

Por ejemplo, se pueden utilizar técnicas de crecimiento de región para determinar el área de una imagen; y para el perímetro o contorno que encierra un área específica se puede determinar indicando la diferencia en la intensidad de un pixel en el contorno y con solo contar los pixels de la región segmentada que son adyacentes a los pixels que no pertenecen a la región, es decir, al otro lado del contorno, y obtenemos esta medida.

La idea en un sistema de reconocimiento es que no dependa del objeto que se exponga a la cámara, con una relación conocida y fija, como lo es nuestro caso.

A continuación se expone una tabla con las características y medidas para la identificación de un objeto bidimensional [8].

TABLA 3: CARACTERÍSTICAS BÁSICAS Y MEDIDAS PARA LA IDENTIFICACION DE OBJETOS BIDIMENSIONALES.

*Nivel de gris (máximo, medio, mínimo).

*Área.

*Longitud del perímetro.

*Diámetro.

*Rectángulo de cierre mínimo.

*Centro de gravedad: Para todos los píxeles (n) en una región donde cada píxel está especificado por las coordenadas (x,y), las coordenadas "x" e "y" del centro de gravedad se definen como : $C.G.x = 1/n(E_x \cdot x)$ $C.G.y = 1/n(E_y \cdot y)$

*Exentricidad: Una medida de «elongación». Existen varias medidas de las cuales la más simple es : Longitud máxima de cuerda A

$$\text{Exentricidad} = \frac{\text{Longitud máxima de cuerda A}}{\text{Longitud máxima de cuerda B}}$$

donde la longitud máxima de cuerda B se elige perpendicular a A.

*Relación de aspecto: La relación longitud-anchura de un rectángulo que rodea al objeto. Un objetivo es encontrar el rectángulo que da la relación de aspecto mínimo.

*Delgadez: Es la medida de lo delgado que es un objeto. Dos de las relaciones que se utilizan son :

$$(a) \text{ Delgadez} = \frac{(\text{perímetro})^2}{\text{área}}$$

Esta se conoce también como su compacidad.

$$(b) \text{ Delgadez} = \frac{\text{diámetro}}{\text{área}}$$

*El diámetro de un objeto, sin considerar su forma, es la distancia máxima obtenible entre dos puntos situados en el contorno de un objeto.

*Agujeros: Número de agujeros de un objeto.

*Momentos: Dada una región, R, y las coordenadas de los puntos (x,y) en o sobre el contorno de la región, el momento de orden pq-ésimo de la imagen de la región viene dado por:

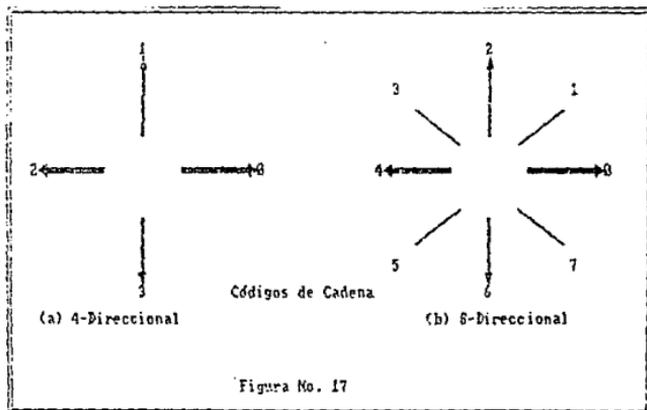
$$M_{pq} = \sum x^p y^q$$

3.1.- DESCRIPTORES DE FRONTERA

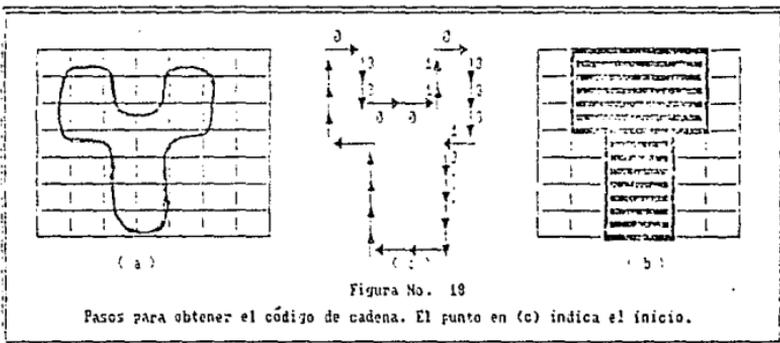
Los descriptores de frontera son técnicas utilizadas en la extracción de características de imágenes de objetos que se basan solamente en la forma externa (silueta) del objeto, sin tomar en cuenta ninguna otra característica, por lo general los contornos de los objetos nos proporcionan mayor información acerca de lo que se está observando, a continuación expondremos de manera informativa algunas de las técnicas utilizadas para describir fronteras.

3.1.1.- CODIGOS DE CADENA

Se usan para representar una frontera como un conjunto de segmentos con longitud y dirección especificadas. Para ilustrar, esta representación se establece sobre una cuadrícula rectangular usando conectividad 4 u 8.



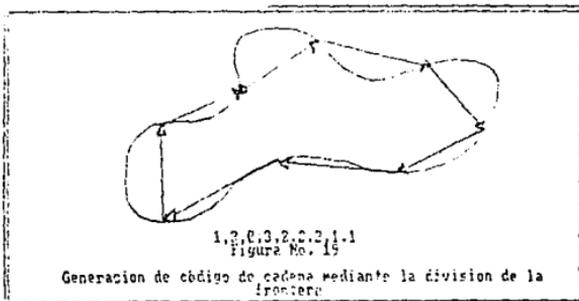
La longitud de los segmentos está dada por la resolución que demos en la cuadrícula y las direcciones dependerán del código que hallamos elegido. Con 2 bits representamos el código 4 y con 3 bits el código 8, aunque se pueden escoger códigos de cadenas con más direcciones, estos son los más utilizados.



Pasos para obtener el código de cadena de una frontera dada:

- 1.- Seleccionar un tamaño de cuadrícula (a).
- 2.- Si una célula incluye más de una cantidad determinada (un 50 %) de área interior a la frontera, se le asigna un 1 a esa célula y un 0 en caso contrario (b).
- 3.- Se codifica la frontera entre dos regiones usando los códigos de dirección. Para la figura anterior se utilizó un código de cadena 4-direccional (c).

Existen otras maneras de utilizar los códigos de cadena, como la generación de códigos de cadena mediante la división de frontera que consiste en dividir la frontera en segmentos de igual longitud (significa que cada segmento tendrá el mismo número de píxeles), y conectar los segmentos de cada segmento mediante una línea recta y asignar a cada línea la dirección más cercana a una de las direcciones permitidas del código de cadena que estemos utilizando.



El código de cadena depende del punto de inicio y por lo tanto se normaliza mediante un sencillo procedimiento:

Dado un código de cadena obtenido, comenzando en un punto arbitrario, se le trata como una secuencia circular de números de dirección y se redefine el punto de inicio de manera tal que la secuencia de números resultante forme un entero de magnitud mínima.

También se normaliza por rotación, usando la primera diferencia del código de cadena en vez del código. La diferencia se calcula contando en sentido horario el número de direcciones que separan dos elementos adyacentes del código. Por ejemplo, la primera diferencia del código de cadena de 4 direcciones 10103322 es 3133030.

Si se trata el código como una secuencia circular, entonces el primer elemento de la diferencia se calcula usando la transición entre el primer y el último componente de la cadena. Entoces el resultado sería 33133030.

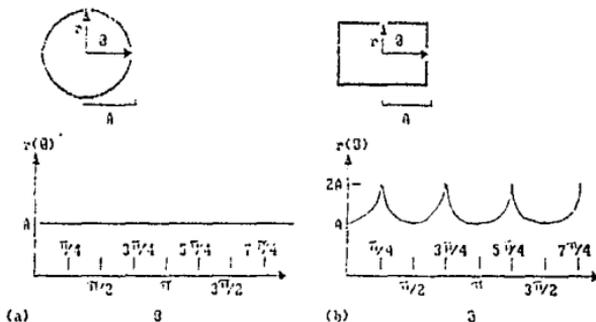
Se puede normalizar por tamaño dividiendo todas las fronteras del objeto en el mismo número de segmentos iguales y ajustando las longitudes de los segmentos de código para que encajen en éstas subdivisiones.

Las normalizaciones anteriores son exactas si las fronteras son constantes respecto a la rotación y al cambio de escala.

3.1.2.- SIGNATURAS

"Una signatura es una representación funcional unidimensional de una frontera" [11].

La forma más sencilla de obtener una signatura, es representar la distancia desde el centro hasta la frontera como una función de ángulo. Las signaturas creadas mediante este método dependen del tamaño y del punto de comienzo.



Signaturas de distancia respecto al ángulo. En (a), $r(\theta)$ es constante mientras que en (b), $r(\theta) = a \text{ seg } \theta$.

Figura No. 2R

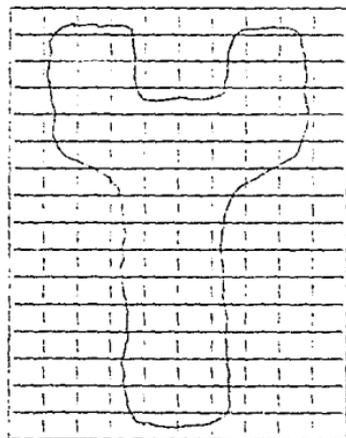
Se hace una normalización de tamaño haciendo que la curva $r(\cdot)$ tenga el valor máximo de una unidad. El problema del punto de inicio queda resuelto obteniendo el código de cadena de la frontera y usando el procedimiento que vimos en la sección anterior.

Existen otros métodos y procedimientos para obtener signaturas (Ambler y Col., Nahin, K.S. Fu) [9],[10],[11].

3.1.3 .-APROXIMACIONES POLIGONALES.

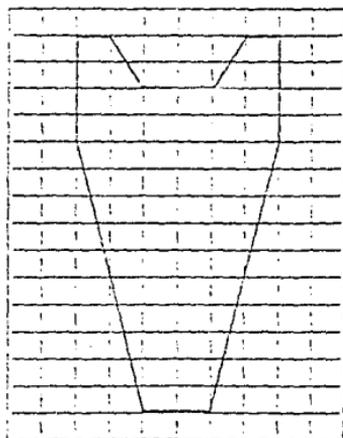
El objetivo de las aproximaciones poligonales es " Captar la esencia de la forma de la frontera con el menor número de segmentos poligonales "[11]. Aunque existen algunos métodos complejos que consumen mucho tiempo máquina, también los hay sencillos y con bajo consumo de tiempo.

Una frontera digital se puede aproximar con una exactitud arbitraria mediante un polígono; para una curva cerrada la aproximación es exacta cuando el número del segmento del polígono es igual al número de puntos de la frontera de forma que cada par de puntos adyacentes defina un segmento de polígono.



(a)

Frontera del objeto encerrada por células.



(b)

Polígono de perímetro mínimo.

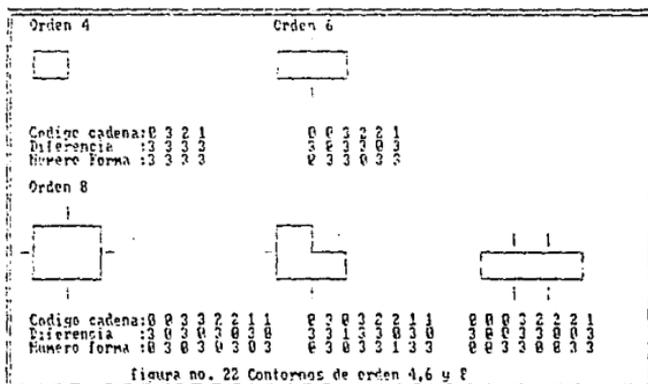
Figura No. 21

Tenemos un conjunto de celdas concatenadas, las cuales contienen una frontera (a), observamos este dibujo como dos paredes, una correspondiente a la frontera externa y otra a la interna de la ordenación de las celdas. Supongamos a la frontera del objeto como si fuera de plastilina y estuviera contenida en estas celdas. Si hacemos que la plastilina se encoja, toma el aspecto de la figura (b) obteniendo así un polígono de perímetro mínimo que encaja en la geometría establecida por la ordenación de las celdas. Si las celdas se toman de forma que cada una de ellas abarque a un solo punto de frontera, el error en cada celda entre la frontera original y la aproximación de la plastilina será un máximo de $2d$, donde d es la distancia entre píxeles. El error se puede reducir a la mitad forzando a que cada celda esté centrada en su píxel correspondiente.

Existen otras técnicas que se aplican para la extracción de características las cuales se pueden estudiar en la referencia [11].

3.1.4.- NUMERO DE CONTORNOS

Quando se codifica una cadena de frontera se observan varias diferencias que dependen del punto inicial, el número de contorno de una frontera, basándose en el código 4-direccional que vimos en los descriptores de frontera se define como la primera diferencia de menor valor. El orden n de un número de contornos se obtiene como el número de dígitos de su representación. El orden n es par para las fronteras cerradas, y su valor nos da el número de posibles contornos diferentes.



En las figuras se observan los contornos de orden 4, 6 y 8, junto con sus representaciones en código de cadena, las primeras diferencias se calculan tratando los códigos de cadena como secuencias circulares de la forma vista anteriormente.

Aunque la primera diferencia de un código de cadena es independiente de la rotación, la frontera codificada en general dependerá de la orientación de la cuadrícula de codificación (Figura 17(a) Descriptores de frontera). Para normalizar la orientación de la cuadrícula se utiliza el siguiente método: El eje mayor de una frontera es el segmento que une los puntos más lejanos de la frontera. El eje menor es perpendicular al eje mayor y de longitud tal que sea posible formar un rectángulo que encierre a la frontera. La excentricidad de la frontera se forma tomando el cociente entre la longitud del eje mayor y el eje menor. Y el rectángulo básico es el que se forma cuando se encierra la frontera. Casi siempre se obtiene un solo número de contorno cuando se alinea la cuadrícula del código de cadena con los lados del rectángulo básico.

Cuando se conoce el orden del contorno deseado, se encuentra el rectángulo de orden n que tenga la excentricidad más cercana a la del rectángulo básico y se usa este nuevo rectángulo para establecer el tamaño de la cuadrícula a utilizar.

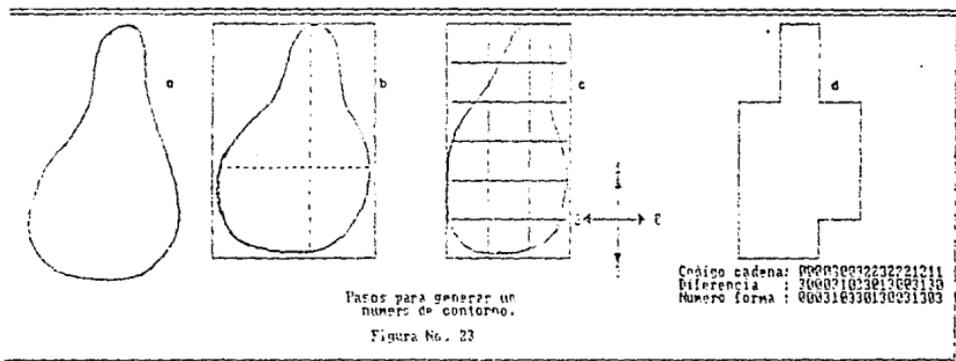
Por ejemplo, si $n = 12$, todos los rectángulos de orden 12 (perímetro = 12) son 2×4 , 3×3 y 1×5 . Si la excentricidad del rectángulo básico de una frontera dada, seleccionaremos una cuadrícula 2×4 contrada en el rectángulo básico y se procede a obtener el código de cadena.

Los números de contornos se obtienen de las primeras diferencias de este código.

Aunque el orden del número de contorno resultante suele ser igual a n debido al método usado para seleccionar el tamaño de cuadrícula, las fronteras con depresiones comparables a este tamaño de cuadrícula nos darán lugar a veces, a números con contorno de orden mayor que n .

En estos casos, se usa un rectángulo de orden menor que n y se repite el procedimiento, hasta tener un número de contorno resultante igual a n .

Supongamos el siguiente ejemplo:



PASOS PARA GENERAR EL NÚMERO DE CONTORNOS

Para la frontera mostrada en la figura (a) $n = 18$, los pasos para obtener un número de contorno de este orden:

- 1) Hallar el rectángulo básico (b).
- 2) El rectángulo mas cercano de orden 18 es un rectángulo de 3×6 , entonces se divide el rectángulo básico como muestra la figura (c).
- 3) Se toman las direcciones del código de cadena alineadas con la cuadrícula resultante.
- 4) Se obtiene el código de cadena y se usa su primera diferencia para evaluar el número de contorno, como se muestra en la figura (d).

3.2 DESCRIPTORES DE REGION

Una región se describe tanto por la forma de su frontera como por sus características internas. Así pues el área de una región se puede describir como el número de píxeles contenidos en su frontera. Esto es útil cuando tenemos imágenes fijas y a una distancia constante de la cámara.

El eje mayor y menor se definen de igual manera que como los definimos en la sección de número de contornos y son útiles para obtener la orientación de un objeto. La excentricidad también es un descriptor global importante de la forma del objeto. El perímetro como descriptor de región se usa para obtener la compatibilidad de una región que se define como $\text{perímetro}^2 / \text{área}$. Una región conectada será aquella en que todos los pares de puntos se puedan conectar mediante una curva que se pueda incluir en la región. Cuando en un conjunto de regiones conectadas alguna, tiene huecos, se considera el número de Euler como descriptor y se define como el "número de regiones conectadas menos el número de huecos".

También existen descriptores de región un poco más complicados como los descriptores de textura; que aunque no hay una definición formal se le relaciona como una medida de propiedades como la suavidad, rugosidad y regularidad. Para hacer descripción de texturas se le entoca desde dos puntos diferentes, el estadístico y el estructural. Cuando se usan métodos estadísticos se obtienen características de textura tales como la suavidad, rugosidad y granulosidad, en las técnicas estructurales se estudia la disposición de las formas básicas de la imagen. Para mayor información [11]. Existen otros descriptores de región como esqueletos de región o momentos invariantes. [7][11][12][21].

CAPITULO IV.

RECONOCIMIENTO DE FORMAS

4.0 INTRODUCCION AL RECONOCIMIENTO DE FORMAS

El siguiente paso de nuestro estudio es identificar el objeto que la imagen representa, para poder efectuar el reconocimiento de los objetos, es necesario efectuar comparaciones de las características extraídas de la imagen mediante alguno de los métodos existentes (capítulo III), con las características extraídas de los modelos previamente memorizados.

Existen diversas técnicas o algoritmos de reconocimiento utilizadas actualmente en sistemas de visión artificial, las cuales se clasifican en dos categorías principales:

- 1.- Métodos de decisión teórica o de coincidencia de plantilla.
- 2.- Métodos estructurales

Los métodos de decisión teórica se basan en descripciones cuantitativas mientras que los métodos estructurales en descripciones simbólicas y sus relaciones, y se estudian mas a fondo en el siguiente punto del presente capítulo.

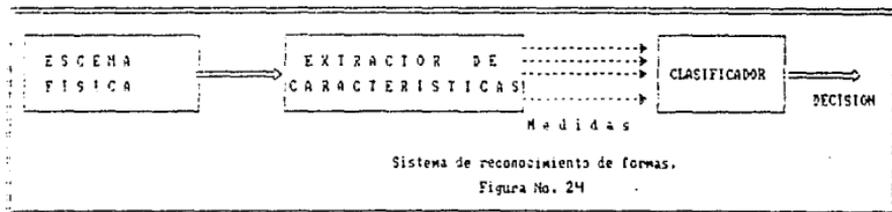
El objetivo principal de éstas técnicas utilizadas en el reconocimiento de formas es "dotar a ciertas máquinas de una capacidad esencial de los animales: poder discriminar el carácter o la composición de una escena física" [29] que se presente en cierto momento ante los sensores adecuados y así esta máquina emita un juicio y actúe según el resultado obtenido.

Mientras mas compleja sea la escena más difícil se va haciendo el reconocimiento de formas y es aquí donde varía la técnica de reconocimiento a utilizar; de esta manera, cuando existe una escena física compleja se recomienda utilizar métodos estructurales y para los casos menos difíciles métodos de coincidencia de plantilla, clasificación ó decisión teórica según los llaman los diferentes investigadores de área.

A continuación se listan las cuatro consideraciones fundamentales que justifican la gran importancia del reconocimiento de formas.[29]

- 1) A la máquina no la afectan factores psicofísicos que pueden afectar a las personas.
- 2) Puede trabajar con magnitudes < no sensibles >, si se cuenta con los sensores adecuados.
- 3) Capacidad de desenvolverse en medios no vitales para el hombre.
- 4) Estas técnicas constituyen el primer paso en el camino de simplificación de las comunicaciones hombre máquina.

Por lo general, los sistemas de reconocimiento de formas están compuestos por un selector de características formado por los sensores que son los encargados de extraer de la escena los rasgos de interés para su estudio, los cuales son procesados por un mecanismo clasificador que decide qué clase de escena u objeto se está captando.



4.1 METODOS.

Los métodos de decisión teórica o de coincidencia de plantilla son un grupo de técnicas estadísticas generales de reconocimiento de modelos que sirven para clasificar objetos en una imagen dentro de categorías predeterminadas.

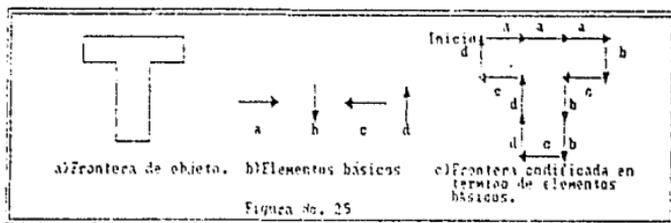
El principal problema de éste primer método es comparar el elemento con un conjunto de características del modelo almacenado, definido como modelo base plantilla. Este modelo es detectado (capítulo I) procesado (capítulo II) y se le han extraído características (capítulo III) que se encuentran almacenadas en algún dispositivo de memoria. Si no contamos con un gran número de modelos en nuestro almacén no es ningún problema; pero en caso contrario, que tengamos demasiados, la complejidad aumenta.

Este método se basa en el uso de un número suficiente de características para minimizar la frecuencia de errores en el proceso de clasificación. Las características del elemento de la imagen (por ejemplo: su área, diámetro, número de ángulos, etc.) se comparan con los valores correspondientes previamente almacenados en las distintas plantillas. Cuando se encuentra una coincidencia dentro del rango que marca el parámetro estadístico que se este utilizando en el proceso de comparación, entonces el elemento ha sido clasificado de la forma adecuada.

La diferencia entre los métodos de decisión teórica y los estructurales es que los métodos de decisión usan patrones de carácter cuantitativo, ignorando todas las relaciones geométricas inherentes a la forma del objeto y los métodos estructurales por el contrario, realizan la discriminación de objetos utilizando estas relaciones.

Las técnicas estructurales de reconocimiento de modelos como mencionamos anteriormente, consideran las relaciones entre las características o bordes de un objeto. Por citar un ejemplo, si podemos subdividir la imagen de un objeto en cuatro líneas (las líneas reciben el nombre de primitivas) conectadas en sus puntos extremos y las líneas que se conectan forman ángulos rectos, entonces el objeto es un rectángulo. A ésta técnica se lo llama método sintáctico, y es la técnica más utilizada.

El objetivo principal del reconocimiento estructural es la descomposición de un objeto en patrones básicos. Para ilustrar esta idea observamos la siguiente figura.



DESCOMPOSICION DE UN OBJETO EN PATRONES BASICOS

En la parte (a) se muestra la frontera de un simple objeto con forma de T, en la parte (b) se muestra un conjunto de elementos básicos con una longitud y una dirección determinadas. En la parte (c) tenemos la frontera (a) codificada, esto se logra comenzando por la parte superior izquierda, recorriendo la frontera en el sentido horario e identificando los elementos básicos (b) conforme se van presentando en el transcurso del recorrido; de esta manera es posible representar la frontera de este objeto como una cadena de caracteres equivalentes: `aaabccbbddcd`. Y así la estructura del objeto queda establecida en términos de la longitud, dirección y orden en que aparecen estos elementos básicos y lograr esta representación particular.

Existen varios métodos estructurales para el reconocimiento de formas como la "correspondencia de números de contornos", [11],[8] "la correspondencia de cadena", [11] o los "métodos sintácticos" [15],[7]; éste último es el más utilizado como reconocimiento estructural.

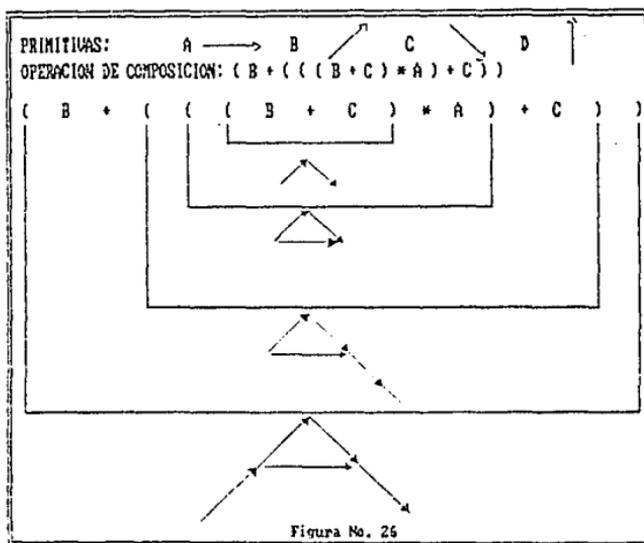
El método de correspondencia de números de contornos se basa en comparaciones de fronteras de objetos que han sido descritas de acuerdo al número de contornos que fueron localizados en ella, tomando como referencia el grado de similitud entre dos fronteras

Los métodos sintácticos se basan en la especificación de elementos básicos de patrones estructurales y conjuntos de reglas (gramática) que nos definen la manera de conexión entre los diferentes elementos que la constituyen.

Estos métodos se realizan mediante representaciones jerárquicas (arborescentes) de objetos en términos de subobjetos más sencillos, los cuales pueden a su vez ser descritos estructuralmente en términos de subobjetos más simples todavía. Para esta construcción de objetos a partir de subobjetos se utilizan operaciones de composición, de manera similar a como se construye una frase a partir de palabras mediante reglas sintácticas y éstas a su vez, a partir de caracteres concatenados mediante reglas ortográficas. Las operaciones de composición se expresan en términos de operaciones matemáticas.

Los subobjetos más simples, que ya no pueden ser descritos estructuralmente, se les llama primitivas, y a las <frases> posibles de objetos, en función de la primitivas y de las operaciones de composición, se denomina, "lenguaje de descripción de formas", la gramática, es el conjunto de reglas que definen este lenguaje.

Así pues en el siguiente ejemplo podemos ver el lenguaje de descripción sintáctica utilizado para describir el carácter "A".



O de una manera mas general:

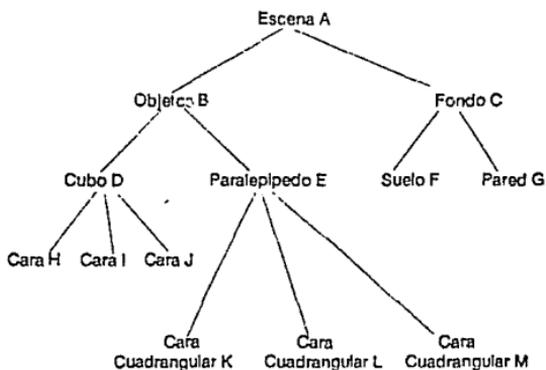
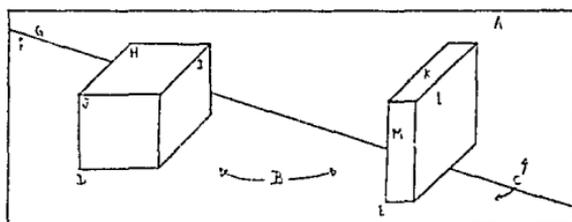


Figura 27

Este tipo de representación arborescente es la que comunmente se usa en el análisis sintáctico.

4.2 PROBLEMAS COMUNES EN EL RECONOCIMIENTO.

Sin duda el principal problema del reconocimiento es la identificación correcta de objetos o de características que forman la imagen. Existen muchísimos otros, como la del reconocimiento de objetos superpuestos o traslapados, el problema aquí radica en que si la computadora no cuenta con un software lo suficientemente poderoso no podrá decidir que objeto está delante de otro; por lo general las pistas que podremos obtener cuando tenemos este tipo de problema, nos las proporcionan las sombras y las diferencias de tonos, sin embargo en un sistema que trabaja con imágenes binarias, no es posible resolver este problema de manera sencilla, por lo que hay que implementar un software que contenga algún algoritmo capaz de resolver esta situación.

Otro problema que se puede presentar es el de imágenes parciales o truncadas. Por ejemplo si nuestra computadora ha sido alimentada con la información referente a un triángulo: tres caras y tres vértices; y si la computadora sólo ve un triángulo con un vértice oculto (por otro objeto o fuera del campo de visión) ¿Cómo podrá la computadora saber que lo que está viendo es un triángulo si ha desaparecido un vértice?

Este ejemplo anterior lo ilustran varios investigadores del área y lo enfrentan de distintas formas, pero quizás en la que mas coincidan es la que llaman "ilusión controlada" que se asemeja a la manera de actuar del ojo humano, suponiendo que lo que se está viendo es un triángulo trata de constatarlo de diferentes formas partiendo de esta suposición inicial, por ejemplo, comprobar que si las líneas del borde de la figura se prolongarán tras la parte oculta de la escena, colidieran en un punto formando entre ellas un ángulo agudo.

El reconocer distintas clases de objetos -que una silla es una silla o que un cuadro es un cuadro es otro problema al que nos enfrentamos los que tratamos de programar la computadora que resuelva este tipo de inconvenientes.

Es más sencillo hacer que la computadora reconozca un objeto específico que hacerla reconocer objetos dentro de una cierta clasificación.

De esta manera si restringimos a solo ciertos objetos específicos con estrictas limitaciones, nuestra escena perdemos la cobertura de todas las posibles variaciones que en un momento dado se puede presentar.

Otro obstáculo que se nos puede interponer en el proceso de reconocimiento es de las ilusiones ópticas, por ejemplo, nosotros los humanos estamos acostumbrados al efecto al efecto de pequeñez relativa que nos produce la vista cuando observamos un objeto lejano, sin embargo en un sistema de visión artificial, de por sí menos perfecto que el humano, interpretará los objetos alejados como pequeños.

Sin embargo hay algunos tipos de ilusiones ópticas que nos confunden a los humanos y que no afectarían a la computadora al efectuar el análisis de esa misma imagen. Por ejemplo observemos la siguiente figura:

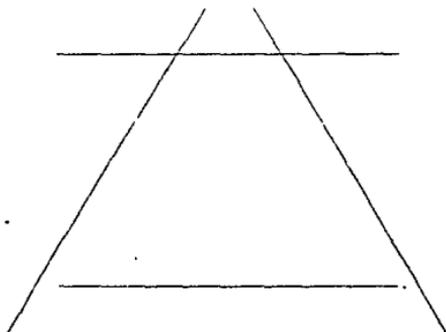


Figura No. 28

Para nosotros los humanos la línea A nos parecerá mas larga que la línea B, y sin embargo ambas líneas A y B tienen la misma longitud; este tipo de ilusión óptica, para nosotros; no pasaría desapercibida por la computadora y evitaría el error de emitir su veredicto a favor de la línea A como mas larga.

Existen tros muchos factores que ya hemos mencionado que impiden la correcta interpretación de la escena, como son: el ruido, las deformaciones, las sombras, la iluminación, etc.

[4][5][6][7][8][9][12][21][22][23][24][26][27][28][29][30][36]

4.3 APLICACIONES DEL PROCESAMIENTO Y RECONOCIMIENTO DE IMAGENES.

Las aplicaciones del procesamiento y reconocimiento de formas son muy diversas en muy distintos campos, por solo mencionar algunas de las áreas donde podemos aplicar los sistemas de visión, listamos los siguientes:

- *En la industria, que es una de las áreas de mayor aplicación actualmente se utiliza en la inspección de líneas de ensamble, clasificación de productos, control de calidad, montaje, eliminación de rebabas, seguimiento de costura y soldadura, detección de alcance y posicionamiento y recuperación de piezas entre muchas más.
- *En la medicina, se utiliza como auxiliar para la correcta interpretación de imágenes médicas, localizaciones patológicas de venas y arterias etc.
- *En ciencias para el análisis de imágenes microscópicas, procesado de fotografías de trayectorias de partículas nucleares, en la robótica y en un sin fin de especialidades.
- *En geografía se utiliza para la delimitación de zonas y localización de vías de comunicación en imágenes topográficas, localización de fallas sísmicas etc.
- *En climatología se interpretan imágenes climatológicas del satélite para la predicción del tiempo, para análisis de la capa de ozono e índices de contaminación entre otras aplicaciones.
- *En astronomía, para la interpretación y procesado de imágenes espaciales y de satélites, para la radioastronomía etc.
- *En criminalística se comparan e interpretan rostros y huellas dactilares, al igual que firmas y rasgos de interés.
- *También tiene aplicaciones en la servopresentación visual y control navegacional.

Así pues tenemos ante nosotros un mar de posibles aplicaciones, donde tan solo es necesario, un problema que con un poquito de ingenio, se pueda adaptar nuestro sistema para resolverlo. [4],[5],[6],[7],[8],[9],[12],[15],[19],[20],[21],[22],[23],[24],[25],[26],[27],[28],[29],[31],[34],[36],[39],[41],[42],[43],[44].

CAPITULO V

PROGRAMACION E INTERPRETACION

5.0 INTRODUCCION A LA PROGRAMACION Y LA INTERPRETACION DE RESULTADOS.

En este capítulo se le asocia un significado a un conjunto de patrones previamente pre-procesado y reconocidos, además se exponen los distintos métodos de pre-procesamiento y reconocimiento estudiados, los cuales fueron sometidos a pruebas y evaluados, se exponen los resultados observados, las dificultades e inconvenientes, nuestro punto de vista y las conclusiones en cada uno de los casos.

La programación de estos algoritmos se llevó a cabo en Turbo C⁺ y en Turbo pascal V.5.0. por el hecho de que son lenguajes de programación populares, que proporcionan una forma estructurada de programar, limitada por la imaginación (creatividad) del programador y la inminente realidad.

5.1 PROGRAMAS DE PREPROCESAMIENTO:

5.1.1 SUAVIZADO:

Se implemento un programa de suavizado para la eliminación de ruido utilizando los métodos de suavizado B1 y B2 vistos en la seccion 2.1.1 del capítulo 2.

El programa consistió en desplegar en el monitor de una microcomputadora figuras (Triángulos, Circulos y Cuadros) en posiciones aleatorias de la escena, se trabajó sobre imágenes binarias, (Figuras blancas sobre fondo oscuro) y se simuló la presencia de ruido en la escena y la figura; el ruido en la escena lo componen pixeles encendidos (blancos) y en la figura pixeles apagados (negros) utilizando los siguientes criterios de evaluación:

CRITERIO UTILIZADO PARA EL SUAVIZADO DE IMAGENES
TABLA No. 4

CRITERIO	% DE RUIDO ESCENA/IMAGEN	CLASIFICACION	SUAVIZADOS
C1	< 30%	Sin Ruido	Ninguno
	> =30% y < 60 %	Poco Ruidó	1*B1 ,1*B2
	> =60% y < 80 %	SemiRuidosa	2*B1 ,2*B2
	> =80%	Muy Ruidosa	3*B1 ,3*B2
C2	< 30%	Sin ruido	1*B1 ,1*B2
	> =30 % y < 60%	Poco Ruidosa	1*B1 ,1*B2
	> =60 % y < 80%	SemiRuidosa	2*B2 ,2*B2
	> = 80 %	Muy Ruidosa	3*B1 ,3*B2

CONTINUACION TABLA Nº 4

CRITERIO	% DE RUIDO ESCENA/IMAGEN	CLASIFICACION	SUAIVIZADOS
C3	< 10%	Sin ruido	1*B1 ,1*B2
	> = 10% y < 60%	Poco Ruidosa	2*B1 ,2*B2
	> = 60% y < 80%	Semi Ruidosa	3*B1 ,3*B2
	> = 80%	MuyRuidosa	4*B1 ,4*B2
C4	< 10%	Sin Ruido	1*B1 ,1*B2
	> = 10% y < 60%	Poco Ruidosa	2*B1 ,2*B2
	> = 60% y < 80%	Semi Ruidosa	3*B1 ,3*B2
	> = 80%	Muy Ruidosa	& 4*B1 ,4*B2 Si & + 2*B1

Los porcentajes se calcularon utilizando la sig. expresión:

$$\% E = \frac{(\text{No. Enc. Pix}) * 100}{\text{No. Total}} \quad \quad \quad \% I = \frac{(\text{No. Ap. Pix}) * 100}{\text{Pix. Total}}$$

Donde: % E = Porcentaje de Ruido en la escena.

% I = Porcentaje de ruido en la imagen.

No. Enc. Pix = Número de pixeles encendidos en la escena (Sin contar los de la imagen).

No.Ap. Pix. = Número de pixeles apagados en ml imagen (Sin contar los de la escena).

No. Total = Número total de pixeles en la escena.

Pix. total = Número total de pixeles en la imagen.

Quando se utilizó el criterio número 1 (C1) Se observó en todos los casos imágenes con mucho ruido y de baja calidad después del suavizado, por lo que se optó por aplicar el criterio número dos (C2), mejorando al menos un poco la calidad después del suavizado.

Al probar el tercer criterio (C3) la calidad de la imagen mejoró notablemente en las imágenes Sin Ruido, Poco Ruidosas y Semi-Ruidosas; en los casos con imágenes muy ruidosas el cambio en la calidad de la imagen fue prácticamente imperceptible por lo que se optó implementar un cuarto criterio, en el cual se aplica un doble procesado de B1. De esta forma mejoró notablemente la calidad obtenida en los casos de imágenes o escenas Muy ruidosas.

La bitácora que se utilizó para evaluar este estudio fue la siguiente:

TABLA No. 5						
No. de prueba	Figura	Escena	Imagen	Criterio	Procesados	Calidad Observada
1	.	Poco Ruidosa	Poco Ruidosa	C4	2*B1 2*B2	Suficiente
2	.	Poco Ruidosa	Muy Ruidosa	C4	4*B1 2*B2, 2*B1	Buena
.
.
.
n	.	Sin Ruido	Poco Ruido	C4	2*B1 1*B2	Buena

Para evaluar la calidad general de la escena se clasificó de acuerdo al observador en Pésima, Mala, Suficiente, Aceptable y Muy buena.

Se observó que en el 90% de los casos en que se presentó el triángulo como figura, el suavizado B2 apaga al pixel del ángulo superior del triángulo.

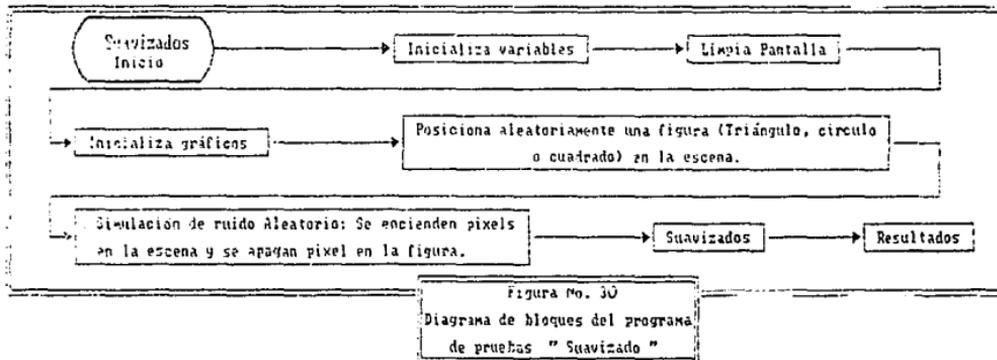
Cuando se presenta el caso de imagen y escena Poco ruidosa. Al aplicar el primer suavizado B2 deja la imagen de la figura con el efecto de escalera. (Ver la sig. figura).



Figura No. 29

El inconveniente de utilizar estos suavizados binarios es el tiempo que toma el procesamiento en mostrarnos una imagen de calidad aceptable.

A continuación se muestra un diagrama de bloques del programa general do suavizado.



Para poder observar más detalladamente el desarrollo aplicado del cuarto criterio de evaluación utilizado en la siguiente página se muestra un diagrama de flujo con el procedimiento de suavizado utilizado.

Suavizados
(inicio)

POELF = Pixels apagados en la figura.
PEEEF = Pixels encendidos en el fondo.

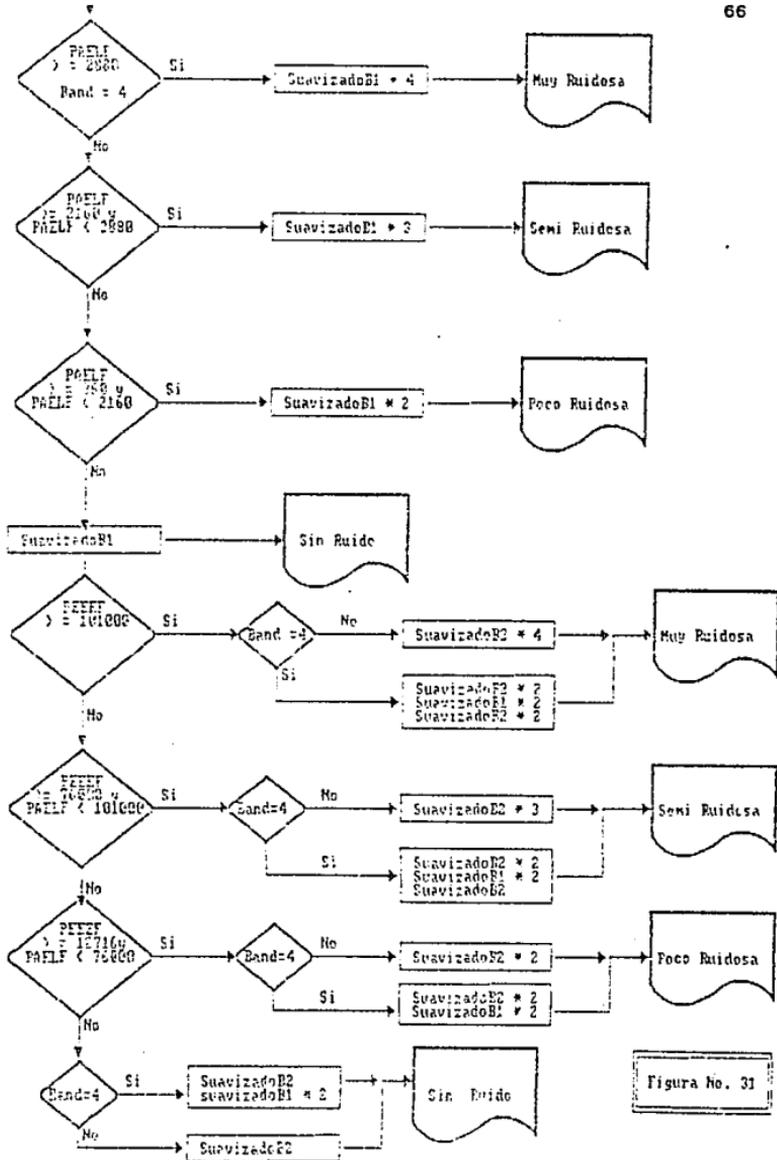
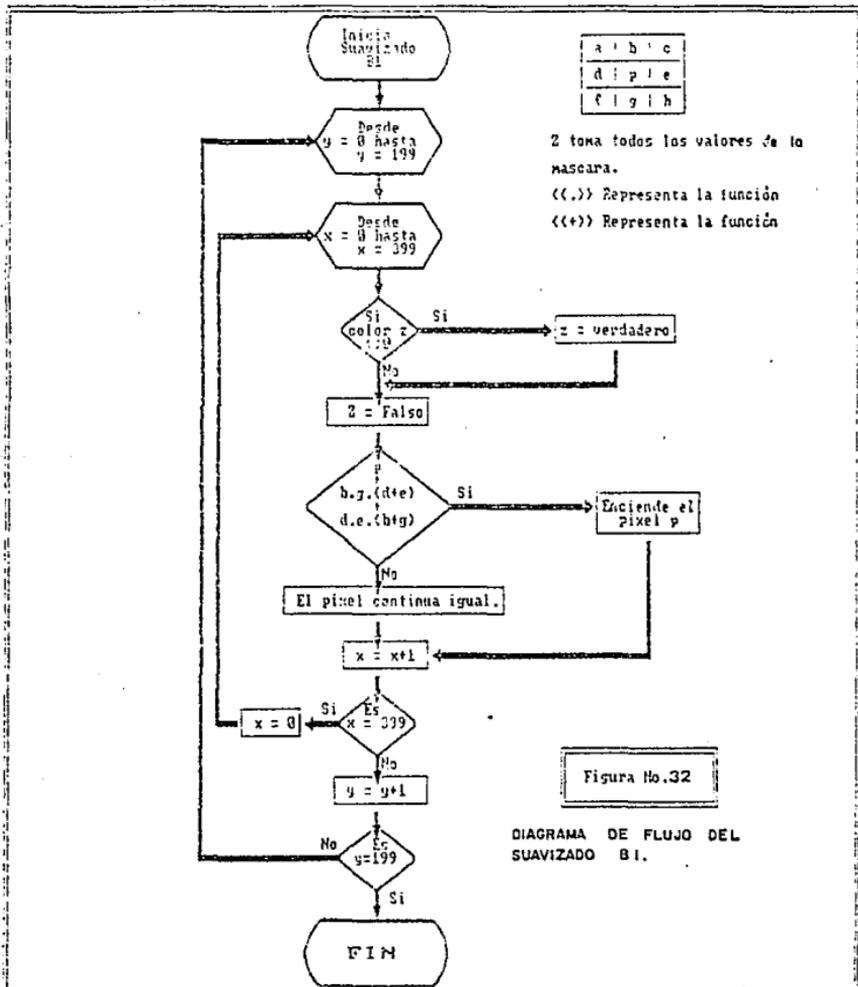


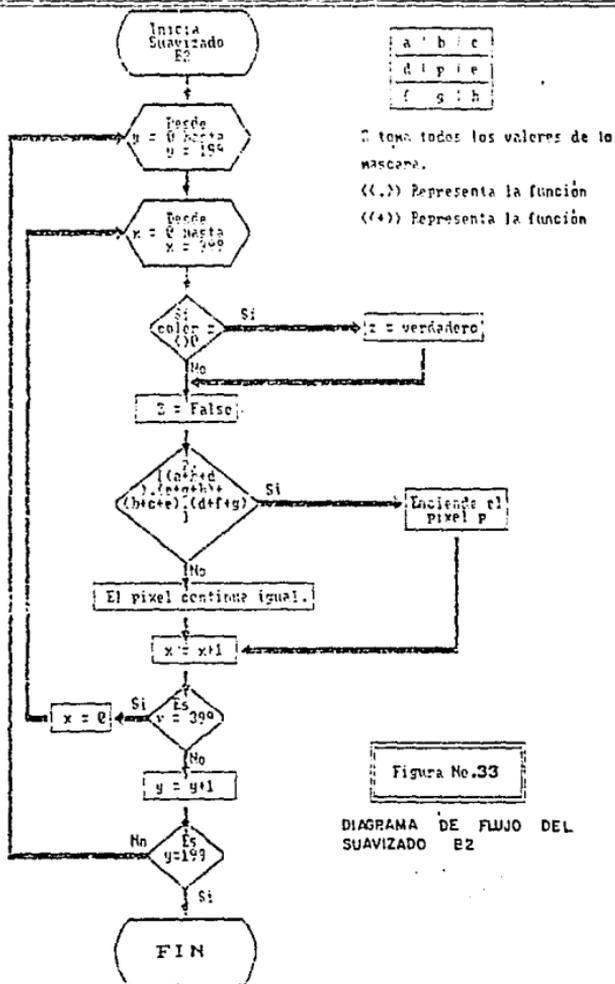
Figura No. 31

El suavizado B1 rellena los pequeños huecos (de 1 pixel) en zonas oscuras además de rellenar los cortes y muescas en segmentos de lados rectos; a continuación se muestra el diagrama de flujo de este tipo de suavizado.



El suavizado B2 elimina los unos aislados y las protuberancias a lo largo de segmentos de lados rectos y se puede representar mediante la siguiente expresión booleana.

$$B2 = p \cdot ((a+b+d) \cdot (e+g+h) + (b+c+e) \cdot (d+f+g))$$



Para ilustrar este programa de suavizado, se muestra una escena ruidosa en la cual aparentemente aparece la figura de un cuadrado.

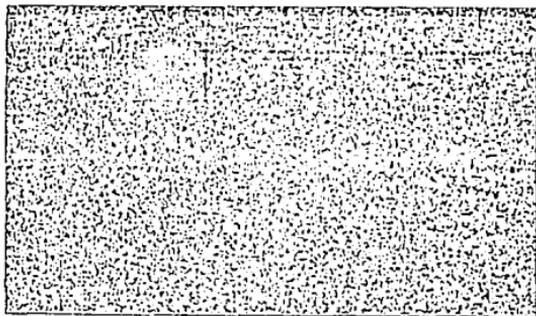


Figura No.34

A la escena anterior se le aplica un suavizado $B1$ a la pura imagen y se obtiene escena que muestra la siguiente figura.

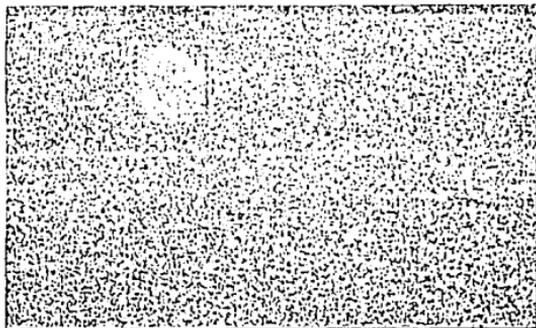


Figura No. 35

Se trata de mejorar aún más esta Imagen por lo que se le aplicó de nuevo cuatro veces consecutivas el suavizado B1 (Ver secuencia en las figuras 36, 37, 38 y 39). Esto se efectuó con el fin de disminuir el ruido en la imagen ya que originalmente tenía bastante ruido.

Posteriormente se le aplica un doble suavizado B2 (Ver figuras 40 y 41) En donde se observa claramente el decremento de ruido del fondo de la escena; aunque al aplicar este tipo de suavizado la calidad de la imagen decae un poco, pues los huecos que existían se incrementan en tamaño ensanchándose y formando ángulos rectos.

Finalmente se trata de reparar la figura aplicando un nuevo suavizado B1 donde la imagen obtiene una mejora casi imperceptible y así obtener como resultado la figura 42.

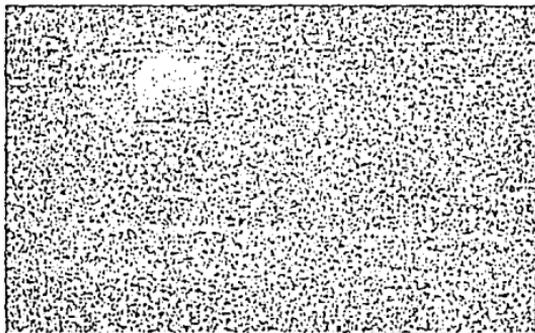


Figura No. 36

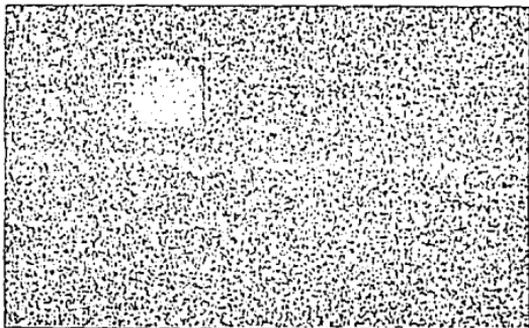


Figura No. 37

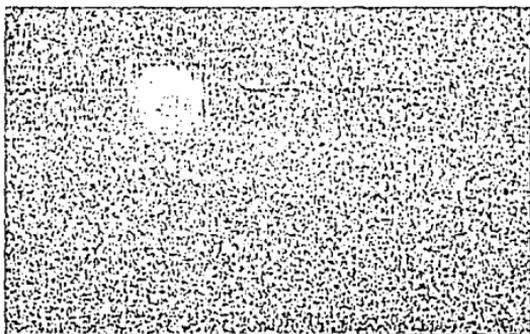


Figura No. 38



Figura No. 39

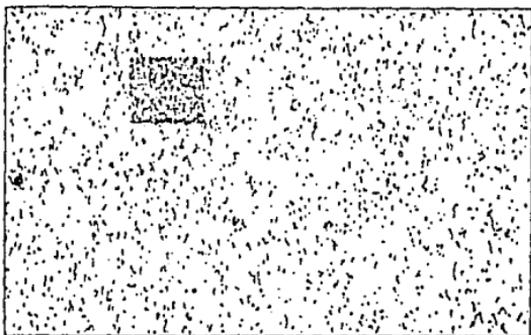


Figura No. 40

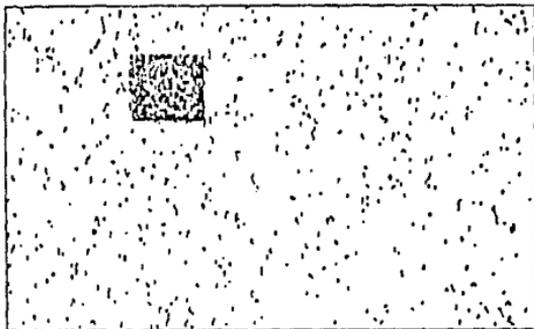


Figura No. 41

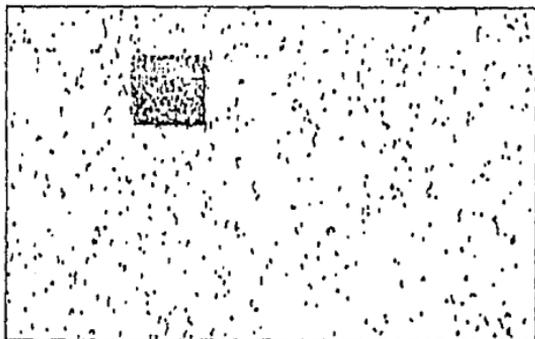


Figura No. 42

5.2 PROGRAMAS DE RECONOCIMIENTO

5.2.1 RECONOCIMIENTO POR ANGULOS

Para poder tener un criterio un poco mas amplio y efectuar un análisis mas profundo se elaboró un programa de reconocimiento de figuras por ángulos el cual está limitado a sólo dos figuras: cuadrado y triángulo isósceles.

Este programa sitúa aleatoriamente o controlando la posición si así se desea de una o dos figuras al azar en la pantalla del monitor pudiéndose presentar cualquiera de los casos siguientes:

- 1) Un triángulo isósceles.
- 2) Un cuadrado.
- 3) Un triángulo y un cuadrado.
- 4) Dos triángulos isósceles.
- 5) Dos cuadrados.

Las figuras estan formadas por asteriscos que el programa trata de identificar como tales, cualquier otro tipo de caracter pasará desapercibido por el reconocedor. En aquellos casos en que se presenta solo una figura el reconocedor verifica solo uno de los dos posibles resultados puesto que el algoritmo está restringido a sólo dos figuras, probamos si es el triángulo; de ser así, se emite el juicio, en caso contrario será cuadrado. De esta misma manera se procede en el caso de dos figuras efectuándose tan solo dos verificaciones en el peor de los casos.

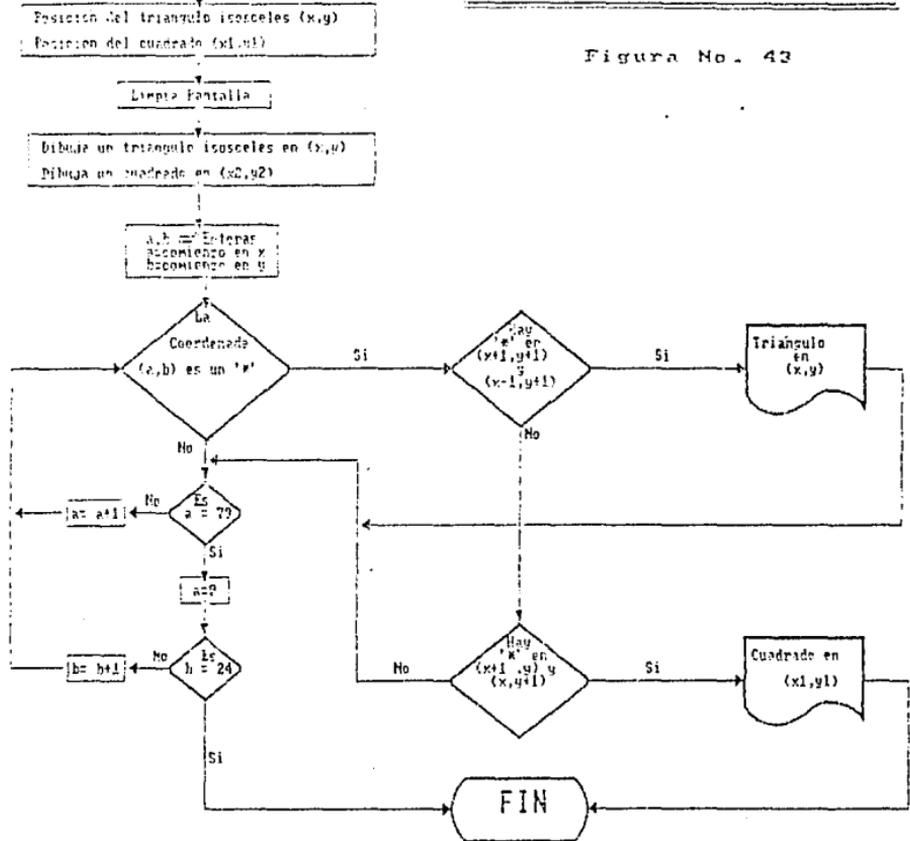
Como el reconocedor está limitado a solo triángulos isósceles y cuadrados solo se efectúan comprobaciones en dos puntos de cualquier vértice. Si los puntos están en ángulo recto, el objeto ha de ser un cuadrado. Si los puntos forman un ángulo de 60° , entonces el objeto es un triángulo isósceles. Solamente se verifican los puntos o celdas adyacentes al vértice para ver si están en el lugar correcto. Por ejemplo, en cualquiera de los vértices de un triángulo habrá un asterisco encendido sobre la diagonal. De manera similar proseguimos para encontrar un cuadrado, excepto que, cada vez que se localice un asterisco compruebo si hay otro en ángulo recto con él.

A continuación se muestra un diagrama de flujo (Figura No. 43) del programa de reconocimiento por ángulos.

Reconocimiento
por
Ángulos

Diagrama del programa de
reconocimiento por
Ángulos

Figura No. 43



En la figura número 46 se provoca un desplazamiento horizontal de diez posiciones a la izquierda de una de las figuras (El cuadrado en este caso concreto) para observar el comportamiento del algoritmo, y se observó un resultado satisfactorio, ya que el programa reconoce perfectamente bien ambas figuras.

triángulo en 20 10
cuadrado en 10 10

```

* * * * *
*           * * *
*         * * * *
*       * * * * *
*     * * * * *
* * * * *

```

Figura No. 46

Para verificar el comportamiento de otro tipo de trazapamiento se produce un desfazamiento vertical de cuatro posiciones y el algoritmo responde nuevamente satisfactoriamente. (Ver figura No. 47).

```

      *
     * *
    * * *
   * * * *
  * * * * *
 * * * * *
* * * * *

```

triángulo en 20 10
cuadrado en 20 14

Figura No. 47

Después de haber analizado este reconocedor por ángulos podemos sintetizar nuestras conclusiones en los siguientes puntos:

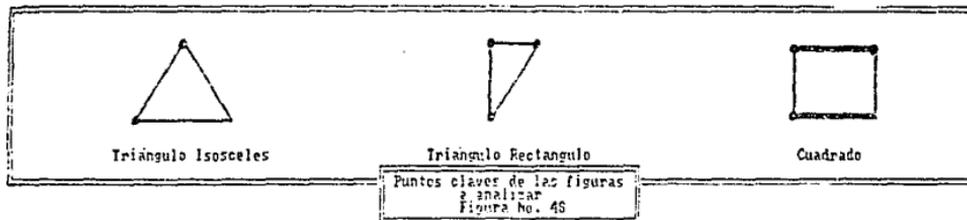
- 1).-En este caso específico de reconocimiento de ángulos la orientación de los objetos sí afecta al reconocimiento pues para obtener buenos resultados el programa requiere que los objetos se encuentren con la orientación mostrada en las figuras, por lo tanto, no se permita rotación de las mismas.
- 2).-El tamaño no afecta al reconocimiento, ya que lo que hacen las rutinas es confirmar el ángulo de intersección entre dos caras.
- 3).-De igual manera el traslapamiento no es un factor de fallo para este reconocedor en particular.
- 4).-Debido a la simplicidad y limitación del programa, es suficiente, en términos generales, por lo que sólo se podrá aplicar a situaciones muy controladas con un número mínimo de objetos variados.

**ESTA TESIS NO DEBE
SALIR DE LA BIBLIOTECA**

5.2.2 RECONOCIMIENTO POR PUNTOS CLAVE

Para tener un rango un poco mas amplio de reconocimiento de figuras se implementó el programa de reconocimiento por puntos clave, agregando una figura a las figuras del programa de reconocimiento por ángulos. (esta figura es el triángulo rectángulo), que con el algoritmo anteriormente mencionado podría confundir al programa en su toma de decisión y optar por un triángulo rectángulo en vez de un cuadrado o viceversa; en este programa tratamos de salvar este problema.

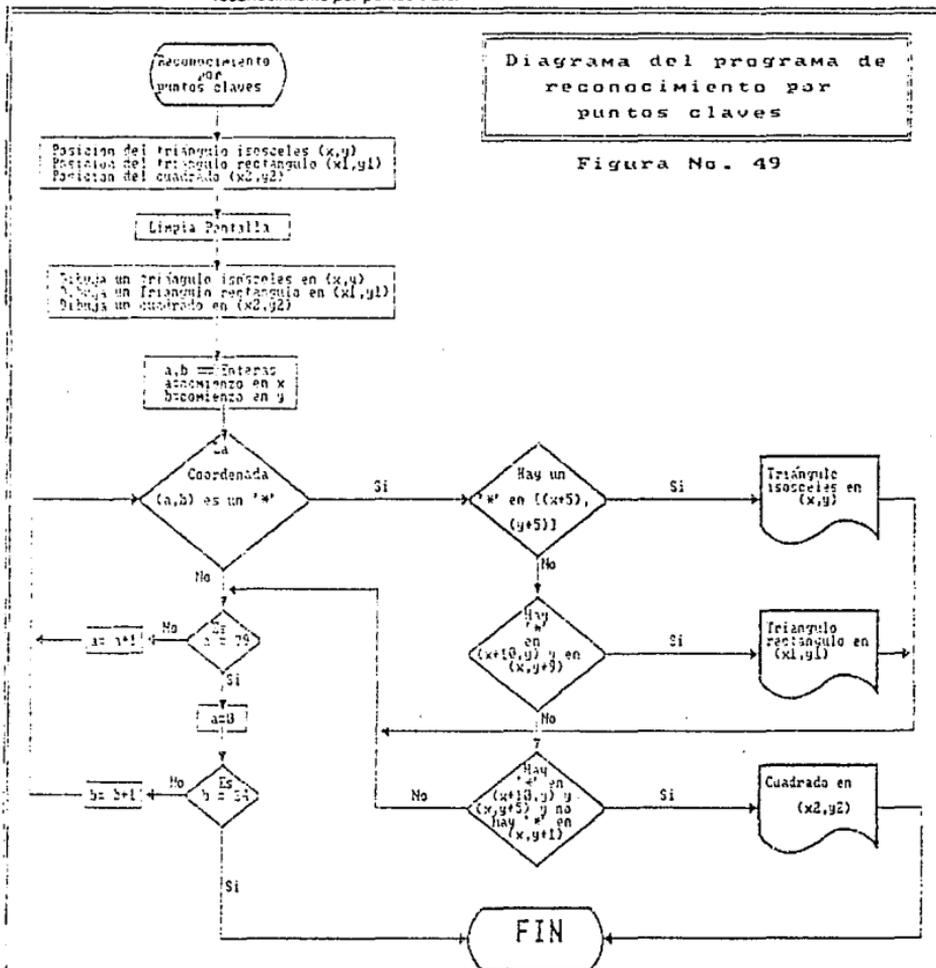
Con este procedimiento hacemos que el programa identifique objetos que siempre tengan el mismo tamaño, examinando solamente puntos clave, los cuales se escogen de manera tal que sólo uno de los objetos satisfaga las condiciones. En la siguiente figura se muestran las tres figuras que debe identificar el programa así como sus correspondientes puntos claves.



Por la experiencia del programa anterior (reconocimiento por ángulos) damos por hecho el preprocesamiento de las imágenes y nos proporciona así, sólo el contorno de las figuras, mostrando siempre las tres figuras en posiciones aleatorias o controladas (según se desee) de nuestro monitor.

Se crea una rutina especial para cada tipo de figura por lo que forzosa-mente se verificarán todas las posibilidades y el tiempo de reconocimiento será constante en cualquier situación.

A continuación se muestra un diagrama ilustrativo del programa de reconocimiento por puntos claves.



El tamaño de los objetos deben ser constantes pues los puntos claves están a distancias fijas.

La eficiencia del programa en términos generales es buena, siempre y cuando se presenten los casos para los cuales está programado; el medio es mucho más restringido que el reconocimiento por ángulos aún cuando este incorpora la opción de triángulo rectángulo. En definitiva es fácil que se equivoque.

El programa funciona eficientemente siempre y cuando no se traslapen las figuras; cuando se rompe esta regla, el reconocedor falla fácilmente al emitir su decisión (Ver las figuras 52, 53 y 54). Aunque se observó que se presentan más decisiones erróneas cuando se traslapan el triángulo rectángulo y el cuadrado que en cualquier otra combinación de figuras.

triángulo isósceles en 20 7
cuadrado en 20 12

triángulo rectángulo en 20



Figura No. 52

En esta figura se observa la equivocación del reconocedor al emitir el juicio de la posición del cuadrado ya que su posición original es ($x=20$ $y=9$)

5.2.3 EL RECONOCEDOR DELTA-D O POR CAMBIOS DE DIRECCION

A los cambios que suele haber en la dirección se les llama en matemáticas o en ciencias informáticas valores delta, es por eso que se les llamó así a este reconocedor.

Para ilustrar este programa, supongamos que comenzamos en un punto del triángulo, siguiendo las líneas rectas que conforman su perímetro hasta regresar al punto inicial, se habrá cambiado tres veces de dirección. De esta manera el número de vértices de la figura será igual al número de cambios de dirección detectados.

El reconocedor por cambios opera siguiendo el contorno de los objetos, en el ejemplo mencionado anteriormente el programa sigue una línea recta hasta encontrar una intersección con otra línea. En el momento de detectar una intersección busca la siguiente línea recta a seguir. Aunque esto no es difícil programar, la decisión de la línea a seguir no es por intuición pues conllevaría a un retroseguimiento redundante. Para evitar este problema se crea una base de datos que llamamos punto-visto, que contiene la lista de todos los puntos visitados esto se hace con el propósito de saber cuando se está de vuelta en el punto inicial; así sabemos cuando se ha recorrido todo el contorno del objeto, y evitar que la rutina siga de nuevo su propio camino.

A continuación se muestra un diagrama en el cual se muestra el algoritmo del reconocedor por cambios de dirección `delta_d`.

Para este caso el programa funciona perfectamente bien; el problema se presenta cuando las figuras coinciden en la misma coordenada, en otras palabras cuando las figuras se traslapan (Fig. No. 57) El reconocedor se viene abajo pues no consigue encontrar ningún triángulo y sólo por accidente detecta el cuadrado.



Figura No. 57

En el caso de la figura anterior no se pueden encontrar los triángulos debido a la gran cantidad de dirección que se presentan, pues se siguen demasiadas líneas diferentes.

La característica fundamental de este reconocedor es que la orientación de los objetos no afecta al proceso de reconocimiento; esto lo hace en cierto modo atractivo aplicarlo a situaciones reales, además de que el tamaño de los objetos no impide la toma de decisiones correctas.

Debido a que el reconocedor por cambios de dirección DELTA-D se puede aplicar de manera más general que el reconocimiento por ángulos o por puntos clave no opera de manera tan eficaz; sin embargo en muchas ocasiones vale la pena pasar desapercibida esta pérdida de eficiencia.

Este reconocedor se equivoca fácilmente cuando hay superposición de objetos, pero no confunde un tipo de objeto por otro dentro de las exigencias de una situación concreta.

5.2.4 RECONOCIMIENTO SINTACTICO

Este programa sitúa una o dos figuras en localizaciones determinadas, la elección de las figuras esta determinada por el usuario del programa, pudiendose presentar cualquiera de los casos ilustrados en la Tabla No. 6.

Una vez presentada(s) la(s) figura(s), se procede a efectuar un barrido en "x" (horizontal) de la imagen, hasta que se encuentra un pixel encendido; se verifica el siguiente pixel y en caso de estar apagado se guarda la posición del pixel encendido; si esta encendido, se continua verificando. Al encontrar un segundo pixel encendido que su consecutivo en orden no lo esté, se calcula la distancia que hay entre los dos pixeles encontrados y se guarda en la memoria de la computadora.

De esta manera se procede a analizar el siguiente renglón, al terminar de revisarlo se compara la diferencia obtenida en este renglón con la obtenida en el anterior y así concluimos que: "Las distancias permanecieron constantes", "Las distancias se fueron incrementando", ó "Las distancias se fueron decrementando". Este procedimiento continúa hasta encontrar un renglón con pixeles apagados.

Se continua aplicando este mismo procedimiento con la siguiente figura.

Una vez analizados estos patrones se procede a asignarle(s) un nombre de acuerdo con el comportamiento que hayan manifestado las distancias, de esta manera decimos que:

<i>Identificador</i>	<i>Comportamiento</i>
Círculo	Incremento-Igualdad- Decremento
Triángulo	Incremento.
Triángulo (Cono)	Decremento
Rectángulo	Igualdad

Programa de Reconocimiento Sintactico.

92

Figura No. 61

DIAGRAMA DEL PROGRAMA DE RECONOCIMIENTO SINTACTICO

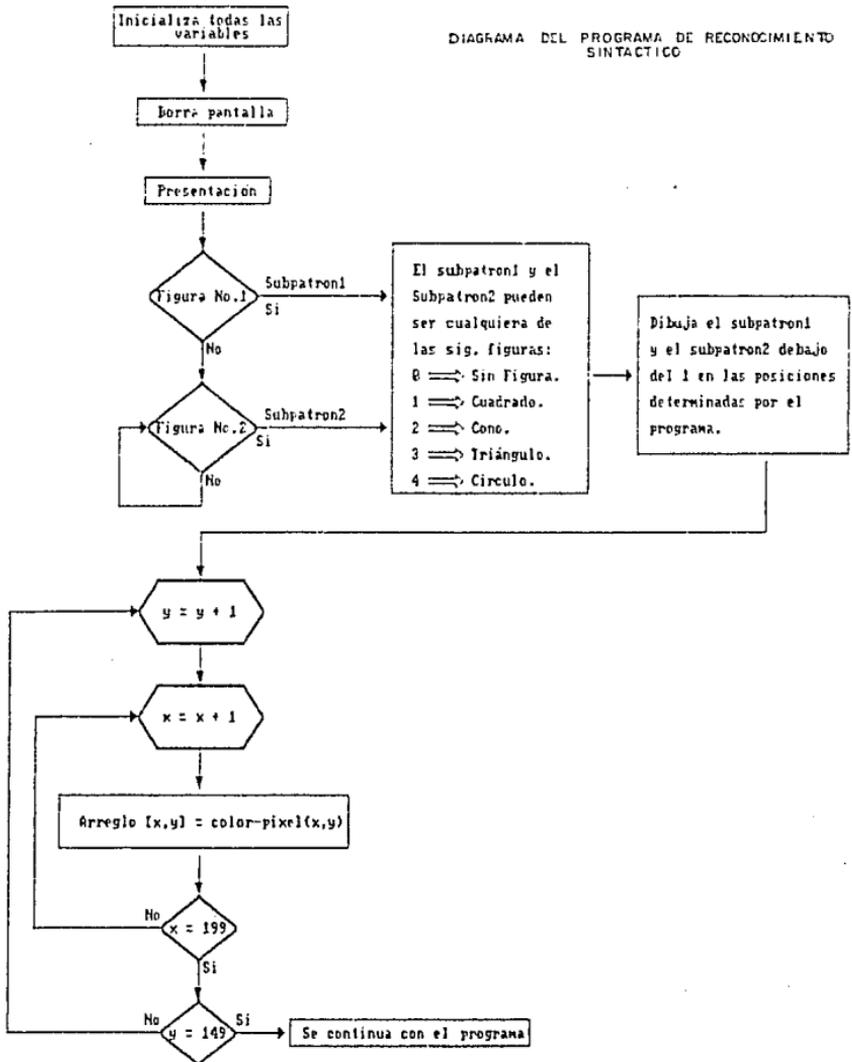


Figura No. 1	Figura No. 2	Figura No. 3	Resultado
		Una línea	Una línea
			Cuadrado
			Círculo
			Triángulo
			Triángulo
		 	Dos Cuadrados
		 	Dos Círculos
		 	Dos Triángulos
		 	Dos Conos
			1 Cuadrado
		 	Sólido
		 	Balanza
		 	1 Cuadrado y 1 Cono
			1 Círculo
		 	Nada
		 	Nada
		 	Esfera
			1 Triángulo
		 	Cono
		 	Pyramide
		 	Bomb
			Triángulo
		 	1 Cono y 1 Cuadrado
		 	1 Cono y 1 Círculo
		 	Bolón de arena

El algoritmo de este programa se ilustra más detalladamente en las siguientes figuras.

La figura número 61 muestra un diagrama con la primera parte del programa, la figura 62 muestra el diagrama de flujo de la segunda parte de este programa, el cual trata básicamente lo que es el reconocimiento sintáctico.

En este programa se le van asignando nombre a las figuras de acuerdo al comportamiento que hayan manifestado; la Tabla número 6 muestra todas las posibles combinaciones que se pudiesen presentar en un determinado momento en el transcurso de operación del programa, así como la interpretación que se le da a cada una de las figuras.

5.2.4.1 ANALISIS DEL PROGRAMA

El programa identifica las figuras en el 100% de los casos; no permite el traslape de figuras, pues, si se hace no funciona. El tamaño de las figuras no altera los resultados, aunque la rotación sí, por lo que el programa sólo permite las figuras en las posiciones y orientaciones determinadas por él.

No es posible obtener ningún resultado acertado cuando hay presencia de ruido en la imagen.

Por lo general este tipo de análisis se hace en escenas complejas; como nuestra escena está algo restringida, en general se obtienen buenos resultados, sin tomar en cuenta las restricciones el programa es bastante eficiente.

La primera pantalla que nos aparece al correr el programa es la presentación del mismo, en la cual se muestran las diferentes opciones de figuras, de las cuales el usuario puede escoger las que desea presionando el número correspondiente a cada figura. (Ver figura número 63).

Reconocimiento de figuras



(1)



(2)



(3)



(4)

Elija Figura 3 u 4:

Figura No. 63

Supongase que se escogen las figuras 3 y 4 con lo cual el programa muestra una imagen como la que se observa en la figura no. 64.

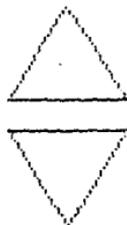
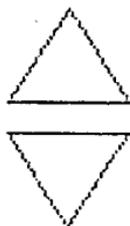


Figura No. 64

El algoritmo procede a hacer el barrido de la escena en el cual se va detectando la figura, y posteriormente tomar la decisión correspondiente al comportamiento manifestado.



Rombo

Figura No. 65

CONCLUSIONES:

Por lo general los sistemas de visión suelen presentar dos inconvenientes principales; el precio elevado y la relativa lentitud de trabajo. sin embargo, existen aplicaciones reales en los que un sistema de visión artificial de no excesiva resolución como el nuestro, podría permitir una automatización eficiente.

El sistema desarrollado está limitado a trabajar únicamente sobre imágenes bidimensionales y binarias, además está limitado cada uno de los casos vistos, a las imágenes aceptadas por cada programa específicamente; permitiéndose o no, según el caso, traslapamiento, rotación y ubicación de los objetos en la escena.

La correcta interpretación de la escena es vital para que la computadora pueda tomar decisiones apropiadas en situaciones distintas, dependiendo de lo que ocurra en su entorno, desafortunadamente, los casos estudiados, sólo podrán emitir juicios acertados para aquellos casos que estén contemplados dentro de su programación.

Los humanos, debido tanto a experiencias vividas, como a su capacidad de deducción, puede obviar una gran cantidad de imperfecciones, ruido, desigual o mala iluminación, partes ocultas o incompletas, desvanecimientos locales etc. Las técnicas que se usan para resolver algunos de estos problemas, son tan variadas como restringidas y por lo general solo se pueden aplicar bajo determinadas condiciones y específicamente de acuerdo al fin propuesto.

Ha quedado establecida la posibilidad de realizar sistemas de procesamiento y reconocimiento de imágenes a costos realmente bajos, con solo contar con una microcomputadora de propósito general y una pequeña inversión para implementar el digitalizador de imágenes, además del tiempo que se llevará la programación.

Este sistema es suficientemente válido (claro que con unas pequeñas modificaciones de adecuación a la aplicación específica) para permitir la automatización de varios procesos industriales de naturalezas diversas, en los que la resolución visual exigida no es elevada y el tiempo de procesamiento no sea tan reducido para lograr el éxito de las tareas implicadas.

XVII

BIBLIOGRAFIA:

- [1] Salvador Mercado M. *¿COMO HACER UNA TESIS?*. Limusa, 1991.
- [2] Hilda Basulto. *CURSO DE REDACCION DINAMICA*. Trillas, 1987
- [3] Andrew S. Glassner. *COMPUTER GRAPHICS A USER'S GUIDE FOR ENGINEERS AND DESIGNERS*. Design Press, 1989.
- [4] Pierre López / Jean Neuma Foulc * Sistema Sensorial, Reconocimiento del Entorno (pp 113,119,124.) * "INTRODUCCION A LA ROBOTICA, ENSEÑANZA, INVESTIGACION Y DESARROLLO". Ediciones Arcadia, S.A., 1987.
- [5] Pierre López / Jean Neuma Foulc * Sistema de Visión Artificial Directa (pp 157,177) * "INTRODUCCION A LA ROBOTICA, ENSEÑANZA, INVESTIGACION Y DESARROLLO". Ediciones Arcadia, S.A., 1987.
- [6] Yoram Koren. * Sensores and Intelligent Robots (vision system) pp. 229,242 * "ROBOTICS FOR ENGINEERS". Mc. Graw Hill, 1987.
- [7] J. Ma. Amngulo Osategui / Rafael Avilés Gonzáles. * Visión de Máquina (pp 257,275) * "CURSO DE ROBOTICA". Editorial Paraninfo, 1989 3ra. edición.
- [8] Mikell P. Groover, Mitchel Weiss, Roger G. Nagel y Nicholas G. Odrev. * Visión de Máquina (pp 177,206) * "ROBOTICA INDUSTRIAL, TECNOLOGIA, APLICACIONES". Mc. Graw Hill, 1989.
- [9] K.S. Fu, R. C. González, C. S. G. Lee. * Detección (pp. 275,30)*. "CONTROL, DETECCION, VISION E INTELIGENCIA". Mc. Graw Hill, 1988
- [10] K. S. Fu, R. C. González, C. S. G. Lee. * Visión de Bajo Nivel (pp 306,374) * "CONTROL, DETECCION, VISION E INTELIGENCIA". Mc. Graw Hill, 1988.
- [11] K. S. Fu, R. C. González, C.S.G. Lee. * Visión de Mas Alto Nivel (pp 375,464) * "CONTROL, DETECCION, VISION E INTELIGENCIA". Mc. Graw Hill, 1988.

XVIII

[12] José Ma. Angulo Usategui * Visión De Máquina, Sistema De Programas De Aplicación *. "ROBOTICA PRACTICA TECNOLOGIA Y APLICACIONES". Paraninfo. 1986.

[13] Newman / Sproull "PRINCIPIES OF INTERACTIVE COMPUTER GRAPHICS". International student edition / Mc. Graw Hill. 1973.

[14] David F. Rogers / J. Alan Adams "MATHEMATICAL ELEMENTS FOR COMPUTER GRAPHICS", Second edition. Mc. Graw Hill. 1990.

[15] Francisco Casacuberta / Enrique Vidal. "RECONOCIMIENTO AUTOMATICO DEL HABLA". Marcombo Boixareu Editores. 1987.

[16] David. F. Rogers. "PROCEDURAL ELEMENTS FOR COMPUTER GRAPHICS". Mc. Graw-Hill Book Company. 1985.

[17] K. S. Fu. "SYNTACTIC METHODS IN PATTERN RECOGNITION MATHEMATICS IN SCIENCE AND ENGINEERING" Vol. 112. Academic Press. 1974.

[18] J. M. Angulo / José N6 "ROBOTICA". Paraninfo . 1986.

[19] Luis Pedro Marzán Peláez / Narciso García Santos. * *Detección de Bordes en imágenes (I)* (pp 203,206) *. "MUNDO ELECTRONICO" No.162. 1986.

[20] Luis Pedro Marzán Peláez / Narciso García Santos, * *Detección de bordes en imágenes (II)* (pp 123-127) *. "MUNDO ELECTRONICO" No. 163. 1986.

[21] Antonio Falcon Martel / Juan M6ndez Rodríguez / Francisco M. Hernández Tejera. * *Detección Visual de la Orientación de Piezas* (pp 99-102) *. "MUNDO ELECTRONICO" No.158. 1986.

[22] F. J. López Aliqué / P. M. Gutiérrez Conae / Ma. Isabel Acevedo. * *Arquitecturas de Ordenador para Tratamiento Digital de Imágenes (I)* (pp 71-78) *. "MUNDO ELECTRONICO". No. 158. 1986.

[23] F. J. López Aliqué / P.M. Gutiérrez Conae / Ma. Isabel Acevedo. * *Arquitecturas de Ordenador Para Tratamiento Digital de Imágenes (II)* (pp 103-110) *. "MUNDO ELECTRONICO". No. 160. 1986.

XIX

[24] V. Arnau / J. Albert / M. Vicens / F. Forri / J. Pelechand * *Tratamiento Digital de Imágenes (I). Perspectiva Histórica.* (pp 87-97). * "MUNDO ELECTRONICO". No. 207. Junio de 1990.

[25] M. Vicens / J. Albert / V. Arnau / V. Cerveron / G. Fabregat. * *Tratamiento Digital de imágenes. Técnicas Básicas* (pp 67-76). * "MUNDO ELECTRONICO". No. 208. Julio de 1990.

[26] J. Amat / A. Casals / V. Ularo. * *Computadores Para El Tratamiento de Imágenes* (pp 95-105) * "MUNDO ELECTRONICO". No. 149. 1985.

[27] Armando Roy / Jesús Navarro / Luis Zorzamo. * *Sistema De Tratamiento De Imágenes Para Robots Industriales* (pp 66-71) * "MUNDO ELECTRONICO". No. 163. 1986.

[28] V. Fernandez / S. Velilla / A. Roy. * *Almacenamiento De Imágenes En Un Computador PDP-11/34* (pp 67-76) * "MUNDO ELECTRONICO". No. 161. 1986.

[29] M. A. Lagunas / R. García / V. Cásares / A. R. Figueiras. * *Reconocimiento de Formas I; Aprendizaje Secuencial y Aplicación al Reconocimiento de Caracteres Escritos.* (pp 115-120) *. "MUNDO ELECTRONICO" No. 99, Octubre de 1980.

[30] M. A. Lagunas / R. García / V. Cásares / A. R. Figueiras. * *Reconocimiento de Formas II; Aprendizaje Secuencial y Aplicación al Reconocimiento de Caracteres Escritos* (pp 125-130) *. "MUNDO ELECTRONICO" No. 103, Febrero de 1981.

[31] L. Brett Glass. * *Digital Video Interactive* (pp 283-289) * "BYTE" Volumen 14. No. 6; May 1989.

[32] Steve Clarcia. * *Build A Gray- Scale Video Digitizer Part 1: Display / Receiver* (pp. 95-106). * "BYTE". Vol. 12. Mayo de 1987.

[33] Steve Clarcia. * *Build A Gray- Scale Video Digitizer Part 2: Digitizer / Transmitter* (pp. 129-138). * "BYTE" vol. 12 Junio de 1987.

[34] Steve Clarcia. * *Using The Image Wise Video Digitizer Part 1: Image Processing* (pp. 113-119). * "BYTE" vol. 12 Julio de 1987.

XX

- [35] Steve Ciarcia. *Using The Image Wise Video Digitizer Part 2: Colorization (pp. 117-121). * "BYTE" Vol.12 August 1987.
- [36]Walt Penney. * *Sound And Images* (pp 243-256) * "BYTE" Volúmen 14 No. 13. December 1989.
- [37] Bobby Saffari. * *Putting DSPs To Work* (pp 259-272) *. "BYTE" Volúmen 14, No. 13. December 1989.
- [38]Raymond Kurzweil. * *Beyond Pattern Recognition* (pp 277-280) *. "BYTE" Volúmen 14, No. 13. December 1989.
- [39]Benjamin M. Dawson. * *Changin Perception Of Reality* (pp 293-304) *. "BYTE" Volúmen 14, No. 13. December 1989.
- [40] Gene Toner. * *Build R-E's Video Frame Grabber* (pp 31-40) *. "RADIO ELECTRONICS" Vol. 6: No.8. Agosto de 1993.
- [41]Gene Toner. * *Video Capture On The Cheap*, (pp 37-46) *. "RADIO ELECTRONICS" Vol. 6:2 No. 12. Diciembre de 1991
- [42]John L. Cuadrado / Clara Y. Cuadrado.* *AI In Computer Vision* (pp 237-258) *. "BYTE" Vol. II No. 1. Enero de 1986.
- [43]Phil Dunbar. * *Machine Vision* (pp 161-173) *. "BYTE" Vol. II. No. 1. Enero de 1986.
- [44]Reinhard Braver / Klaus Dunemann / Thomas Rastetter. * *Robots Learn To See* (pp. 18-21) * "ENERGY & AUTOMATION (SIEMENS)" Vol. 9 No. 6 1987.
- [45]Steve Wood. "USING TURBO PASCAL V.4". Edit. Borland Osborne Mc. Graw Hill. 1988.
- [46]Syteve Wood. "TURBO PASCAL" Mc. Graw Hill. 1986.
- [47] Stephen K. O'Brien. "TURBO PASCAL 5.5". Mc Graw Hill. 1989
- [48]Brian W. Kernighan / Dennis M. Ritchie. "EL LENGUAJE DE PROGRAMACION C". Prentice-Hall. 1991.

[49] Herbert Schildt. "TURBO C PROGRAMACION AVANZADA". Borland. Osborne / Mc. Graw Hill. 1990.

[50] Herbert Schildt. "TURBO C /C ++ THE COMPLETE REFERENCE". Borland. Osborne / Mc. Graw Hill. 1990.

[51] Herbert Schildt. "UTILIZACION DE C EN INTELIGENCIA ARTIFICIAL". Osborne / Mc. Graw Hill. 1989.

[52] Lee Adams. "VISUALIZATION GRAPHICS IN C". Mc Graw hill 1991.

[53] Peter Norton / Richard Wilton. "THE NEW PETER NORTON PROGRAMMER'S GUIDE TO THE IBM PC & PS/2".

[54] Ronald J. Tocci "SISTEMAS DIGITALES PRINCIPIOS Y APLICACIONES" Prentice Hall, 1987

[55] Enrique Mandado "SISTEMAS ELECTRONICOS DIGITALES" Marcombo S.A., 1987

[56] Ferrel G. Stremier "SISTEMAS DE COMUNICACION" Alfaomega, 1989.

[57] Kaufman / Seidman. "MANUAL PARA INGENIEROS Y TECNICOS EN ELECTRONICA". Mc. Graw Hill. 1988.

[58] Benjamin C. Kud. "SISTEMAS AUTOMATICOS DE CONTROL". CECSA. 1982.

[59] Misha Schwartz. "TRANSMISIONES DE LA INFORMACION, MODULACION Y RUIDO". Mc Graw Hill.

[60] Irwin Miller John E. freud. "PROBABILIDAD Y ESTADISTICA PARA INGENIEROS". Prentice Hall. 1987.