

2ej.



**UNIVERSIDAD NACIONAL AUTONOMA  
DE MEXICO**

**FACULTAD DE INGENIERIA**

**“UN ALGORITMO DE CORRESPONDENCIA ENTRE  
IMAGENES ESTEREO A MULTIPLES RESOLUCIONES ”**

**T E S I S**

**QUE PARA OBTENER EL TITULO DE:**

**INGENIERO EN COMPUTACION**

**P R E S E N T A N:**

**FERNANDO ARELLANO DIAZ**

**ODIN SALGADO TINOCO**

*DIRECTOR DE TESIS:*

*DR. BORIS ESCALANTE RAMIREZ*



**MEXICO, D. F.**

**1998**

**TESIS CON  
FALLA DE ORIGEN**

266284



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 esta 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.

## **AGRADECIMIENTOS :**

A Dios, por estar siempre conmigo y por haberme permitido realizar una meta más en mi vida.

A mis padres, por su apoyo a lo largo de todos mis estudios.

Alvaro Arellano Molina

Celia Díaz Vázquez

A mis hermanos, por su apoyo.

Mireya Arellano Díaz.

Norma Arellano Díaz.

Marisela Arellano Díaz.

Alvaro Arellano Díaz.

Héctor Arellano Díaz.

Javier Arellano Díaz.

A Sonia Marcela Alvarado Huerta por su compañía, apoyo y comprensión.

Al Dr. Boris Escalante Ramírez, por todo su apoyo y disposición en la realización de la Tesis.

A la Universidad Nacional Autónoma de México, por haberme permitido realizar mis estudios en sus aulas y por poner a mi disposición a través de sus instituciones, toda la riqueza cultural que tiene.

A todos mis maestros, por los conocimientos que me transmitieron.

**Fernando Arellano Díaz.**

## **AGRADECIMIENTOS :**

A Dios, por haberme permitido completar mi carrera.

A mis padres, por todas sus atenciones y apoyo, que hacen posible este logro.

Leodegario Salgado Garcia

Esperanza Tinoco Martinez

A mis hermanos, por todo su apoyo.

Nelson Salgado Tinoco.

Rosa Isela Salgado Tinoco.

Irene Salgado Tinoco.

A Luz Adriana Soto Ulloa, por todo su apoyo y comprensión.

Al Dr. Boris Escalante Ramírez, por su guía y asistencia en el tema de Tesis

A la UNAM, por la formación que me dió y por su gran riqueza cultural que siempre estuvo a mi disposición.

**Odin Salgado Tinoco**

# UN ALGORITMO DE CORRESPONDENCIA ENTRE IMÁGENES ESTÉREO A MÚLTIPLES RESOLUCIONES.

## ÍNDICE.

### Introducción.

### Capítulo 1 Marco Teórico.

1.1	Visión por computadora	1
1.1.1	Concepto.	
1.1.2	Aplicaciones.	
1.2	Estereoscopia.	3
1.2.1	Digitalización de imágenes.	
1.2.2	Principios.	
1.2.3	Correspondencia entre imágenes estéreo.	
1.2.4	Métodos para resolver el problema de correspondencia.	
1.3	Múltiples Resoluciones	7
1.3.1	Concepto.	
1.3.2	Filtros Gaussianos	
1.3.3	Submuestreo.	
1.4	Métodos de detección de contornos.	11

### Capítulo 2 Algoritmo de Correspondencia PMF.

2.1	Gradiente de disparidad	14
2.2	Detalles del algoritmo.	16
2.2.1	Características	
2.2.2	Ventajas y Desventajas.	
2.3	Diagrama de flujo del algoritmo PMF.	19
2.4	Pseudocódigo.	20
2.5	Resultados obtenidos con el algoritmo PMF	24

### Capítulo 3 Correspondencia PMF a Múltiples Resoluciones.

3.1	Fundamentación.	26
3.1.1	El Espacio Escala	
3.1.2	Submuestreo	
3.2	Algoritmo de correspondencia sin submuestreo.	28
3.2.1	Esquema piramidal del algoritmo PMF a múltiples resoluciones.	
3.2.2	Pirámide Gaussiana.	
3.2.3	Estrategia del cálculo de la correspondencia dentro de la estructura.	

piramidal.	
3.3 Algoritmo de correspondencia con submuestreo.	38
3.3.1 Esquema piramidal del algoritmo PMF con submuestreo.	
3.3.2 Estrategia del cálculo de la correspondencia aplicando submuestreo.	
3.3.3 El problema de interpolación.	

## Capítulo 4 Resultados y Evaluación.

4.1 Resultados del Algoritmo PMF.	42
4.1.1 Sin aplicar múltiples resoluciones.	
4.1.2 Usando la pirámide Gaussiana.	
4.1.3 Usando submuestreo.	
4.2 Evaluación.	45
4.2.1 Tiempo de Cómputo.	
4.2.2 Calidad ( Porcentaje de puntos correspondidos).	
4.2.3 Calibración de la cámara	
4.2.4 Estimación de la distancia.	
4.2.5 Exactitud en el Cálculo de la Distancia	

Conclusiones.

Bibliografía.

## INTRODUCCIÓN

Debido a los avances tecnológicos principalmente en hardware (velocidad, integración a gran escala, capacidad de memoria, etc.) es actualmente posible implantar sistemas de Inteligencia Artificial que requieren ser procesados en algo muy próximo al tiempo real, o que necesitan mayores recursos de memoria, dependiendo de la aplicación para la que sea destinado el sistema. Es por esto que nos pareció interesante y útil tomar ventaja de estas capacidades y contribuir a la investigación en el campo de Visión Artificial, con inspiración en algunos conceptos de Percepción Visual.

El principal problema de la visión 3D es que se necesita obtener información en 3 dimensiones a partir de imágenes que son bidimensionales y que no determinan de forma unívoca la escena que las originó. Existe una variedad de algoritmos para resolver este problema los cuales caen dentro de dos tipos de procesamiento.

El primero se basa en la interacción percepción - memoria en la que se capta una única imagen y se reconocen los objetos que hay en ella; después se extrae de memoria (donde ha sido almacenada con anterioridad) la estructura tridimensional de los objetos, y finalmente, con esa información es posible obtener la estructura 3D de la escena.

El segundo método se basa en un proceso perceptivo sin recurrir a información previamente almacenada en memoria. Se captan dos o más imágenes de la misma escena y se analiza la estructura 2D de cada una por separado; después, se determinan que elementos de las distintas imágenes son proyección de la misma entidad física, y se calcula la disparidad entre sus posiciones, la que se origina por la diferencias de perspectiva con que fueron tomadas las imágenes; finalmente a partir de la información contenida en las distintas imágenes, de las disparidades detectadas y de la geometría de captación de los sensores (cámaras), se calcula la estructura 3D de la escena. Sobre este último método se trabajó en la presente tesis, ya que es el que resulta más eficiente debido a que no se deben tener conocimientos previos de la estructura 3D de los objetos y por la gran variedad de algoritmos que hay para resolver la dificultad a la que se enfrenta la Visión en 3D.

De todos los procedimientos para la obtención de varias imágenes de la misma escena, el más conocido es el de nuestro sistema ocular con el que captamos dos imágenes que reciben el nombre de par estereoscópico. Basándonos en este tipo de Visión 3D se plantean dos principales problemas: determinar que rasgos de las dos imágenes son proyección de la misma entidad física, y a partir de las disparidades entre sus posiciones y de otra información complementaria, obtener una representación de la escena en tres dimensiones.

El sistema visual humano ha resuelto estos problemas de una forma muy eficiente, a través de un potente sistema de computación en paralelo, y en su funcionamiento está

inspirado el algoritmo que se usa en la presente tesis, por lo cual resulta importante contar con un equipo de cómputo bastante rápido.

El objetivo principal del presente trabajo es implantar un método de correspondencia basado en el algoritmo PMF (Pollard, Mahew & Frisby) complementado con módulos que ayudan a incrementar la velocidad del tiempo de proceso basados en el concepto de múltiples resoluciones del cual se hablará más adelante.

En cuanto a la organización de la tesis creímos conveniente dividirla en cuatro capítulos en los cuales se trata de llevar una secuencia de la resolución del objetivo empezando desde un panorama general de los conceptos que se utilizan en los procesos de Visión por Computadora. A continuación damos una breve descripción del contenido de los cuatro capítulos:

En el primer capítulo se tratan conceptos generales de Visión por Computadora, así como de Estereoscopia, Múltiples resoluciones y Métodos de detección de contornos, bases que son necesarias para comprender el desarrollo del presente trabajo.

El segundo capítulo contiene las bases teóricas del algoritmo de correspondencia PMF así como el proceso de implantación (diagrama de Flujo y explicación de éste) además, se muestran resultados de la aplicación del algoritmo.

En el tercer capítulo se da una explicación de las ventajas que se logran al complementar el algoritmo PMF con las múltiples resoluciones. Posteriormente se aplican dos métodos diferentes de este concepto, como son la pirámide Gaussiana y el Submuestreo de imágenes, mostrando las fases de ambos procesos mediante imágenes.

Por último en el cuarto capítulo se hace una breve comparación de estos métodos, y se presentan los resultados obtenidos con cada uno. Además en estos resultados se muestra como obtener la distancia real de los objetos a partir de imágenes estéreo.



---

## CAPITULO 1: MARCO TEÓRICO.

### 1.1 VISIÓN POR COMPUTADORA

#### 1.1.1. Conceptos.

La visión es el sentido mediante el cual podemos **generar una descripción simbólica de una escena a partir de sus imágenes**. El estudio de los mecanismos de la visión humana ha tenido larga tradición en la historia de la ciencia pero ha sido hasta la aparición de las computadoras digitales, las cuales han dotado a la ciencia de nuevas y potentes herramientas, que ha sido posible la implantación y desarrollo de estos mecanismos y que actualmente pueden ser estudiados en una forma más profunda y continua.

El término “Visión por Computadora” dentro del campo de la Inteligencia Artificial puede considerarse como el **conjunto de todas aquellas técnicas y modelos que nos permitan el procesamiento, análisis y explicación de cualquier tipo de información espacial obtenida a través de imágenes digitales**. Desde sus inicios la Visión por Computadora ha inspirado sus desarrollos en el estudio del sistema visual humano el cual sugiere la existencia de diferentes tipos de tratamiento de la información visual dependiendo de metas u objetivos específicos, es decir, la información visual percibida es procesada en distintas formas con base en las características particulares de la tarea a realizar, por lo que la visión por computadora propone varias técnicas que permiten obtener una representación del mundo a partir del análisis de imágenes obtenidas desde cámaras de video.

Debido a que la información visual es una de las principales fuentes de datos del mundo real, resulta útil el proveer a una computadora digital del sentido de la vista (a partir de imágenes tomadas con cámaras digitales o analógicas), que junto con otros mecanismos como el aprendizaje hagan de esta una herramienta capaz de **detectar y ubicar objetos en el mundo real**, objetivo principal de la Visión por Computadora.

#### 1.1.2. Aplicaciones.

Entre las aplicaciones en las que la Visión por Computadora ofrece importantes aportaciones, destacan las siguientes:

En el ramo industrial se aplica para la inspección automatizada; permite identificar que las piezas no tengan defectos.

En la geología permite estudiar las modificaciones de los causes de los ríos; también sirve para detectar bancos de pesca o variaciones en las corrientes marinas por cambios de clima.

En la medicina facilita el diagnóstico automático de las enfermedades del corazón, a partir de fotogramas del movimiento del mismo y de un análisis de las deformaciones que se producen.

En el ámbito militar se desarrollan complejas armas como misiles que se guían por sí solos por zonas previamente grabadas; en la identificación de aeropuertos, barcos, tanques o cualquier imagen tomada desde aviones o satélites.

Entre las operaciones básicas con las que se construyen diversas aplicaciones específicas de Visión por Computadora, podemos destacar las siguientes:

Identificación de objetos (por ejemplo identificar personas a partir de su fotografía).

Determinación de la posición en que se halla un objeto.

Establecimiento de relaciones espaciales entre varios objetos. (Para guiar a un robot en el ensamblaje de piezas).

Determinación de las coordenadas de puntos importantes de un objeto.

Realización de mediciones angulares.

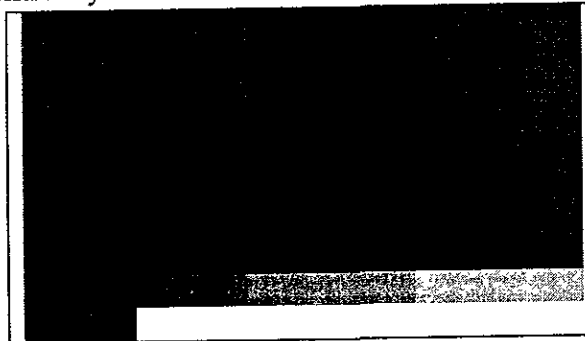
Mediciones tridimensionales (por ejemplo para ayudar a un robot a librar obstáculos)

## 1.2 ESTEREOSCOPIA.

### 1.2.1 Digitalización de imágenes.

Una forma de obtener una representación de una escena del mundo real consiste en colocar dos cámaras (conectadas a una computadora) paralelamente a una distancia de 3 cm entre ellas, para así poder obtener dos vistas de la misma escena. Estas vistas se conocen como imágenes estereoscópicas. Las imágenes pueden ser tomadas a color pero frecuentemente se manejan solo niveles de grises ya que para una gran variedad de tratamientos posteriores no es necesario la información de color.

La intensidad de la luz en un pixel en la imagen es proporcional a la cantidad de luz dirigida hacia la cámara por la superficie que se proyecta hacia el pixel. Esto depende de las propiedades de reflectividad de la superficie, la posición y la distribución de las fuentes de luz (Russell,95). También depende de las propiedades de reflectividad del resto de la escena ya que las superficies de la escena pueden servir como fuentes de luz indirecta al reflejar la luz recibida en la superficie en cuestión. Por esto la imagen ya digitalizada puede verse como una matriz con diferentes valores entre 0 y 255 si es en tonos de gris o como un conjunto de tres matrices con valores también entre 0 y 255 cuando es a color.



**Fragmento de imagen después de haber sido digitalizada.**

80	89	89	86	89	80	100	100	95	80
80	83	83	92	83	92	83	83	85	80
80	80	78	80	80	78	80	80	80	78
80	80	74	80	80	74	80	76	80	74
80	80	80	80	74	80	80	76	80	80
80	80	80	80	78	80	72	80	80	80
80	80	80	80	80	80	80	80	72	80
80	80	80	76	76	69	69	76	69	69
80	80	95	107	107	110	118	125	133	133
80	80	126	230	230	233	236	236	241	241

**Representación matricial de un fragmento de imagen después de haber sido digitalizada.**

### 1.2.2 Principios

Debido a que nuestros ojos están colocados con una distancia de separación, no ven exactamente la misma vista. Es tan sencillo de confirmar tan solo poniendo nuestro dedo índice a la altura de nuestra nariz a una distancia de unos 15 cm. Posteriormente se cierre el ojo derecho y se compare la ubicación del dedo con respecto a objetos distantes. Ahora se cierre el ojo izquierdo y se hace la misma comparación. Con esta sencilla prueba se podrá notar que los objetos se desplazan significativamente, indicando las dos diferentes vistas. Al mover el dedo lejos de la cara se reducirá el corrimiento mientras que al acercarlo el corrimiento se hace mayor.

El corrimiento entre la imagen izquierda y la derecha es llamado disparidad y está relacionado con la distancia entre el observador y a la característica que está siendo vista. De hecho, si el punto de fijación de las dos vistas está en el infinito, la disparidad es inversamente proporcional a la profundidad. Al comparar las disparidades relativas de las características correspondientes en las dos imágenes un proceso de la recuperación de la profundidad puede ser iniciado conduciendo a algunas formas de reconstrucción en tres dimensiones de la escena vista.

En el sistema visual humano, en donde la disparidad es calculada por el cerebro y utilizada para obtener profundidad por medio de triangulación, la recuperación de la profundidad no es un factor que necesite la existencia de información de alto nivel. La vista humana recobra fácilmente la profundidad relativa, pero falla al recobrar los valores exactos.

El principio de la estereoscopia es tener la vista del ojo izquierdo en y solo la imagen izquierda y tener al ojo derecho en la imagen derecha. Al hacer esto se producirá una tercera imagen en el centro. Esta tercera será la imagen en 3 Dimensiones, conocida comúnmente como imagen Ciclopeana. Este método de visión es conocido como vista paralela. El sistema visual humano posee un sistema de ejes coordenados convergente, es decir, en el que los ojos se orientan a un punto específico. El cálculo de la ubicación de este punto en este tipo de sistema coordenado es muy complejo y por lo tanto se hace computacionalmente muy costoso, por lo que es más común utilizar el sistema de ejes coordenados paralelos, en el que se parte de dos cámaras receptoras ubicadas a la misma altura.

### 1.2.3 Correspondencia entre imágenes estéreo.

Dadas un par de imágenes estéreo, el problema de encontrar la pareja de una característica de la imagen izquierda en la imagen derecha con el fin de encontrar la disparidad existente entre ellas, es conocido como el Problema de Correspondencia. Pero aunque parezca muy difícil no se tiene que buscar el punto correspondiente en una superficie de dos dimensiones, ya que los puntos que se corresponden están sobre las mismas líneas **epipolares** (Young, 89). Estas líneas corresponden a las intersecciones de un plano epipolar (plano a través de un punto en la escena y los puntos nodales de los dos ojos) con los planos de las imágenes izquierda y derecha.

Explotando esta restricción epipolar, se reduce una búsqueda inicial de dos dimensiones a una búsqueda de una sola dimensión.

Existen otras restricciones que nos pueden ayudar:

1 ) **Unicidad:** Un punto en una imagen puede corresponderse a lo más con un punto de la otra imagen, y se dice a lo más uno porque incluso podría no aparecer en la otra imagen.

2 ) **Posición continua de las superficies de una escena:** El hecho de que el mundo es continuo según la posición significa que puntos cercanos en la escena tienen valores cercanos de profundidad, y consecuentemente de disparidad, excepto en objetos de las orillas y en los que no aparecen en ambas.

#### 1.2.4 Métodos para resolver el problema de correspondencia.

Uno de los métodos es el de correlación, el cual asume que las imágenes izquierda y derecha son simplemente versiones recorridas una de otra y que la medición de este corrimiento es equivalente a determinar el paralaje estéreo (Con esto se asume que son superficies paralelas a la misma altura, proyección ortográfica e idénticas condiciones de luz como se ven con los dos ojos). El corrimiento correcto es aquel que da cero al hacer la integral del cuadrado de la diferencia entre las dos imágenes. Por lo tanto, la correlación (la integral de su producto) tiene un máximo. Esta medida puede ser localizada espacialmente al restringir la correlación a una ventana, típicamente especificada usando una Gaussiana. Desgraciadamente las vistas izquierda y derecha no son vistas recorridas una de otra. La simple correlación de ventana de la imagen de renglones no puede trabajar con escenas que están distorsionadas por la perspectiva de proyección o por diferencias interoculares en iluminación. El mismo pedazo de superficie aparecerá con diferentes brillos cuando es vista desde dos direcciones distintas y ese es un problema de los métodos de correlación.

Otra familia de aproximaciones está basada en encontrar bordes y después buscar puntos que se correspondan. Los bordes se estiman compatibles si están lo suficiente cerca en orientación y tienen la misma señal de contraste por toda la orilla. Se asume que los bordes que se corresponden obedecen al mismo orden de izquierda a derecha en cada imagen, lo que nos permite restringir el número de posibles correspondencias y esto lo lleva a ser uno de los algoritmos eficientes basados en programación dinámica. Sin embargo, con una aproximación basada en bordes la información de la profundidad resultante es escasa, debido a que está disponible solo en la posición de los bordes. Por lo que se necesitaría un paso posterior para interpolar la profundidad a través de las superficies en la escena.

En el presente trabajo nos enfocaremos a esta última aproximación basada en bordes y en un algoritmo llamado PMF en honor a las personas que lo desarrollaron (Pollard, Mahew y Frisby,91) del cual hablaremos a detalle más adelante.

Una tercera familia de algoritmos es la que se basa en la medición de la fase en los cuales la información de la fase en la salida de filtros lineales paso banda es útil para la medición de la disparidad binocular. Apoyado por el teorema de corrimiento de Fourier, se dice que una de las mayores ventajas de

la información de la fase es su estabilidad para afinar deformaciones y diferencias de contraste entre las vistas izquierda y derecha. En particular, la fase es considerablemente más estable que la amplitud. Esto es importante si la disparidad de un punto de la imagen va a ser usada como una aproximación a la disparidad entre puntos de la escena.

A pesar de la estabilidad general de la información de la fase, existen regiones de comportamiento patológico cercano a singularidades de fase donde la estimación de la disparidad no debería ser tan confiable, pero se pueden usar algunas restricciones para detectar estas regiones y descartar las mediciones incorrectas. Por último, la frecuencia local de la respuesta del filtro juega un papel clave tanto en la detección de vecindades singulares y la construcción de un predictor de disparidades más exacto.

## 1.3 MÚLTIPLES RESOLUCIONES

### 1.3.1 Concepto.

La aproximación de procesamiento jerárquico (piramidal) o análisis de imágenes a múltiples resoluciones paraleliza la operación del Sistema Visual Humano en los diferentes procesos, desde un simple nivel descriptivo hasta los niveles de realce y refinamiento. La aproximación jerárquica es aplicable a las tareas de segmentación, extracción de características, descripción y correspondencia de características entre los pares estereoscópicos. Una de los mayores beneficios de esta aproximación es que economiza los procesos de cómputo, motivo por el que resultó conveniente su utilización y análisis en el presente trabajo de tesis.

### 1.3.2 Filtros Gaussianos

El análisis de imágenes jerárquico o de múltiples niveles involucra la creación de una secuencia de imágenes a varias resoluciones espaciales o escalas. Una escala muy fina puede producir extraños detalles, y muy borrosa causa que características significantes no se detecten o se distorsionen. Además, las representaciones de imágenes a múltiples escalas deben ser consistentes, ya que las características en las diferentes resoluciones deben estar relacionadas.

Una representación espacio - escala de una imagen puede formarse al convolucionar la imagen de entrada (la imagen de más alta resolución, también denotada como  $f(x,y)$ ) con una función que contenga un parámetro de escala. El parámetro de reducción de escala lo denotamos por  $\sigma$ , y el respectivo núcleo de escala es  $g(x,y,\sigma)$  en donde  $\sigma$  es positivo. Un incremento en  $\sigma$  produce una reducción en la escala, este núcleo de escala es conocido también como parámetro de ancho de banda,  $\sigma'$ ; que es inversamente proporcional a  $\sigma$ . La salida de esta operación de espacio - escala es una superficie denotada por  $\Phi(x,y,\sigma)$ , la cual es solo definida sobre medio plano en el espacio  $(x,y)$ , la modificación de escala para una función en una dimensión esta dada por

$$\Phi(x,y,\sigma) = f(x,y) * g(x,y,\sigma) \quad (1-1)$$

donde  $*$  denota convolución, por lo que:

$$\Phi(x,y,\sigma) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(u,v) g(x-u, y-v, \sigma) dudv \quad (1-2)$$

En la ecuación 1-2 se produce un ensamble de imágenes como una función del parámetro de escala  $\sigma$ . Conforme  $\sigma$  se incrementa, el rango de valores que representan a  $x$  decrece. Una propiedad que notamos en esta ecuación es que conforme  $\sigma$  se aproxima a 0,  $\Phi(x,y,\sigma)$  se aproxima a  $f(x,y)$ , esto requiere que  $g(x,y,\sigma)$  se aproxime al impulso en  $\sigma=0$ .

Una función  $g(x,y,\sigma)$  para filtrado de espacio - escala, debe satisfacer los siguientes

puntos:

1.  $g(x, y, \sigma) = (1/\sigma)h(x, y, \sigma)$ . esto fuerza a definir un parámetro de escala.
2.  $g(-x, y, \sigma) = g(x, y, \sigma) = g(x, -y, \sigma)$ . es decir se requiere que  $g$  sea simétrica.
3.  $g(x, y, \sigma)$ , para todo  $\sigma > 0$ , es normalizada, es decir,

$$\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} g(x, y, \sigma) dx dy = 1$$

Una función que satisface los anteriores requerimientos es el kernel Gaussiano, esto es:

$$g(x, y, \sigma) = (1/2\pi\sigma^2) e^{-\frac{1}{2}\left(\frac{x^2+y^2}{\sigma^2}\right)}$$

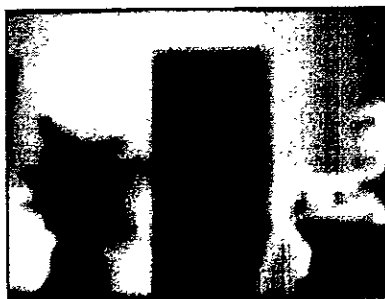
Varios de los métodos de reducción de escalas, no satisfacen totalmente los requerimientos arriba señalados, sin embargo son aproximaciones de la función Gaussiana ideal. A continuación se muestra una imagen a la que se le aplicaron 3 filtros Gaussianos:



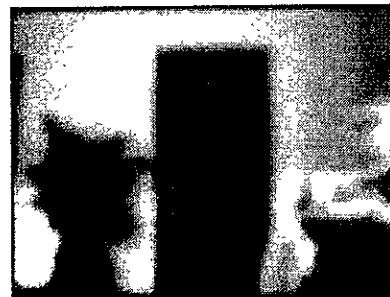
Imagen Original.



Aplicando 1 filtro Gaussiano.



Aplicando 2 filtros Gaussianos.

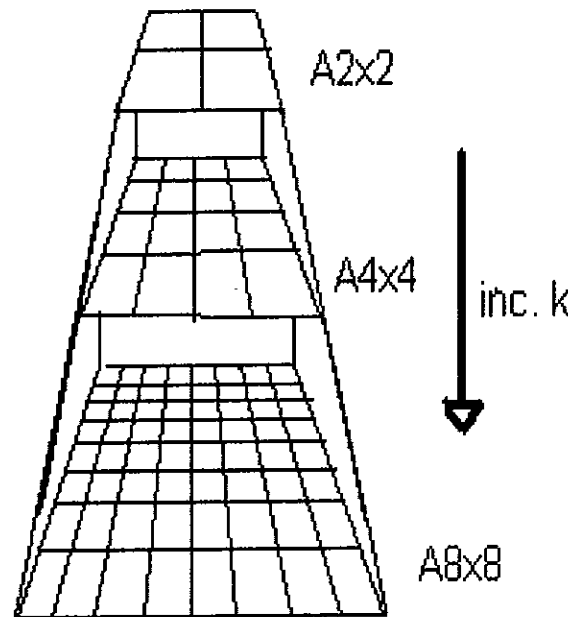


Aplicando 3 filtros Gaussianos.



### 1.3.3 Submuestreo.

El submuestreo o concepto de árbol cuádruple ha sido introducido como una técnica de descomposición espacial sin estar relacionado a la representación de intensidades con niveles de gris en cada nivel (Schalkooff,84). Se utiliza una simple aproximación de valores promedios (los cuales son procesos de computo simples). El objetivo es la creación de una secuencia de arreglos de dos dimensiones con variables resoluciones como se muestra en la siguiente figura:



**Figura de la Estructura Piramidal**

Una pirámide,  $P$ , es un conjunto de imágenes.

$$P = \{A_{1 \times 1}, A_{2 \times 2}, A_{4 \times 4}, \dots, A_{256 \times 256}, \dots, A_{m \times m}\} \quad (1-3)$$

Donde  $m = 2^n$ , en esta representación piramidal,  $A_{m \times m}$  es el tope o la imagen de más alta resolución;  $A_{1 \times 1}$  es una representación de un píxel de la imagen de resolución reducida. La pirámide puede ser vista como una entidad de 4-D en donde la intensidad es una función de tres argumentos: un asignador de nivel, y un conjunto de índices con soporte dependiente de nivel, esto es la pirámide se denota como  $f(i, j, k)$ , donde  $i, j$  son índices espaciales y  $k$  es el nivel de la pirámide.

La imagen tiene una extensión espacial de  $2^n \times 2^n$  puntos, es decir el total puntos o píxeles es de  $2^{2n}$ , o  $4^n$ , el grado base de cada nivel de la pirámide es de 4, el primer nivel de la pirámide representa la imagen entera (imagen original), cada nivel excluyendo la base debe tener 4 nodos,

esto es el nivel  $k$ -ésimo contiene  $4^k$  nodos.

La creación de una pirámide y representación de imagen se muestra a continuación:

$$f(i,j,k) = \frac{1}{4} \sum_{p=0}^1 \sum_{q=0}^1 f(2i+p, 2j+q, k+1) \quad (1-4)$$

donde para una imagen  $2^n \times 2^n$ ,

$$k \in [0, n]$$

y en el nivel  $k$ , los índices son restringidos a

$$0 \leq i, j \leq 2^k - 1$$

El procedimiento de la ecuación 1-4 se repite hasta que se tenga una imagen del nivel deseado (o que consista de un solo pixel), en la figura siguiente podemos ver una imagen a varias resoluciones:



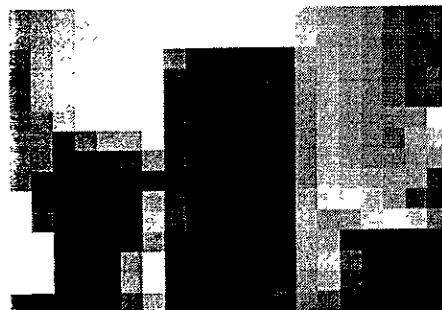
(a)



(b)



(c)



(d)

Pirámide de imágenes a 4 resoluciones.

(a) 160 x 120    (b) 80 x 60    (c) 40 x 30    (d) 20 x 15

## 1.4 MÉTODOS DE DETECCIÓN DE CONTORNOS

Para el análisis de imágenes digitales la extracción de los bordes de dichas imágenes resulta de gran utilidad. Podemos considerar que existe un borde cuando en un conjunto de puntos de la imagen aparece un cambio significativo en la intensidad luminosa (Sanfeliu,87). Existen básicamente dos formas de obtener los bordes de una imagen, estas son:

1. - Método basado en las primeras derivadas. En este método primero se miden los cambios de intensidad en todos los puntos de la imagen, haciendo uso de la primera derivada, después se seleccionan como puntos de bordes aquellos puntos en los que el cambio de intensidad rebasa algún umbral preestablecido.

Una imagen puede representarse matemáticamente por una función  $f$  cuyo valor en el punto  $(x, y)$  representa la iluminación existente en ese punto, el cambio de iluminación de un punto en la imagen está representado gráficamente por la pendiente que allí tiene la superficie que representa a  $f$ , y normalmente resulta diferente según sea la dirección considerada, la dirección en la que el cambio es más intenso esta dada por el vector gradiente en ese punto.

$$\nabla f(x, y) = \frac{\partial f}{\partial x}(x, y), \quad \frac{\partial f}{\partial y}(x, y)$$

Y la magnitud del campo esta dada por el módulo del gradiente, por lo que la medición de los cambios de iluminación se puede realizar por medio este. Para evitar que estos procesos consuman mucho tiempo de procesador, normalmente se hace solo una aproximación del gradiente, dicha aproximación esta dada por  $|\nabla f(x, y)| = |\nabla_i| + |\nabla_j|$ , en donde  $\nabla_i$  y  $\nabla_j$  son las aproximaciones discretas de las derivadas parciales. A continuación se describen cuatro de los principales métodos basados en la primera derivada.

*Método del gradiente.* Consiste en utilizar una aproximación discreta de la definición matemática de las dos derivadas parciales

$$\nabla_i = f(i+1, j) - f(i, j)$$

	$j$	$j+1$
$i$	-1	0
$i+1$	1	0

$$\nabla_j = f(i, j+1) - f(i, j)$$

	$j$	$j+1$
$i$	-1	1
$i+1$	0	0

*Método Roberts.* Es equivalente al anterior, si se rotan los ejes 45 grados, las máscaras serán entonces:

	$j$	$\nabla i$
$i$	-1	0
	0	1

	$j$	$\nabla j$
0	1	0
-1	0	1

*Método Prewitt.* Para aproximar la derivada sobre el pixel  $(i, j)$  en la dirección horizontal, se toma en consideración no sólo el cambio de intensidad hacia la derecha en ese punto, sino que también el que aparece hacia la izquierda, además lo promedia con los valores que obtiene en la misma posición de las filas contiguas encima y debajo; de este modo se llega a una estimación de la derivada horizontal que es muy resistente al ruido presente en las imágenes. Para estimar la derivada en dirección vertical se usa un procedimiento análogo. Las máscaras que utiliza son:

	$j$	$\nabla i$
-1	-1	-1
0	0	0
0	1	1

	$j$	$\nabla j$
-1	0	1
-1	0	1
-1	0	1

*Método de Sobel.* Sigue la misma lógica que el anterior, sólo que da más peso a la variación de intensidad que aparece en la línea que contiene al pixel sobre el que se está estimando la derivada. Eso se implementa duplicando los coeficientes de esa línea. Las máscaras utilizadas son:

	$j$	$\nabla i$
-1	-2	-1
0	0	0
1	2	1

	$j$	$\nabla j$
-1	0	1
-2	0	2
-1	0	1

2.- El método basado en la segunda derivada consiste en determinar directamente en que puntos de la imagen aparecen los picos de los cambios de iluminación utilizando las segundas derivadas. En los puntos en que los cambios de intensidad tiene un máximo, se anula la derivada segunda de la intensidad tomada en la dirección en que se tiene lugar la transición de claro a oscuro, para esto se puede cambiar el cálculo de las derivadas por el Laplaciano.

$$\nabla^2 f(x, y) = \frac{\delta^2}{\delta x^2} f(x, y) + \frac{\delta^2}{\delta y^2} f(x, y)$$

Este operador tiene la ventaja de que sólo exige calcular dos derivadas segundas, para calcularlo en imágenes digitales se puede aproximar utilizando la máscara siguiente:

	$j$		
	0	1	0
$i$	1	-4	1
	0	1	0

Este método identifica como puntos de contorno aquellos de coordenadas  $(i, j)$  en los que el Laplaciano se anula; aunque puede resultar preferible tomar aquellos en los que cambia de signo. Otras veces sólo se toman aquellos en los que además de anularse el Laplaciano, el módulo del gradiente rebasa un cierto umbral. El operador laplaciano, utilizado como detector de puntos de bordes, presenta ventajas y desventajas, entre las ventajas está producir puntos de contorno que al ser agrupados forman trazos finos, por lo que no se requieren otros procesos de adelgazamiento. Entre los inconvenientes está no ofrecer directamente información sobre la dirección ni el sentido en que tiene lugar la transición de claro a oscuro en cada punto de contorno, ni sobre la magnitud de dicha transición.

## CAPITULO 2 : ALGORITMO DE CORRESPONDENCIA PMF

En este capítulo se da la explicación de los conceptos en los que está basado el funcionamiento del algoritmo, como son el gradiente de disparidad, características y detalles del algoritmo, el ciclo de ejecución y una explicación detallada del pseudocódigo.

### 2.1 GRADIENTE DE DISPARIDAD

La información de profundidad útil acerca de una escena puede ser fácilmente recuperada a partir de las disparidades que existen entre los puntos que forman al par de imágenes analizadas, la existencia de dichas disparidades es debido a que ambas son tomadas desde diferentes perspectivas, es decir, existe un desplazamiento entre ellas y que es normalmente sobre el eje de las abscisas (eje X); Sin embargo medir estas disparidades requiere una solución al difícil "Problema de Correspondencia Estéreo", que en nuestro caso está basado en el cálculo del Gradiente de Disparidad(Young,89).

Por estudios realizados al sistema visual humano se sabe que si dos puntos (uno captado por el ojo izquierdo y otro por el derecho) están simultáneamente fusionados binocularmente por el sistema visual humano, es decir ambos puntos son el mismo en la escena física, entonces el radio de la diferencia de disparidad entre los puntos a su separación ciclopeana no debe exceder un límite de 1. Este radio es conocido como el gradiente de disparidad y la existencia de un límite de este gradiente implica que incluso pequeñas diferencias de disparidad no producirán fusión si la separación espacial entre puntos es también pequeña, estos conceptos se explican a continuación ya con el enfoque para implantarse en computadora digital.

Consideremos un estereograma simple compuesto de dos puntos en cada campo, entendiendo por estereograma a un par de imágenes estéreo que previamente fueron sometidas a detectores de contornos para tener en ellas solo los principales rasgos de la escena de los que conocemos su ubicación en dos dimensiones.

El gradiente de disparidad está definido por un par de puntos que se corresponden, donde cada uno de estos asocia un rasgo en la imagen izquierda con uno en la imagen derecha. Si dos rasgos se corresponden mutuamente entonces el rasgo de la imagen izquierda y el de la derecha fueron producidos por el mismo rasgo en el mundo real (escena). Sin embargo es necesario analizar muchos rasgos que no son correctos para establecer los correctos.

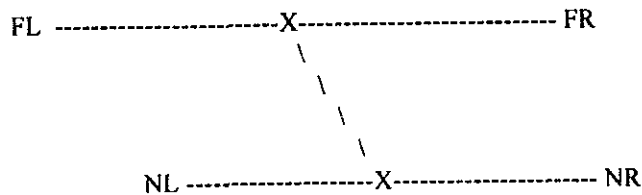
Supongamos que el rasgo FL en la imagen izquierda es tentativamente correspondido con el rasgo FR en la imagen derecha, y el rasgo NL en la izquierda es correspondido con la característica NR en la derecha. Entonces la disparidad para el par

correspondido FL  $\leftrightarrow$  FR es solamente la coordenada X de FL menos la coordenada X de FR . La diferencia de disparidad para el par de rasgos correspondidos FL  $\leftrightarrow$  FR y NL  $\leftrightarrow$  NR es la disparidad entre el par F menos la disparidad entre el par N (DD).

Ahora imaginemos las dos imágenes sobrepuestas (figura 2.1), la Separación Ciclopeana (SC) es la distancia del punto medio de la línea que une FL y FR al punto medio de la línea que une NL y NR.

Por lo tanto el Gradiente de Disparidad (GD) es el valor absoluto de la diferencia de disparidades dividido entre la separación ciclopeana.

Esquemáticamente, sobreponiendo las dos imágenes, los dos rasgos que tentativamente se corresponden se verían de la siguiente forma :



**Figura 2.1**

En conclusión, la Diferencia de Disparidad (DD) es la diferencia en longitud entre las dos líneas horizontales. La Separación Ciclopeana es la longitud de la línea inclinada y el Gradiente de Disparidad es el radio de estas dos.

Supongamos que ambos puntos se corresponden mutuamente, entonces los dos rasgos están involucrados, uno produciendo FL y FR y el otro produciendo NL y NR, el Gradiente de Disparidad para estas correspondencias será pequeño (menos de 1) solamente si la línea que une a los dos rasgos esté en un ángulo recto a la línea de vista, esto es que esté orientado exactamente hacia el observador, la relación exacta entre la pendiente de la línea y el gradiente de disparidad depende de la distancia interocular y la profundidad de los rasgos.

El algoritmo PMF opera sobre estas bases que para la mayoría de pares que sean correspondidos mutuamente, el gradiente de disparidad debe ser muy pequeño (menor que 1). Los posibles candidatos en pares que satisfagan este criterio serán aceptados.

## 2.2 DETALLES DEL ALGORITMO.

### 2.2.1 Características

Además de la restricción del Gradiente de Disparidad, el algoritmo PMF explota otras dos restricciones: (i) la restricción epipolar, un límite sobre las posibles colocaciones de los puntos que se corresponden; y (ii) la restricción de unicidad, un límite sobre el número de correspondencias permitidas para una entidad sencilla de una imagen(Mahew,91).

**i) Restricción Epipolar :** Para la clase de geometrías de las imágenes estéreo con las que se trabaja (aquellas en las que los ejes ópticos de los dos dispositivos captadores de imágenes están sobre el mismo plano) todos los primitivos que se corresponden mutuamente aparecen en pares izquierda / derecha de líneas epipolares rectas. Aquellos puntos a lo largo de un miembro del par epipolar solo pueden corresponderse con puntos situados a lo largo del otro miembro, y viceversa. En el caso especial donde los ejes principales son de hecho paralelos, todos los pares epipolares serán horizontales y los puntos correspondidos serán encontrados sobre los rasters correspondientes.

El algoritmo PMF usa esta restricción para limitar la búsqueda de posibles correspondencias a una sola dimensión. En versiones actuales del PMF se asume que la geometría de la cámara es paralela o al menos algo muy aproximado, por ejemplo, con imágenes cuadradas de 128 x 128 pixeles y cubriendo un ángulo visual a lo más de  $10^\circ$ , una disparidad vertical de no más de medio pixel existirá en las esquinas de la imagen.

En el PMF todas las correspondencias potenciales dentro de un rango de disparidad largo de  $\pm 30$  pixeles que satisfacen un criterio de correspondencia para primitivos de imágenes izquierda / derecha son seleccionadas por disambigüedad subsecuente. La elección del criterio de correspondencia depende de la naturaleza de los primitivos adoptados. Para estereogramas artificiales simplemente se usan puntos definidos cuando el estímulo está siendo creado. Para imágenes naturales se ha encontrado conveniente primitivos tipo bordes dados por cruces por cero.

En la siguiente figura se puede ver lo que es la Geometría Epipolar:



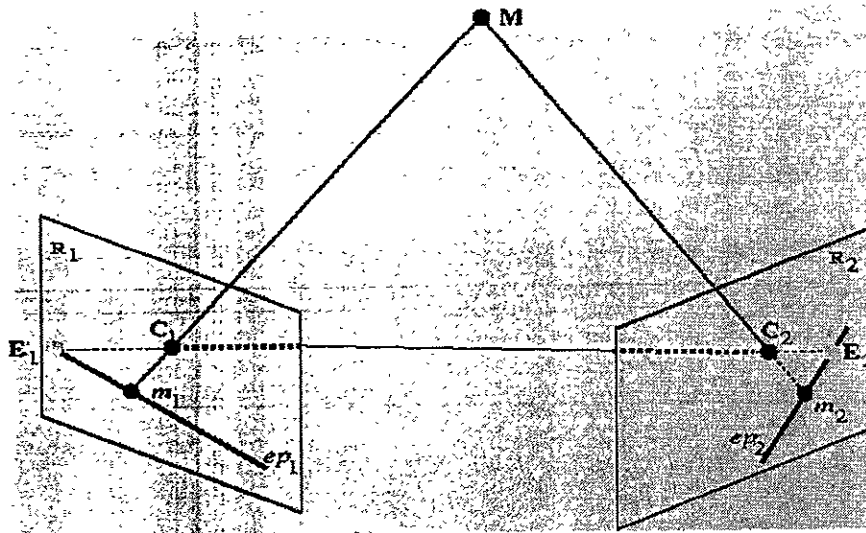


Figura 2.2

Sean  $C_1$  y  $C_2$  los centros ópticos de dos cámaras cuyos planos imagen  $R_1$  y  $R_2$  se encuentran a la distancia  $f_1$  y  $f_2$  de sus respectivos centros ópticos. Sea  $M$  un punto en el espacio, siendo  $m_1$  y  $m_2$  sus proyecciones en los planos imagen  $R_1$  y  $R_2$  respectivamente.

En la Figura 2.2 se puede observar que, dado el punto  $m_1$  del plano imagen  $R_1$ , todos los posibles puntos  $M$  que pueden haberse proyectado en él se encuentran sobre la recta  $(m_1, C_1)$ . Como consecuencia directa, todas las posibles correspondencias del punto  $m_1$  en el plano  $R_2$  se encuentran sobre la proyección en el plano imagen  $R_2$ , de dicha recta. La proyección de  $(m_1, C_1)$  en el plano  $R_2$  es la línea infinita  $ep_2$ , denominándose línea epipolar del punto  $m_1$  en la imagen  $R_2$ . Todas las líneas epipolares de la imagen  $R_2$  pasan por el punto  $E_2$ , llamado epípolo, que es la intersección de la recta que une los centros ópticos de ambas cámaras  $(C_1, C_2)$ , con el plano  $R_2$ . Por tanto, la restricción epipolar establece que, dado un punto  $m_1$  del plano  $R_1$ , sus posibles correspondencias en el plano  $R_2$  se encuentran sobre la línea epipolar correspondiente.

**ii) Restricción de Unicidad:** Esta restricción se refiere a que un rasgo se podrá corresponder a lo más con un rasgo.

**El algoritmo PMF tiene básicamente dos etapas de procesamiento:**

#### **Etapas 1:**

El peso de correspondencia de cada rasgo candidato es obtenido de la suma de las contribuciones recibidas de todos los candidatos potenciales en su vecindad que satisfacen el gradiente de disparidad de 1 con respecto a éste. Debido a que la probabilidad de que una vecindad a corresponder que cae dentro del límite se incrementa con la lejanía de la distancia del punto correspondiente, la contribución del peso de un rasgo es inversamente

proporcional a su distancia con este. La unicidad es explotada en esta etapa al requerir que a lo máximo un rasgo asociado con un solo candidato en una u otra imagen hace una contribución a la asignación del peso.

### **Etapa 2 :**

En una primera comparación algunos rasgos que tengan los pesos más altos para ambos primitivos de las dos imágenes que los formaron son escogidos inmediatamente como correctos, por ejemplo los rasgos son seleccionados cuyos primitivos no tienen pesos más altos con algunos otros rasgos que pueden formar. Entonces, de acuerdo a la restricción de unicidad, todos los otros rasgos asociados con los dos primitivos que formaron cada candidato escogido son eliminados para ser considerados después. Esto permite a los rasgos posteriores que no fueron previamente aceptados o eliminados, ser seleccionados como correctos porque ahora tienen pesos más altos para ambos primitivos. Usualmente solo se necesitan cuatro o cinco iteraciones para propagar la restricción de unicidad.

### **2.2.2 Ventajas y Desventajas**

La principal ventaja de este algoritmo radica en que sus primitivos son los bordes de las imágenes por ejemplo en una imagen de 320 x 240 pixeles, que contiene 76,800 posibles puntos, solo se trabajará con aproximadamente una centésima parte de estos puntos, (esto depende del número de objetos en la escena) lo que implica un procesamiento bastante rápido en comparación con algoritmos que analizan las imágenes en tonos de grises y que tienen que analizar todos los puntos. Además con la ayuda de las restricciones epipolar y de unicidad se reduce el tiempo de proceso y por lo tanto se hace un algoritmo ideal para poder trabajar en tiempo real, característica necesaria para aplicaciones de navegación principalmente.

La principal desventaja es que no se logra un 100% de rasgos correspondidos, sin embargo el porcentaje de correspondencia es alrededor del 90%, datos suficientes para obtener resultados muy exactos (como se ve en el capítulo 4 de resultados y evaluación) en aplicaciones de navegación robótica.

Algo que vale la pena señalar es que por el tipo de procesos a que son sometidas las imágenes originales se pierde mucha información, es decir, solo recuperamos la información correspondiente a los contornos de los objetos en la imagen que nos ayuda a obtener la ubicación de estos, y el resto de la información que describe principalmente color y textura se pierde, esta limitante no es una desventaja, por que como ya se ha mencionado todo depende de la aplicación, que en nuestro caso es conocer en donde están los objetos y no cómo son. Aún no existe una técnica que nos permita recuperar todo este tipo de información (ubicación de los objetos, color, textura, etc.) al mismo tiempo, por lo que sería necesario una conjugación de varias de ellas.

2.3 DIAGRAMA DE FLUJO DEL ALGORITMO PMF

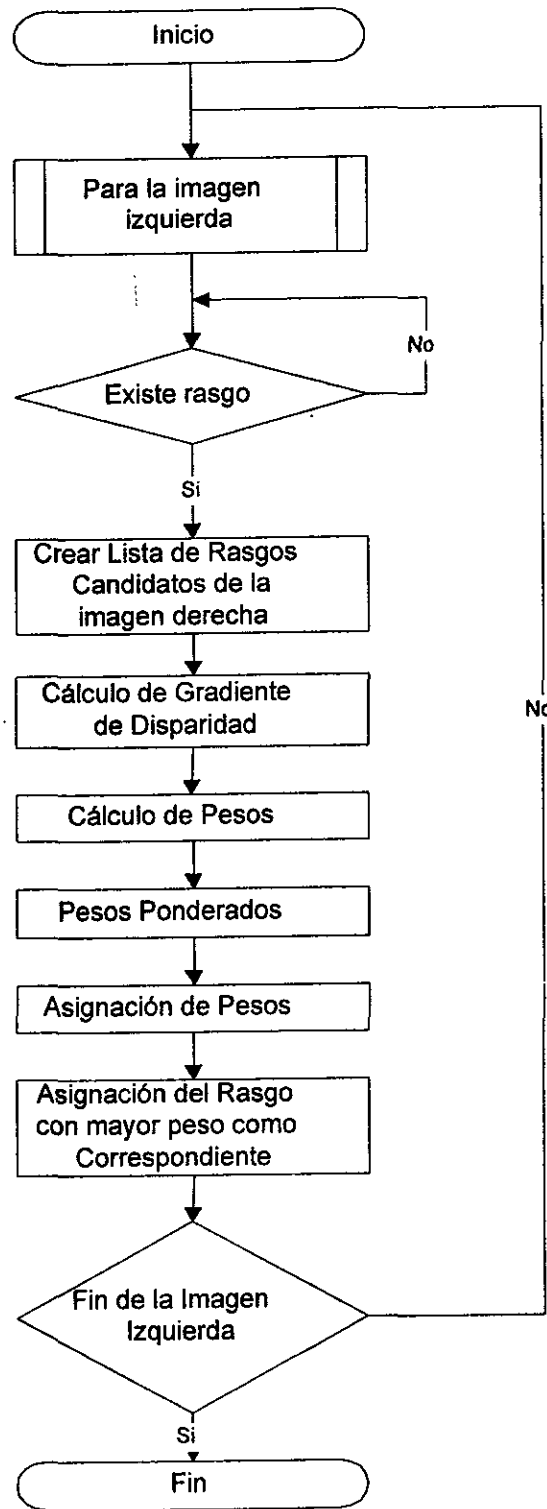


Figura 2.3

## 2.4 PSEUDOCÓDIGO.

En este punto se da la descripción del algoritmo de correspondencia PMF con un enfoque directo a lo que es su implantación en computadora digital, por lo que se explica en varios bloques. Cabe recordar que el algoritmo se aplica después de tener el par de imágenes estéreo, las que a su vez ya han sido reducidas a un par de imágenes binarias que contienen solo los rasgos de la escena que será analizada.

Puntos a seguir para implantar el algoritmo de correspondencia PMF en computadora digital :

A.- Para cada punto de la imagen izquierda  $P_i(x_i)$  seleccionar una lista de puntos  $P_d(x_d)$  en la imagen derecha que serán los candidatos a ser correspondidos con el punto de la imagen izquierda ( $P_i(x_i)$ ).

Para hacer la selección de candidatos se pueden tomar varios criterios, por ejemplo un caso sería si las imágenes de rasgos no fueran binarias, los candidatos podrían ser todos aquellos rasgos o puntos cuyo valor de tonalidad sea semejante a la tonalidad del punto a corresponder, otra posibilidad es seleccionar un cierto número de puntos (el número de puntos va a depender del tamaño de las imágenes y el desplazamiento que halla entre ellas) llamados candidatos que se encuentren con posición  $x_d$  que es menor a la posición  $x_i$  del punto analizado, es decir, sabemos que el punto de la imagen derecha tendrá siempre una coordenada en X menor que la del punto de la imagen izquierda, esto debido al desplazamiento que existe entre las dos imágenes que fueron tomadas desde distintas perspectivas y por lo tanto los objetos en la imagen derecha tienen coordenadas en X menores que los mismos objetos de la imagen izquierda, este último criterio fue el que adoptamos para la implantación del algoritmo, esto para lograr menores tiempos de proceso que es nuestro principal objetivo.

Debemos tener en cuenta que la restricción de disposición paralela entre imágenes nos dice que cualquiera que sea el criterio a tomar, los candidatos deben estar sobre la misma línea (raster) de la imagen opuesta, por lo que en el peor de los casos el criterio de selección de candidatos podría ser que los candidatos fueran todos los puntos del raster contrario. Para que el concepto de raster quede más claro tenemos la siguiente figura.

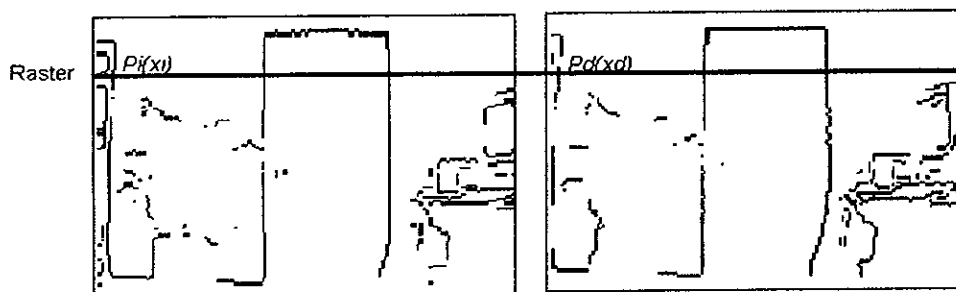


Figura 2.4

**B.- Asignar pesos a los rasgos de la imagen izquierda.**

Asignar pesos involucra calcular un peso ponderado para cada candidato a ser correspondido :

**B.1.- Por cada rasgo de la imagen izquierda ( $P_i$ )**

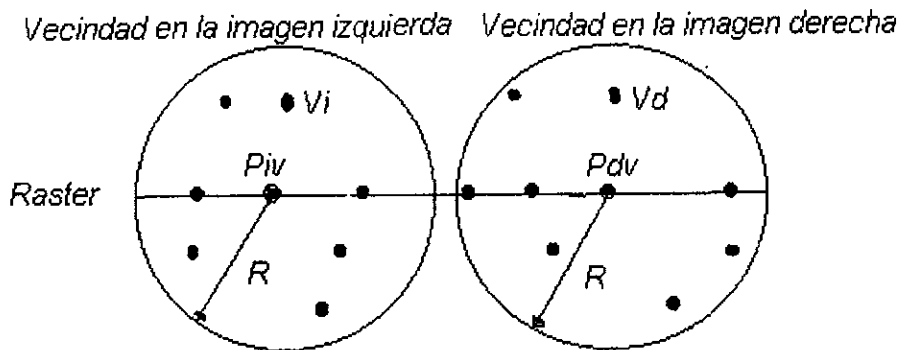
**B.1.1.- Por cada rasgo candidato ( $P_d$ ) a ser correspondido con  $P_i$**

**B.1.1.1.- Calcular Peso Ponderado entre  $P_i(x_i)$  y  $P_d(x_d)$**

Para el cálculo de peso ponderado debemos tomar en cuenta que el algoritmo PMF está soportado por vecindad, y es en el cálculo de pesos ponderados en donde se explota esta característica. A continuación se explica el concepto de vecindad.

Consideremos los puntos  $P_i$  (punto de la imagen izquierda que está siendo analizado) y  $P_d$  (punto candidato de la imagen derecha), entonces  $P_{iv}$  y  $P_{dv}$  se consideran puntos base para definir la vecindad y están dados por las siguientes ecuaciones:

$$P_{iv} = P_i(x_i) \text{ y } P_{dv} = P_d(x_d)$$



**Figura 2.5**

En donde  $R$  es el radio de vecindad (el valor del radio de la vecindad está dado en píxeles y se selecciona de acuerdo al tamaño de las imágenes, así como el desplazamiento que existe entre los rasgos de la imagen izquierda respecto a los de la imagen derecha),  $V_i$  y  $V_d$  son los vecinos de la izquierda y de la derecha respectivamente. Todo punto o rasgo que se encuentre dentro del área definida por el radio de vecindad se consideran vecinos del rasgo base.

Como se explicó en el punto uno de este capítulo, el cálculo del Gradiente de disparidad requiere de dos pares de puntos, para este ejemplo son  $(P_{iv}, V_i)$  y  $(P_{dv}, V_d)$ , estos pares de puntos deben cumplir que  $V_i$  y  $V_d$  están en el mismo raster al igual que  $P_{iv}$  y  $P_{dv}$ .

Una vez teniendo claro el concepto de vecindad para el algoritmo PMF, continuemos con el procedimiento del cálculo del peso ponderado, el procedimiento es el siguiente:

\* Peso Ponderado = 0

\* Por cada vecino ( $V_i$ ) del punto  $Piv$

\* Contribución = 0

\* Por cada  $V_d$  ubicado en el mismo Raster que  $V_i$

\* Calcular Peso dado por  $(Piv, V_i(x_i))$  y  $(Pdv, V_d(x_d))$

\* Si  $Peso > Contribución$

\*  $Contribución = Peso$

\*  $Peso Ponderado = Peso Ponderado + Contribución$

El cálculo del Peso es la función que hace válida la condición que el Gradiente de Disparidad dado por  $(Piv, Pdv)$  y sus respectivos vecinos sea menor o igual a uno y asigna un Peso al par de puntos analizado y está dado por el siguiente procedimiento :

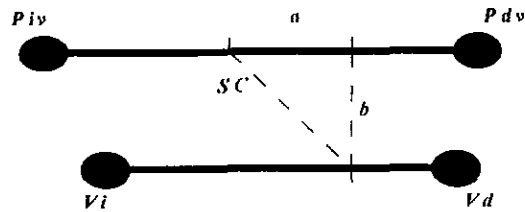
\* Si el Gradiente de disparidad dado entre  $(Piv, Pdv)$  y sus vecinos  $(V_i, V_d) > 1$

\*  $Peso = 0$

\* Si no es así

\*  $Peso = 1 / Distancia entre Piv y V_i$

en donde a su vez el Gradiente de Disparidad (GD) esta dado por :


**Figura 2.6**

Disparidad ( $DP$ ) =  $Piv - Pdv$

Disparidad entre vecinos ( $DV$ ) =  $Vi - Vd$

Diferencia de Disparidades ( $DD$ ) =  $DP - DV$

$a = (DP / 2) - (DV / 2)$

$b$  = Separación entre rasters

Separación Ciclopeana ( $SC$ ) =  $\sqrt{a^2 + b^2}$

Gradiente de disparidad ( $GD$ ) =  $DD / SC \leq 1$

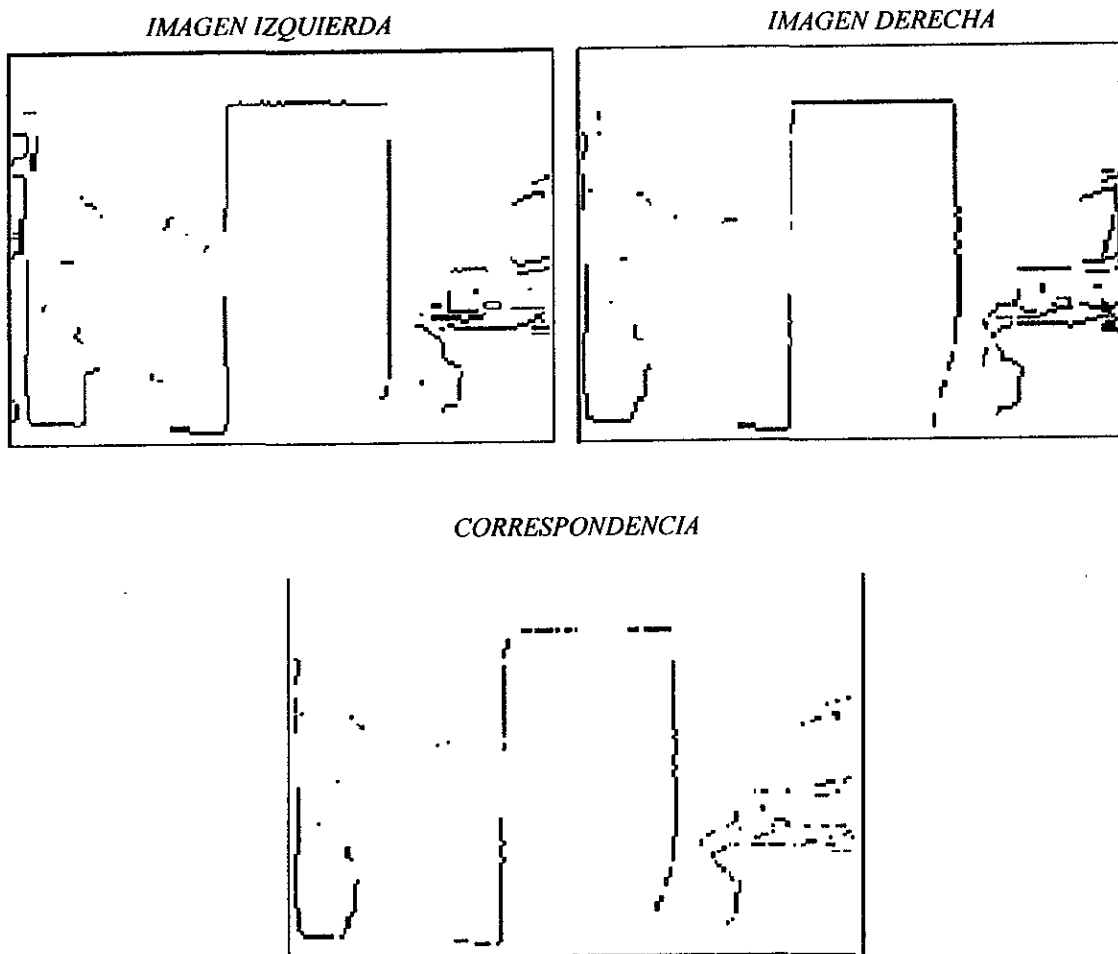
**C.-** Crear una lista ( $Lr$ ) con los Pesos Ponderados de los candidatos de  $Pi$ .

**D.-** De la lista de pesos ponderados ( $Lr$ ) se selecciona aquel que tenga mayor peso, debido a que es el mejor candidato, y por lo cual el rasgo correspondiente al punto  $Piv$  que se analizó.

**E.-** Una vez que cada rasgo de la imagen izquierda tiene rasgo correspondiente en la imagen derecha, se almacenan sus posiciones, las cuales son útiles, ya que con ellas generamos nuestro mapa de disparidades.

## 2.5 RESULTADOS OBTENIDOS CON EL ALGORITMO PMF.

El algoritmo fue programado en MATLAB para Windows, a continuación se muestran las imágenes resultantes



**Figura 2.7**

En el resultado anterior podemos observar que en la imagen de correspondencia solo están los puntos que existen tanto en la imagen izquierda como en la derecha, es decir, un rasgo que exista en la imagen izquierda pero que no este contenido en la imagen derecha, no aparecerá en la imagen de correspondencia, en los siguientes resultados se observa más claro:



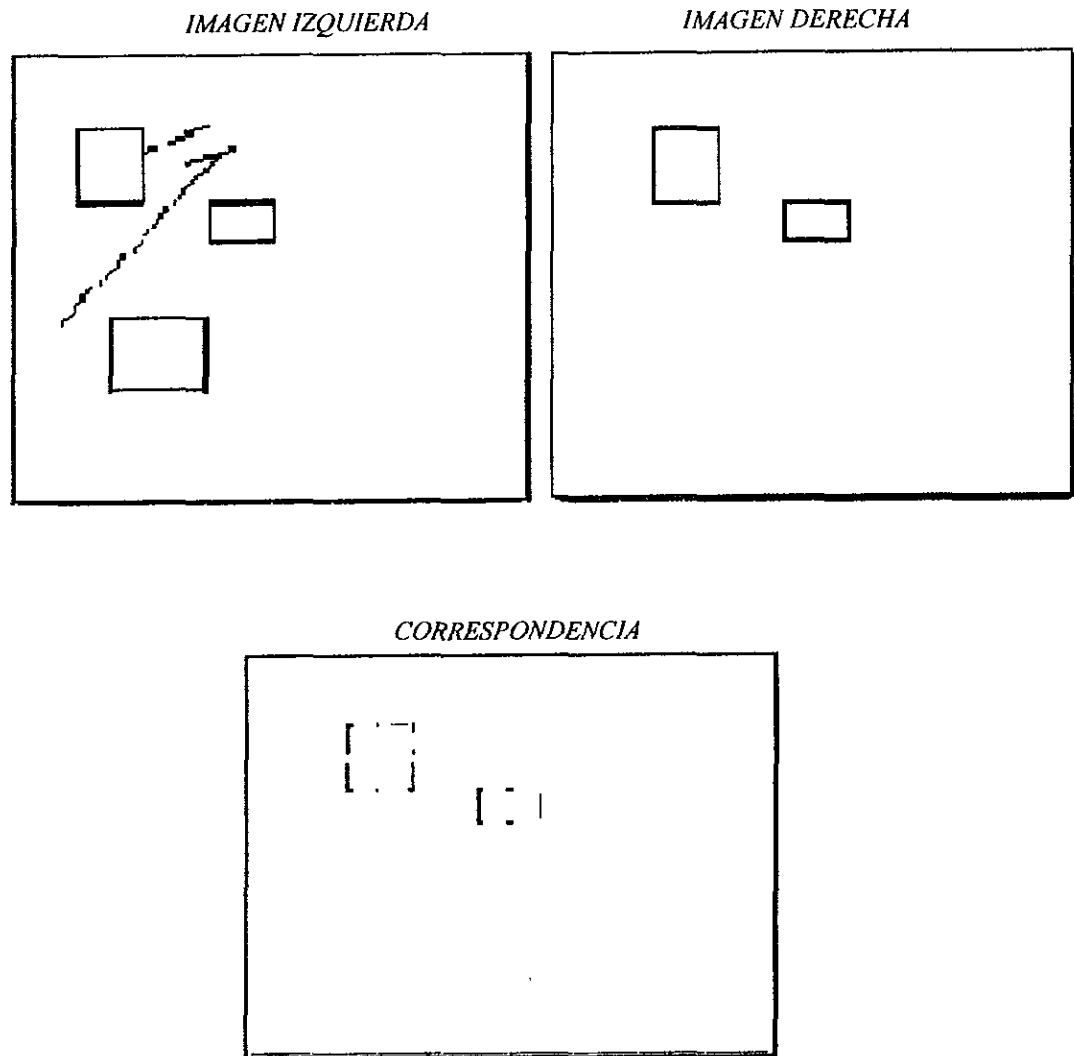


Figura 2.8

## CAPÍTULO 3 CORRESPONDENCIA PMF A MÚLTIPLES RESOLUCIONES.

### 3.1 FUNDAMENTACIÓN.

Debido a que el principal objetivo del presente trabajo es encontrar la manera de mejorar el tiempo de procesamiento del algoritmo PMF, nos basamos en dos conceptos muy importantes que son parte del Procesamiento Digital de Imágenes a Múltiples Resoluciones, la aplicación de filtros Gaussianos y el submuestreo de imágenes.

#### 3.1.1 El Espacio Escala

Witkin y Koenderink (1984) desarrollaron la Teoría del Espacio - Escala y demostraron que mediante la difuminación de una imagen con una función Gaussiana cuya desviación estándar  $\sigma$  (llamada parámetro de escala) varía continuamente, se obtiene una descripción única de la imagen. Esta descripción consiste entonces en una secuencia continua de imágenes que van perdiendo detalle conforme  $\sigma$  aumenta, hasta llegar a una mancha uniforme cuando  $\sigma \rightarrow \infty$ . La función Gaussiana es la única que garantiza que todos los rasgos que existen en esta representación para una  $\sigma_1$  corresponden con rasgos en otra representación  $\sigma_2$ , en donde  $\sigma_2 < \sigma_1$ , es decir, conforme  $\sigma$  aumenta, no se generan rasgos espurios ajenos a la imagen original.

Esta representación multi - escala ha encontrado muchas aplicaciones en el procesamiento de imágenes, tales como codificación (Burt y Adelsen, 1983) y en restauración (Escalante,1992). En este capítulo mostramos cómo la representación multiescalar puede simplificar grandemente en problema de correspondencia entre imágenes estéreo.

El principio fundamental consiste en analizar representaciones a baja resolución del par de imágenes estéreo, con el fin de remover detalles espurios de las imágenes que no son necesarios para la estimación de correspondencias de objetos en imágenes estéreo, y que además prolongan el tiempo de cálculo.

Como se mencionó anteriormente, al aplicar los filtros Gaussianos se obtienen imágenes difuminadas las cuales van presentando menos detalles definidos, es decir, los rasgos de la imagen son más difíciles de distinguir, a medida que el orden del filtro es mayor. Partiendo de aquí, al aplicar detección de contornos a cada una de estas imágenes, la de mayor grado de difuminación presentará menos rasgos (puntos), y ya que en ambas imágenes aparecerá el mismo conjunto de rasgos dentro del cual están los que se corresponden, cada punto tendrá a los vecinos más cercanos y se evitarán cálculos con

vecinos más lejanos que normalmente son poco probables de ser el rasgo correspondiente. Con esta Pirámide de imágenes de Correspondencia se busca evitarle trabajo a los niveles de mayor resolución, al ir encontrando puntos correspondientes, los cuales no serán procesados en estos niveles.

Otra de las ventajas de este método es que aunque el proceso de cálculo de la pirámide Gaussiana involucra subprocesos como son convolucionar las imágenes con el filtro para cada nivel, el tiempo de procesamiento es mucho menor en comparación a las operaciones realizadas en el algoritmo PMF ya que este último se basa en una serie de operaciones (cálculo de pesos, contribuciones, disparidades, entre otros) de vecindades por cada rasgo que encuentra en la imagen.

### 3.1.2 Submuestreo

Como se ha mencionado antes, la representación en el espacio escala se logra mediante la convolución con una función Gaussiana con un parámetro  $\sigma$  variable. En nuestro caso particular, usaremos una función Gaussiana con  $\sigma = \sqrt{2}$  pixel para generar el subconjunto de miembros de la representación en el espacio - escala de la imagen que utilizaremos para efectuar el análisis de correspondencias, las cuales llamamos también Pirámide Gaussiana. La convolución iterativa con la función Gaussiana produce el efecto de una sola convolución con valores de  $\sigma$  mayores.

Ahora bien, si notamos que el ancho de banda de una función Gaussiana con  $\sigma = \sqrt{2}$  pixel es de aproximadamente  $\pi$  radianes, entonces el teorema de Nyquist demuestra que es posible realizar un submuestreo de la imagen con un periodo de 2 pixeles en ambas direcciones espaciales, sobre la imagen convolucionada.

En cuanto al método utilizando submuestreo, cabe mencionar que se busca trabajar con una cantidad considerablemente menor de rasgos (imágenes más pequeñas) que describan de igual manera la misma escena pero con menos pixeles. Así al aplicarle el algoritmo a esta imagen submuestreada se encuentra la correspondencia en baja resolución, que al interpolarla se convertirá en una estimación de la correspondencia en un nivel de resolución mayor.

### 3.2 ALGORITMO DE CORRESPONDENCIA SIN SUBMUESTREO.

Las imágenes que se utilizaron durante el presente trabajo fueron de 160 x 120 pixeles tomadas desde cámaras paralelas con separación entre ellas de 3 cm. A continuación se muestran en niveles de grises y en rasgos:

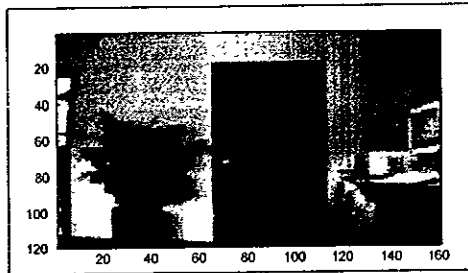


Imagen Izquierda

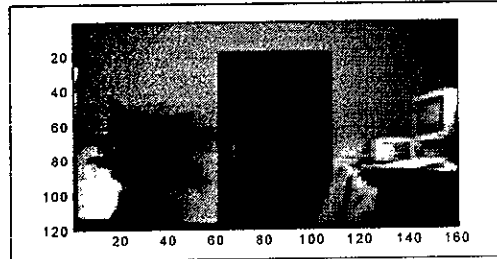


Imagen Derecha

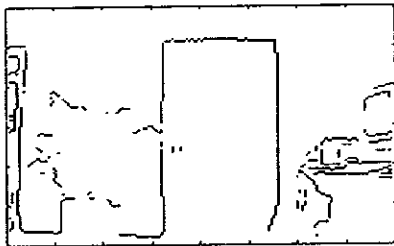


Imagen Izquierda

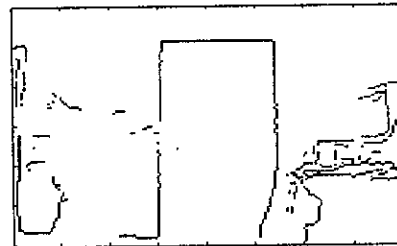


Imagen Derecha

Imágenes tomadas en el Laboratorio Nacional de Informática Avanzada (LANIA, A.C.), Jalapa, Ver.

**Figura 3.1**

#### 3.2.1 Esquema Piramidal del Algoritmo PMF a Múltiples Resoluciones.

Para poder tomar ventaja del concepto de pirámide Gaussiana en lo que es el proceso de correspondencia entre imágenes estéreo es necesario aplicar el algoritmo a cada una de las capas para así distribuir la densidad de puntos en cada capa y con esto lograr reducir el tiempo de procesamiento. A continuación se muestra el diagrama de flujo de este método:

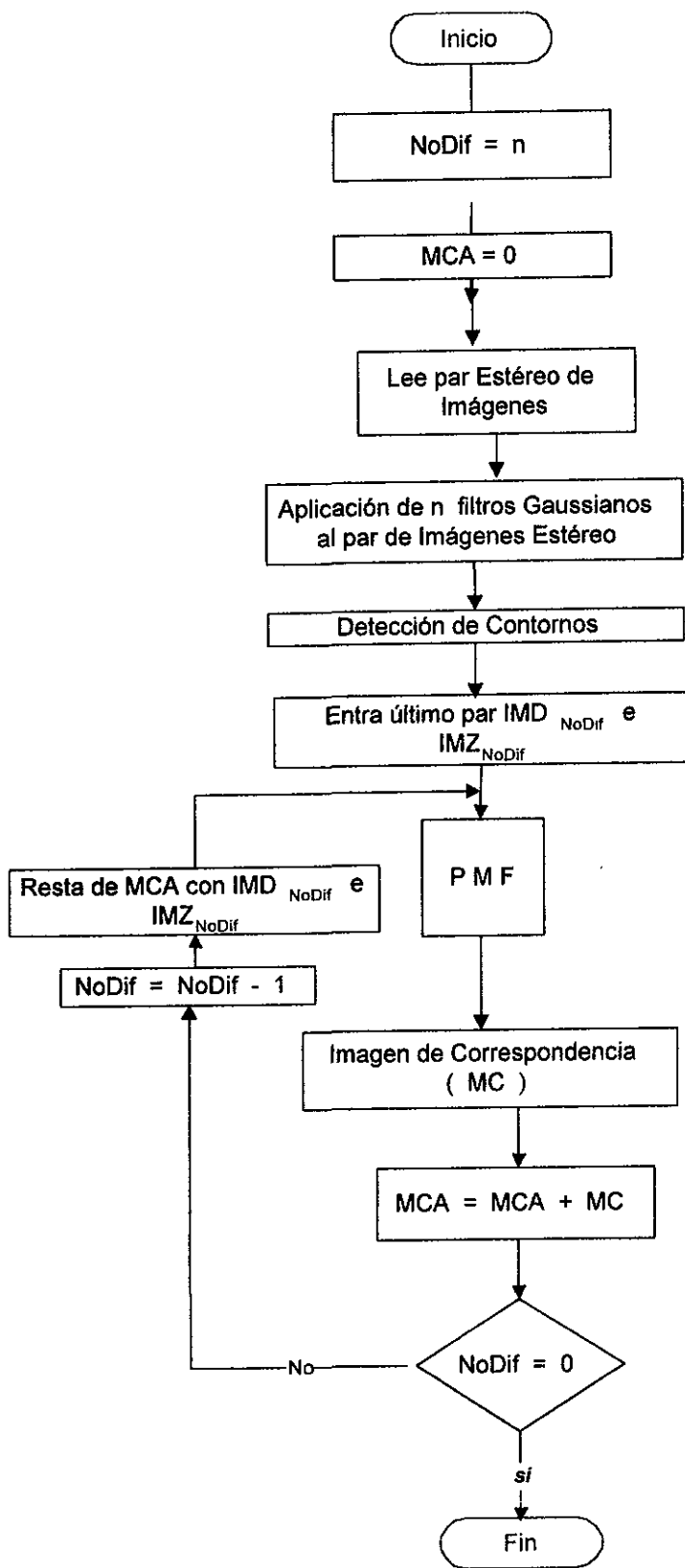
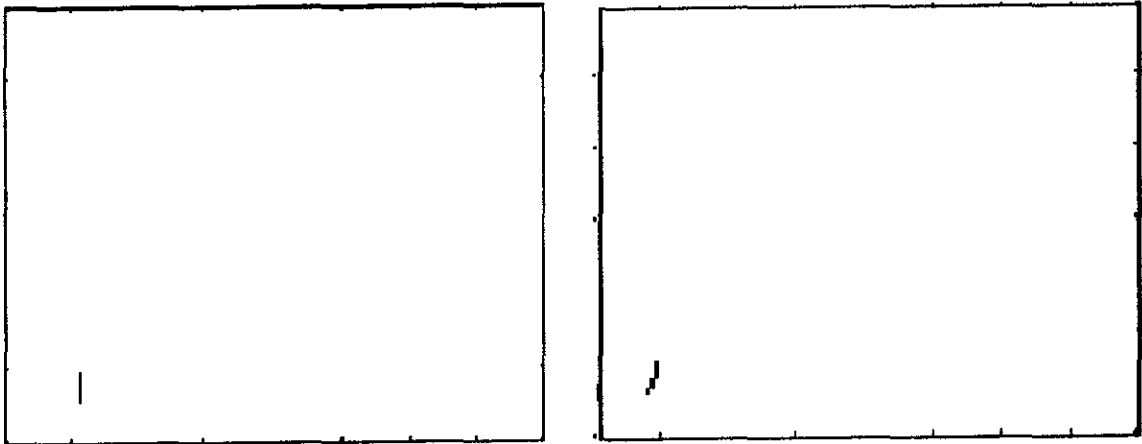


Figura 3.2

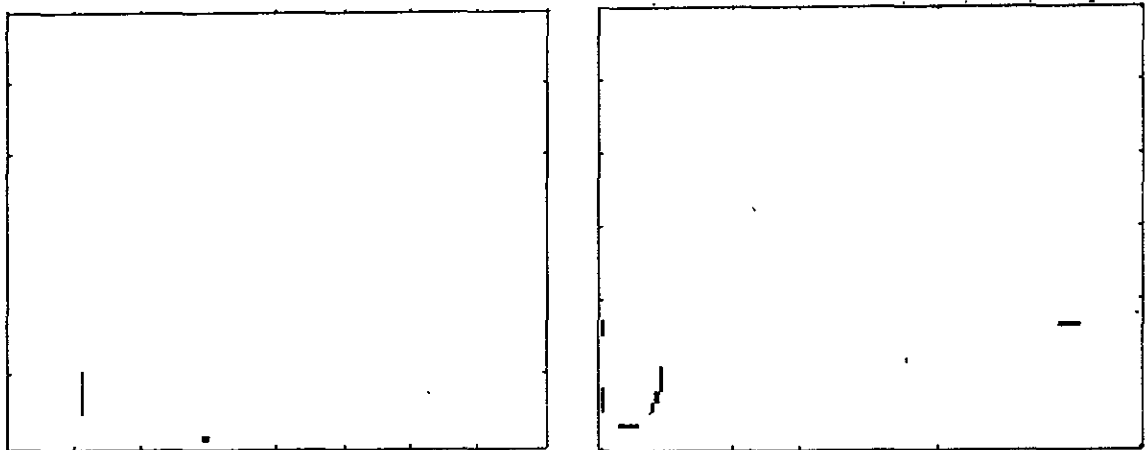
El algoritmo mostrado en la figura 3.2 se explicará a detalle en el punto 3.2.3 “Estrategia del cálculo de la correspondencia dentro de la estructura piramidal”.

### 3.2.2 Pirámide Gaussiana.

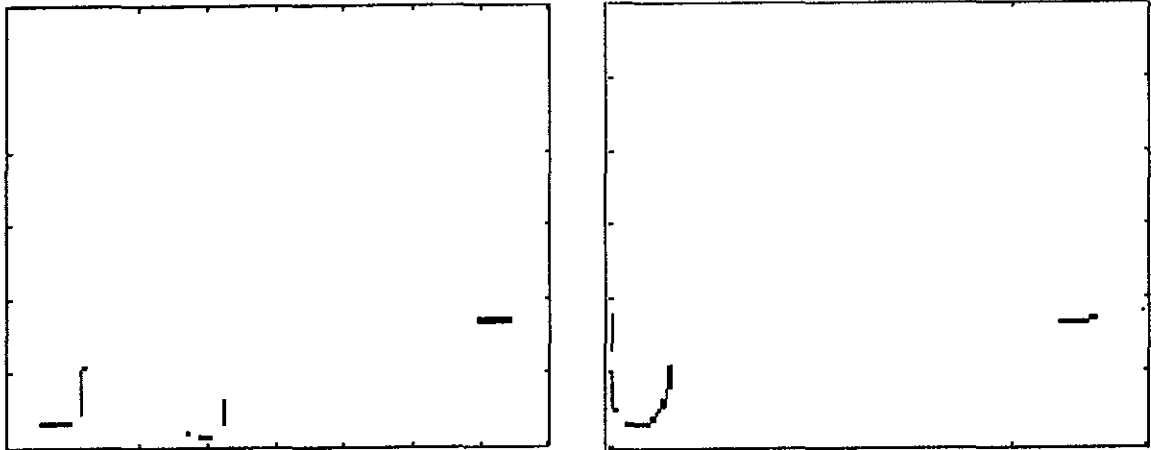
Aquí podemos ver las imágenes de orillas en cada nivel de la pirámide observando que de los niveles 7 al 9 se encuentran muy pocos rasgos:



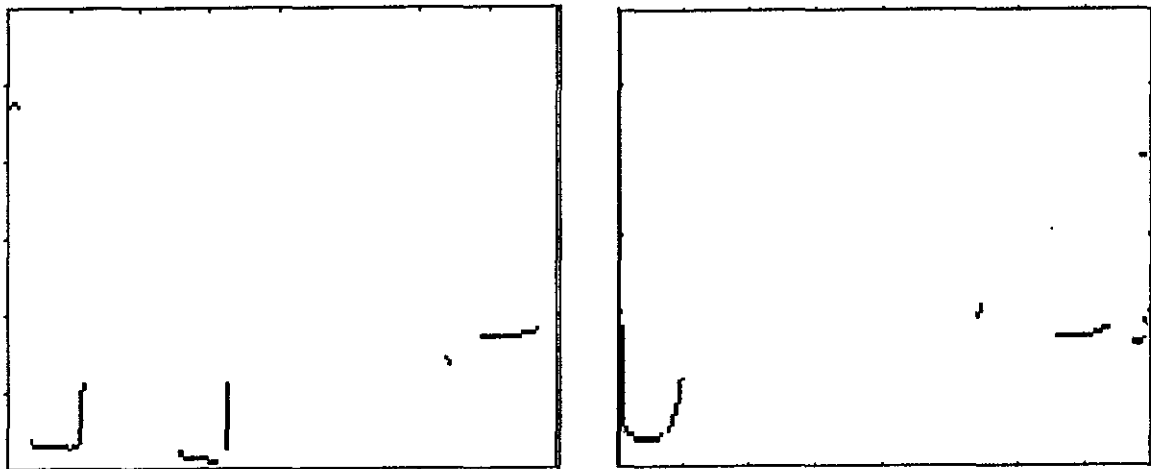
**Par Estéreo del 9° nivel de la Pirámide Gaussiana**



**Par Estéreo del 8° nivel de la Pirámide Gaussiana**

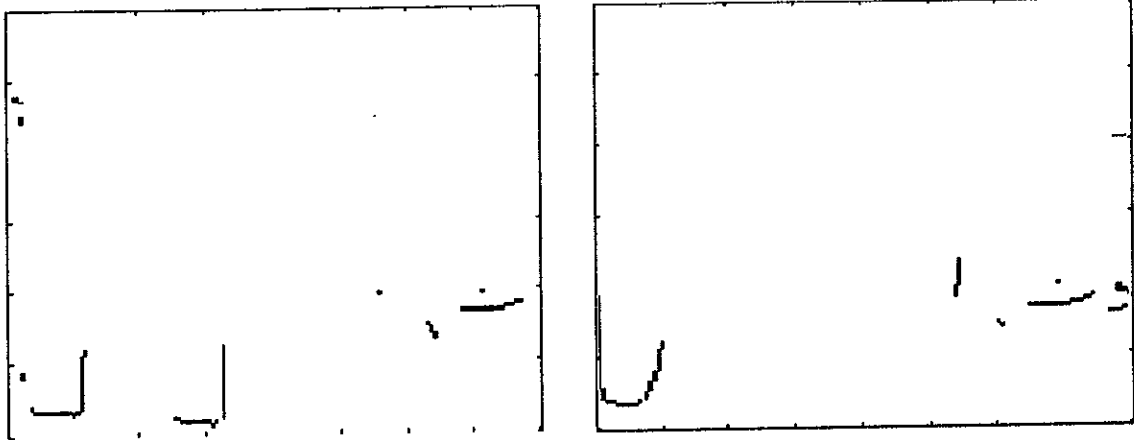


**Par Estéreo del 7° nivel de la Pirámide Gaussiana**



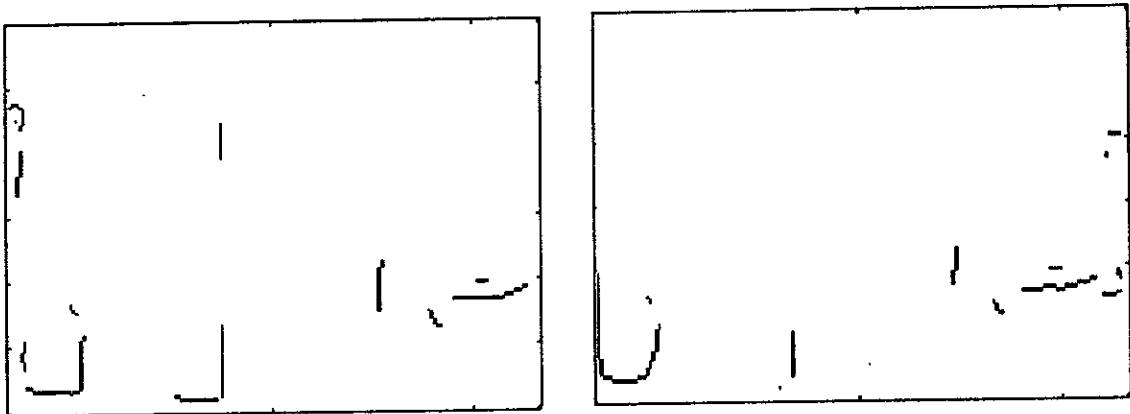
**Par Estéreo del 6° nivel de la Pirámide Gaussiana**

A partir de este nivel se empezó la aplicación del algoritmo PMF ya que como se puede apreciar en las imágenes del 9° al 7° nivel se tienen muy pocos rasgos por lo cual no es conveniente aplicarles a estos niveles el algoritmo debido a que no contribuyen en gran medida y se pierde tiempo llenando en contra de nuestro principal objetivo.



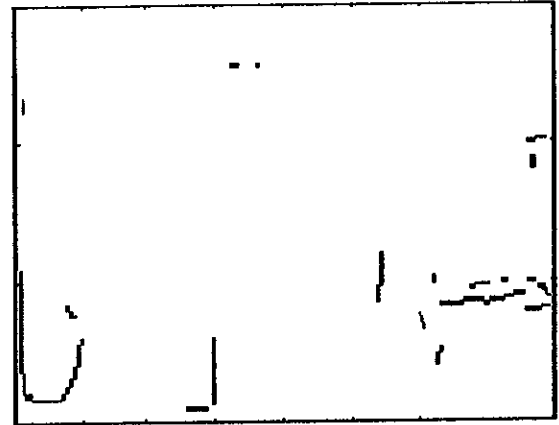
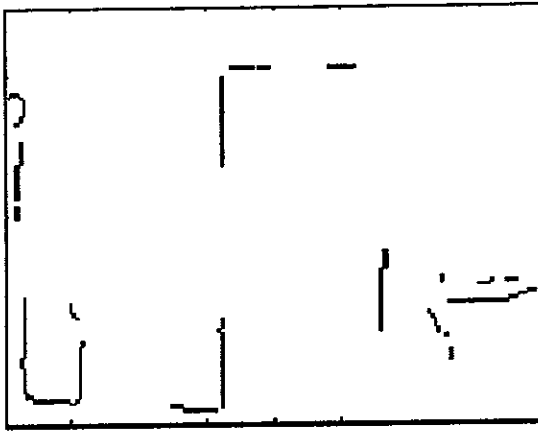
**Par Estéreo del 5º nivel de la Pirámide Gaussiana**

Con la aplicación del PMF a partir del nivel anterior muchos de los rasgos encontrados son restados a las imágenes de este nivel por lo que el tiempo de proceso va disminuyendo al ser repartido. El tiempo ganado es porque las operaciones del algoritmo (basado en vecindades y candidatos) disminuyen al tener menos rasgos en cada nivel.

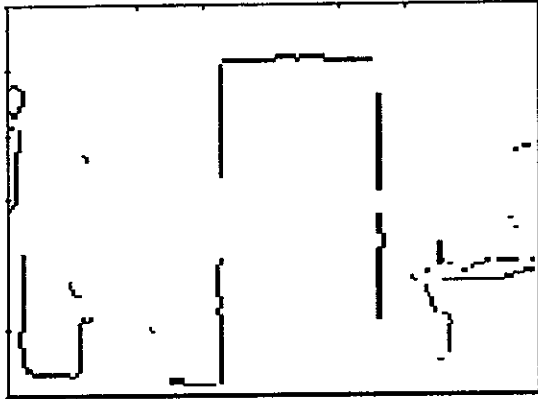


**Par Estéreo del 4º nivel de la Pirámide Gaussiana**

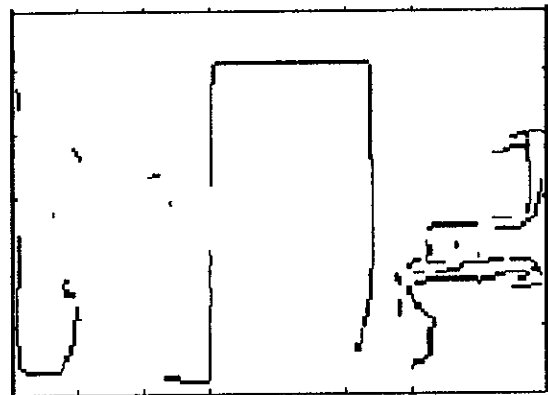
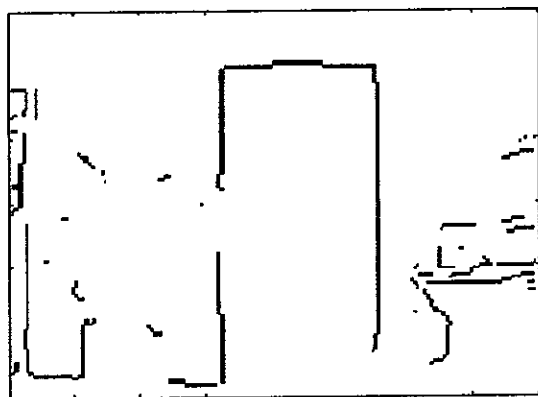




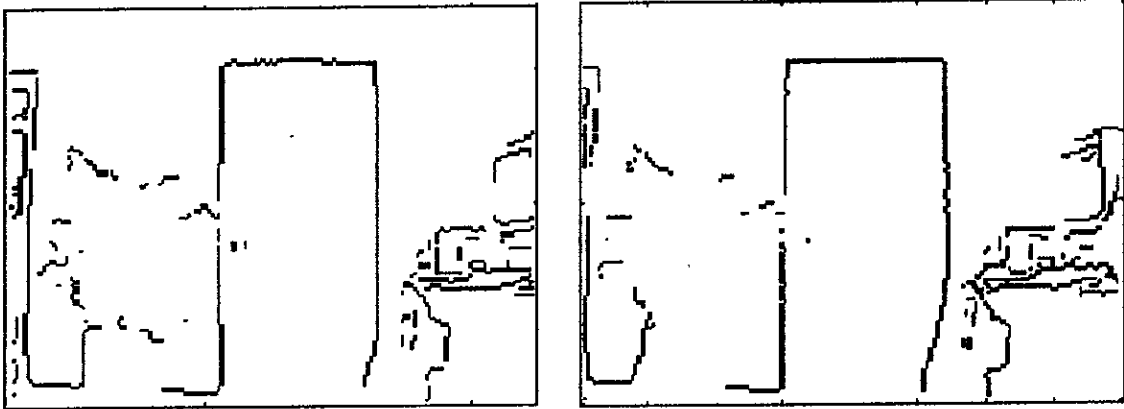
**Par Estéreo del 3<sup>er</sup> nivel de la Pirámide Gaussiana**



**Par Estéreo del 2<sup>o</sup> nivel de la Pirámide Gaussiana**



**Par Estéreo del 1<sup>er</sup> nivel de la Pirámide Gaussiana**



Par Original

### 3.2.3 Estrategia del cálculo de la correspondencia dentro de la estructura piramidal.

Para la aplicación del Algoritmo PMF con una estructura piramidal Gaussiana, es necesario definir un número  $n$  conveniente de imágenes a múltiples resoluciones con el objetivo de distribuir la carga de operaciones en cada nivel de la pirámide (ya que en cada nivel es necesario aplicar el PMF con mucho menos rasgos y posteriormente disminuirle trabajo al siguiente nivel) y con esto disminuir el tiempo de proceso total. En el presente trabajo se encontró que el número más conveniente es 6 .

Una vez que se eligió este número, se crea una imagen vacía con las dimensiones de las imágenes estéreo a procesar (Imagen de Correspondencia Acumulada , *MCA*) en la cual se irán acumulando los puntos que han sido correspondidos, hasta tener la imagen de correspondencia total después de haber pasado por todos los niveles.

El siguiente paso es leer las imágenes estéreo en niveles de grises, ya que de acuerdo a uno de nuestros objetivos planteados, el cual pretende recuperar únicamente la ubicación de objetos en profundidad, sería irrelevante el manejo de imágenes en color porque su procesamiento requeriría estructuras de datos más complejas y por lo cual más tiempo de proceso.

A partir del par de imágenes se obtendrán una serie de  $n$  pares de imágenes en donde cada par es el resultado de la aplicación de un filtro Gaussiano a su nivel anterior. A continuación se muestra el filtro utilizado, el cual es un filtro binomial de  $5 \times 5$  y aproxima a la función Gaussiana :

1/256\*

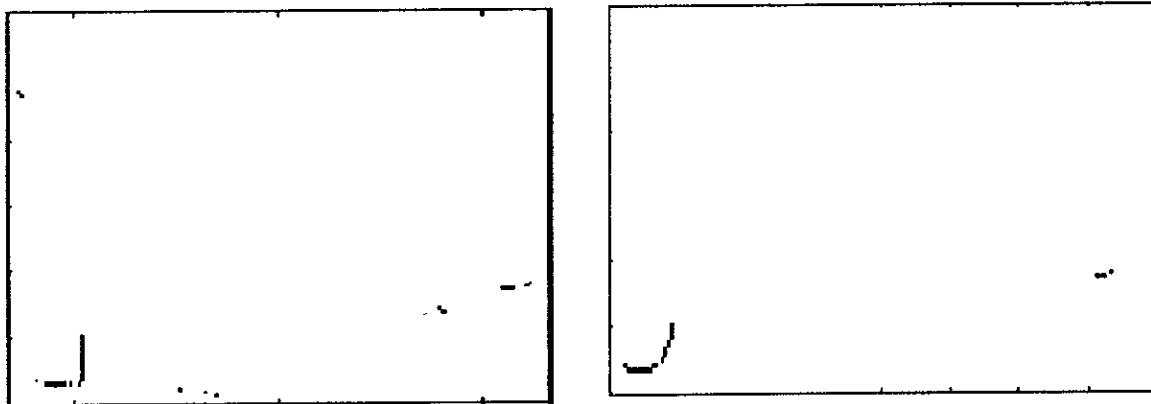
1	4	6	4	1
4	16	24	16	4
6	24	36	24	6
4	16	24	16	4
1	4	6	4	1

Una vez que se cuenta con todos los pares de imágenes involucrados se procede a obtener sus imágenes de contornos necesarios para la aplicación del algoritmo, mediante un filtro de Sobel.

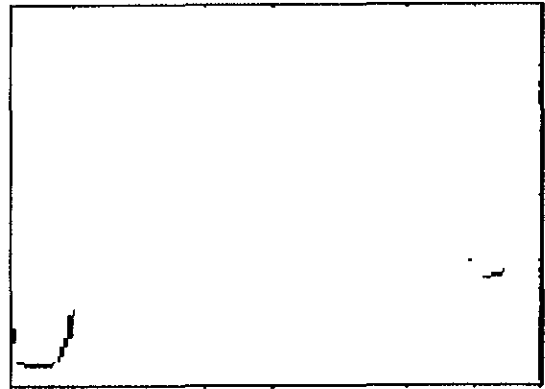
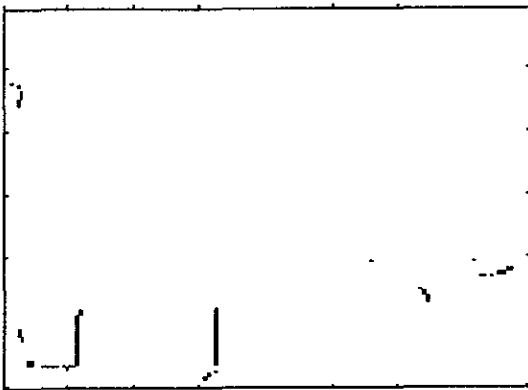
Al tener las imágenes de bordes se inicia un ciclo con el último par obtenido (que culminará con el par original) ,es decir , al que se le aplicaron  $n$  filtros Gaussianos y por lo tanto estas imágenes tienen una menor resolución (borrosas) y al aplicarle detección de contornos obtenemos un número menor de puntos que los pares que tienen mayor resolución y dicho par será procesado por el PMF con lo cual se inicia la distribución de los puntos a procesar.

Se obtiene entonces la imagen de Correspondencia la que se acumula en nuestra imagen *MCA*. En dicha imagen se conocen la ubicación de los puntos correspondidos, es decir las coordenadas de la imagen izquierda y derecha, con lo que es posible hacer una resta de los puntos ya correspondidos al par estéreo del siguiente nivel ( $IMD_{NoDif}$  e  $IMZ_{NoDif}$ ), y será la resta la que se procese y de la misma manera con las imágenes de los siguientes niveles hasta que el último par a procesar sea el original.

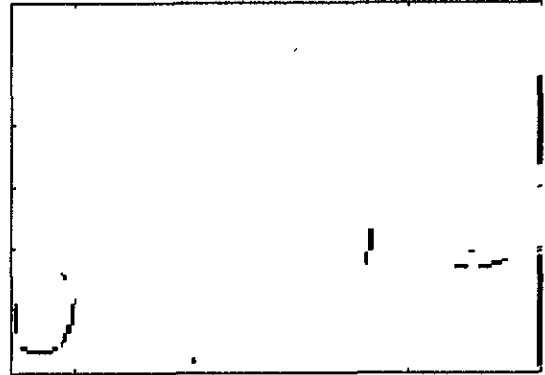
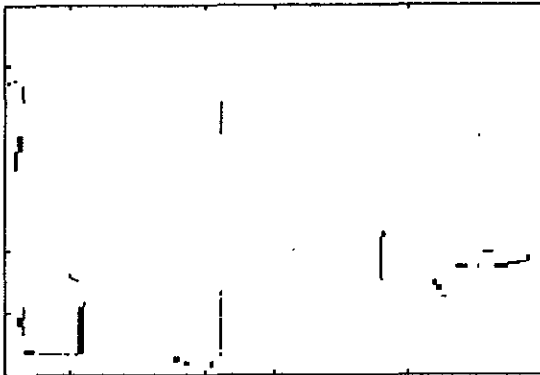
A continuación se muestran los resultados después la aplicación del PMF con los pares de imágenes obtenidas en cada nivel de la Pirámide Gaussiana, en la cual se pueden ver claramente las restas de los puntos que ya fueron correspondidos en niveles anteriores y con esto como se distribuyó el proceso total:



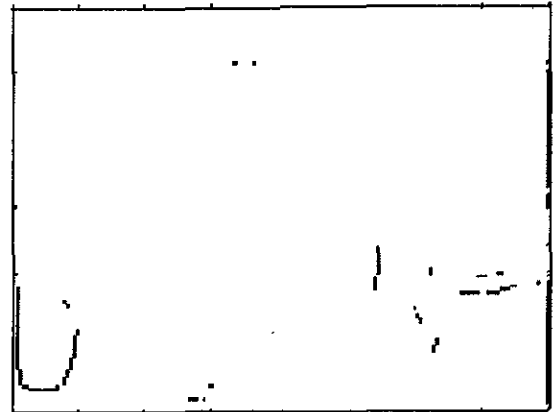
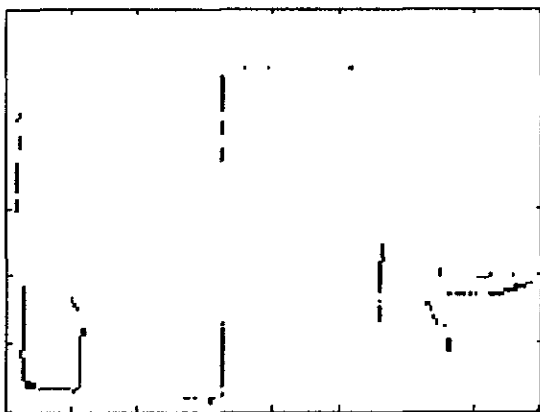
Resta de la imagen MCA con el par del nivel 5



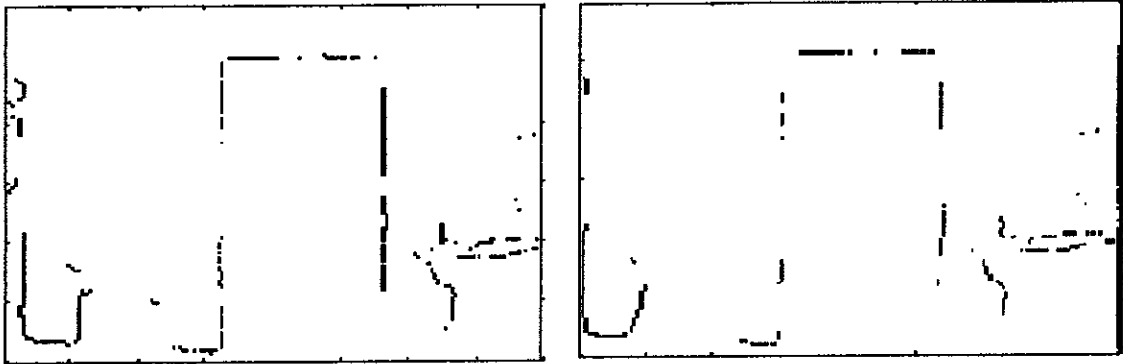
**Resta de la imagen MCA con el par del nivel 4**



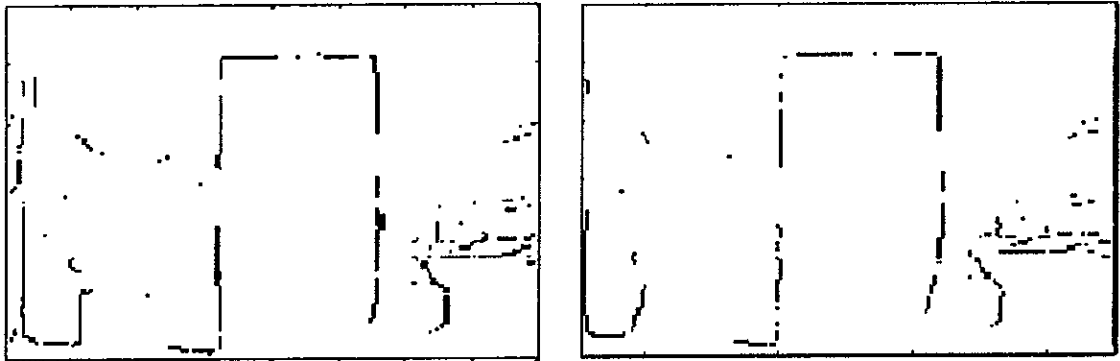
**Resta de la imagen MCA con el par del nivel 3**



**Resta de la imagen MCA con el par del nivel 2**



**Resta de la imagen MCA con el par del nivel 1**



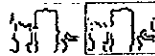
**Resta de la imagen MCA con el par original**

Después de aplicar el PMF podemos observar que el trabajo que se realizará en la resta del 1<sup>er</sup> par con la imagen original (figura anterior) será menor el tiempo que procesará este par de imágenes que si se procesara el par original por el número de puntos a corresponder.

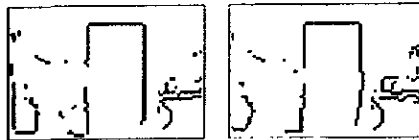
### 3.3 ALGORITMO DE CORRESPONDENCIA CON SUBMUESTREO.

#### 3.3.1 Esquema piramidal del algoritmo PMF con submuestreo.

A continuación se muestran las imágenes obtenidas al aplicar dos niveles de árbol cuádruple al par estéreo original. Es claro que encontrar la correspondencia a partir de imágenes más pequeñas es mucho más rápido ya que el número de puntos posibles también reduce considerablemente, sin embargo la exactitud reduce ya que las disparidades son muy similares y pueden causar errores en la estimación de la distancia. Este método puede ser útil dependiendo la aplicación, es decir, si requiere o no detalle de las características o si solo es para navegación que no requiera mucha exactitud.



Par Estéreo de 40 x 30 píxeles ( 2 submuestreos )



Par Estéreo de 80 x 60 píxeles ( 1 submuestreo )



Par Original

Figura 3.3

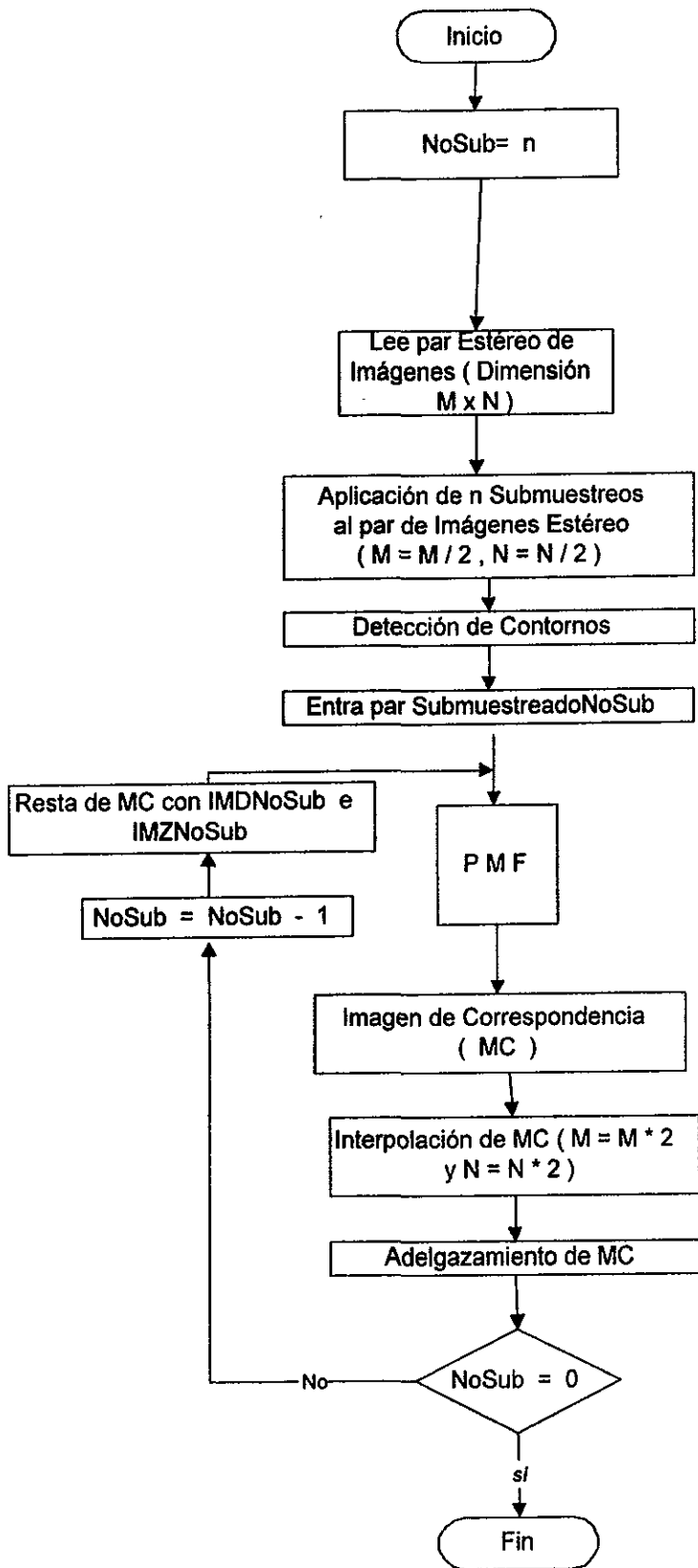


Figura 3.4

### 3.3.2 Estrategia del cálculo de la correspondencia aplicando el submuestreo.

Para aplicar este método se necesita definir un número de submuestras que se aplicará al par estéreo original, el cual será un número pequeño ya que se va disminuyendo la imagen a la mitad y por lo regular no se usan imágenes de más de 320 x 240 pixeles por el objetivo de disminuir los tiempos de proceso.

Ya definido el número de submuestras o aplicaciones del árbol cuádruple, se procede a leer el par de imágenes cuya dimensión original es de  $M \times N$  pixeles, con las características mencionadas anteriormente. Una vez aplicado el submuestreo, el primer par obtenido tendrá una resolución de  $M/2 \times N/2$  pixeles y el siguiente de  $M/4 \times N/4$  y así hasta el número  $n$  de niveles deseados. A los pares obtenidos se les aplica la detección de contornos y se procede a aplicar el PMF al par de imágenes de menor dimensión obteniendo así una imagen de correspondencia  $MC$  en ese tamaño.

Esta imagen  $MC$  se interpola para obtenerla con un tamaño del doble y posteriormente se debe adelgazar, ya que los métodos usados para interpolar hacen los rasgos doblemente gruesos horizontal y verticalmente, para así tener una imagen más aproximada a la del siguiente nivel. Hecho esto se resta  $MC$  ya interpolada al par de imágenes  $IMD_{NoSub-1}$  e  $IMZ_{NoSub-1}$ . Con esto se le restará una aproximación de los rasgos correspondidos a estas imágenes, es decir, si se encontraron correspondidos  $x$  rasgos, al realizar la interpolación y el adelgazamiento se convertirán en el doble.

### 3.3.3 El problema de interpolación.

Para aplicar el Submuestreo o concepto de árbol cuádruple al algoritmo de correspondencia, el tamaño del par de imágenes debe ir disminuyendo a la mitad en cada una de las capas del submuestreo, y al llegar al par de menor tamaño éste se procesa por el algoritmo y se obtiene su imagen de correspondencia, es aquí cuando resulta necesario realizar la interpolación en base a los resultados con el fin de proyectarlos al tamaño inmediato superior (al doble) y en este se restarán los puntos de las imágenes interpoladas con los puntos de las imágenes de dicho nivel, para que en dicho nivel no se procesen los puntos ya encontrados en el nivel anterior y acelerar el proceso de correspondencia, de esta manera se debe llegar al nivel original del par de imágenes.

Cuando se interpolan las imágenes de correspondencia, es necesario utilizar un algoritmo para realizar esta operación. En el presente trabajo se aplicó el algoritmo de interpolación bilineal el cual funciona de tal manera que todos los rasgos (que originalmente se intenta que sean de un pixel de ancho, esto con fines de optimar tiempos) quedan aumentados al doble de ancho por lo que resulta necesario realizar procesos de adelgazamiento por lo que al llegar al resultado final (con el tamaño de las imágenes originales) encontramos que los puntos pierden sus posiciones originales, lo



que significa que este método no es confiable. A continuación se muestran sobrepuestas dos imágenes de correspondencia de las cuales la de color más claro fue obtenida con ayuda de la interpolación y se nota claramente como se pierden las coordenadas originales (tono de gris más oscuro) después de realizar los procesos de interpolación y adelgazamiento horizontal y vertical, además que estos algoritmos adicionales que sería necesario aplicar hacen más lento el proceso de correspondencia, por lo que este método de esquema piramidal del algoritmo PMF se descartó y solo se aplicó el algoritmo a las imágenes submuestreadas en cada nivel de submuestreo de forma independiente, de los cuales se obtuvieron resultados muy buenos en cuanto a los tiempos de proceso.

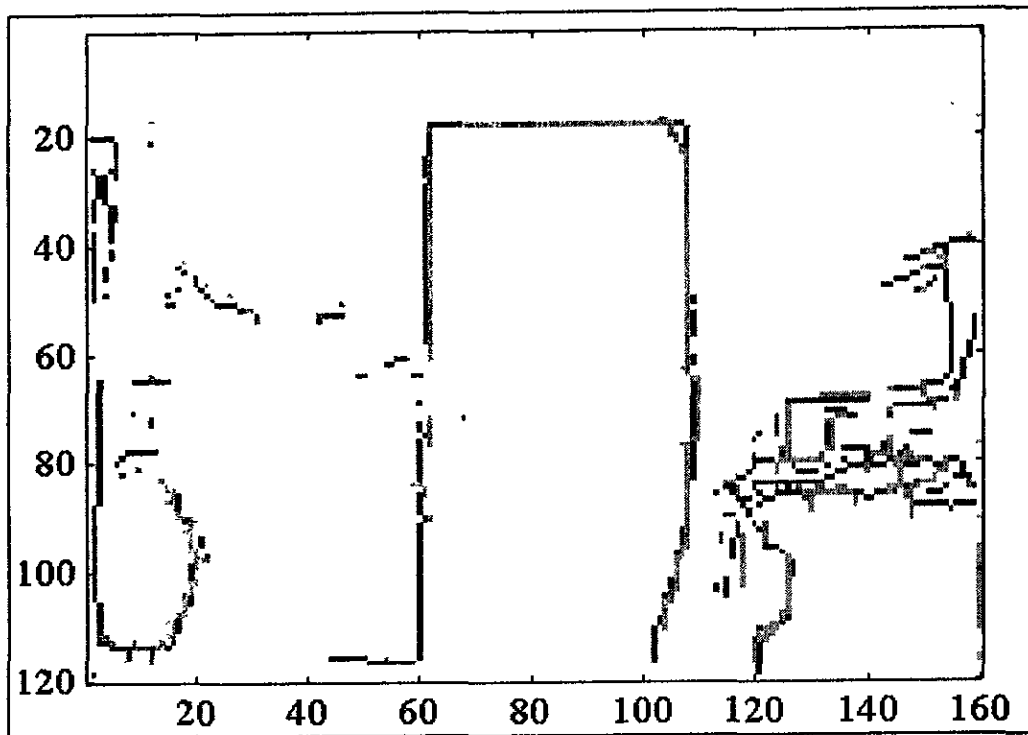


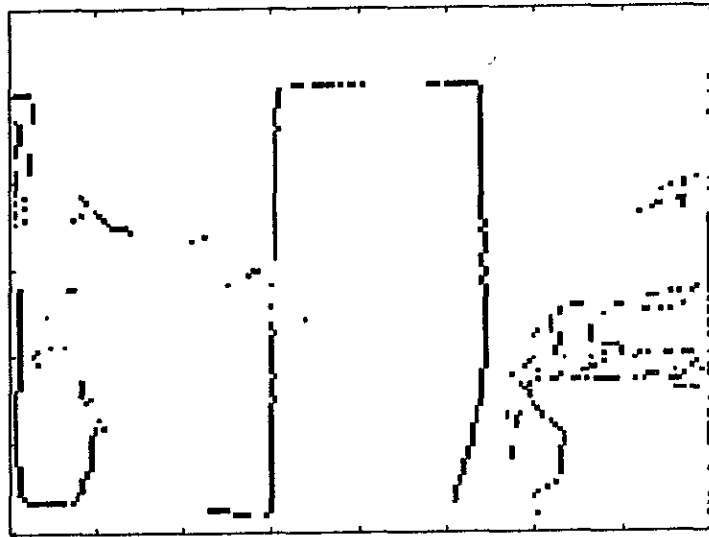
Figura 3.5

## CAPÍTULO 4: RESULTADOS Y EVALUACIÓN.

### 4.1 RESULTADOS DEL ALGORITMO PMF.

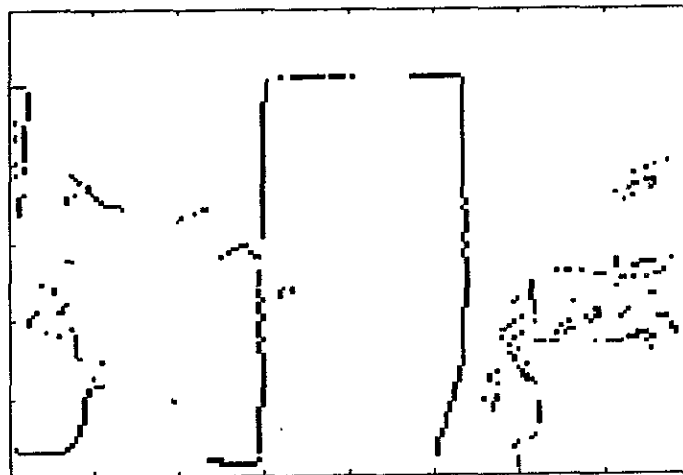
Para mostrar los resultados que se obtuvieron aplicando los diferentes métodos que utilizan el algoritmo PMF se usaron las imágenes ya mostradas en el capítulo 3 y partiendo de ellas tenemos los siguientes resultados.

#### 4.1.1 Sin aplicar múltiples resoluciones.



*IMAGEN DE CORRESPONDENCIA CON EL ALGORITMO PMF*

#### 4.1.2 Usando la pirámide Gaussiana.



*IMAGEN DE CORRESPONDENCIA CON EL ALGORITMO PMF USANDO MÚLTIPLES RESOLUCIONES*

Para los dos casos anteriores se usaron vecindades de 14 píxeles, ya que este algoritmo utiliza el concepto de vecindades las cuales ayudan al cálculo de los pesos ponderados, así como una lista de rasgos candidatos de 15 píxeles ya que la disparidad real más grande vista comparando ambas imágenes fue de 15 píxeles en imágenes de 160 x 120 píxeles y con separación entre cámaras de 3 cm, y así con estos parámetros de entrada se puede obtener con certeza el punto correspondiente.

### 4.1.3 Usando submuestreo.

a) Aplicación del PMF a imágenes con dos niveles de submuestreo (40 x 30) a partir de las imágenes originales (160 x 120)

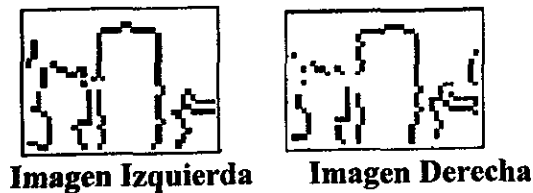


Imagen de Correspondencia

b) Aplicación del PMF a imágenes con 1 nivel de submuestreo (80 x 60) a partir de las imágenes originales (160 x 120)

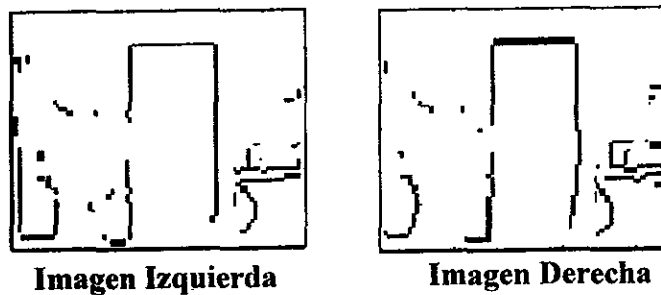


Imagen de Correspondencia

Debido al problema provocado porque en este caso no se manejan imágenes de grises en la interpolación no resulta muy conveniente el usar este método junto con el PMF por lo que solo se aplica este algoritmo a las imágenes submuestreadas y se encuentra la imagen de correspondencia en este tamaño, entonces resulta posible encontrar la matriz de disparidades y por lo tanto la aproximación a la distancia real de los objetos con respecto a la cámara.

## 4.2 EVALUACIÓN.

### 4.2.1 Tiempo de Cómputo.

Método	Tamaño Imagen	Tamaño de Vecindad	Rasgos Candidatos	Tiempo (seg)
PMF	160 x 120 píxeles	14 x 14 píxeles	15 píxeles	136
PMF con Pirámide Gaussiana	160 x 120 píxeles	14 x 14 píxeles	15 píxeles	114
PMF con 1 nivel de Submuestreo	80 x 60 píxeles	8 x 8 píxeles	10 píxeles	18
PMF con 2 niveles de Submuestreo	40 x 30 píxeles	4 x 4 píxeles	5 píxeles	2.5

### 4.2.2 Calidad (Porcentaje de puntos correspondidos).

Método	Tamaño Imagen	Rasgos ID	Rasgos II	MNRC	RC	Eficiencia ( $\eta_c$ ) %
PMF	160 x 120 píxeles	746	763	687	505	73.50
PMF con Pirámide Gaussiana	160 x 120 píxeles	746	763	687	628	91.41
PMF con 1 nivel de Submuestreo	80 x 60 píxeles	325	362	318	258	81.13
PMF con 2 niveles de Submuestreo	40 x 30 píxeles	131	140	125	106	84.80

**Donde :**

**MNRC** Es el Máximo Número de Rasgos a Corresponder

**Rasgos ID** Es el Número de Rasgos en la Imagen Derecha

**Rasgos II** Es el Número de Rasgos en la Imagen Izquierda

$\eta_c$  Es la eficiencia del algoritmo con respecto a los rasgos correspondidos y está dada por la siguiente expresión:

$$\eta = \frac{RC}{MNRC}$$

**RC** Son el número de Rasgos Correspondidos

### 4.2.3 Calibración de la Cámara

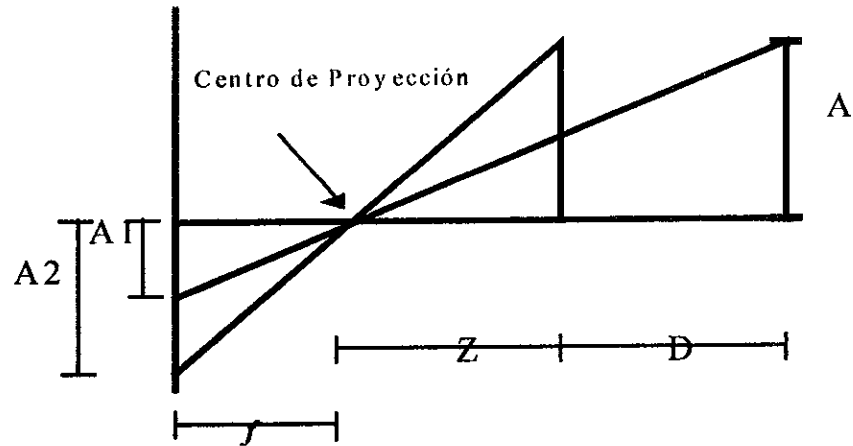
El proceso de calibración de cámara es la técnica mediante la cual se logra relacionar las imágenes con el mundo real, ya que las dimensiones en la representación digital están en función de la relación existente entre las unidades de longitud focal y su equivalencia en pixeles, es decir, resulta necesario convertir las unidades de  $f$  a pixeles ya que de esta forma se conservará la relación entre las medidas reales de los objetos en la imagen y su representación digital.

Cabe mencionar que mediante el proceso de calibración es posible determinar la posición y orientación de la cámara en el espacio así como de detectar y corregir deformaciones de digitalización producidos por imperfecciones en la lente.

Existen varios procedimientos de calibración de cámara cuyas metodologías varían dependiendo de las características particulares de la aplicación a la que estén integradas. A continuación se detalla el procedimiento que se realizó en nuestro caso.

Sabemos que una vez obtenido el mapa de disparidades lo que resta es obtener es obtener la longitud focal de la cámara, la cual se obtiene a partir de triángulos semejantes que se forman de un objeto tomado a dos diferentes distancias.

Dado un objeto de altura  $A$  se obtendrán distintas proyecciones en función de la distancia  $D$  de la cámara al objeto, dichas proyecciones se pueden describir con la siguiente relación:



Por triángulos semejantes :

$$\frac{f}{A1} = \frac{Z+D}{A} \quad , \quad \frac{f}{A2} = \frac{Z}{A}$$

$$Z = \frac{f * A}{A1} - D \quad \dots\dots (1)$$

$$Z = \frac{f * A}{A2} \quad \dots\dots (2)$$

De (1) y (2)

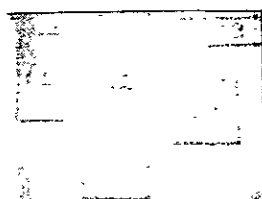
$$D = \frac{f * A}{A1} - \frac{f * A}{A2} \quad , \quad D = f \left( \frac{A}{A1} - \frac{A}{A2} \right)$$

Despejando a  $f$ , obtenemos la longitud focal:

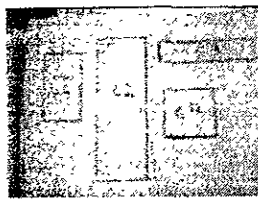
$$f = \frac{D}{A \left( \frac{1}{A1} - \frac{1}{A2} \right)}$$

Considerando que las unidades de  $D$  y  $A$  están en cm y las de  $A_1$ ,  $A_2$  y  $A_3$  en pixeles. se obtiene el valor de  $f$  expresado en pixeles.

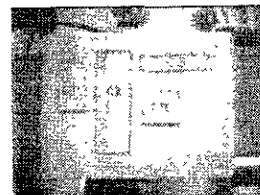
El procedimiento de calibración consistió en adquirir tres imágenes que contenían figuras de alturas conocidas ( $A$ ) y tomándolas en pares todas las combinaciones de las tres imágenes se adquirieron diferentes valores de la longitud focal, las cuales variaron según los rangos de distancia tomados.



A1 (52 cm)



A2 (68 cm)



A3 ( 100 cm )

En las siguientes tablas se expresan los resultados obtenidos para la longitud focal:

a) Cálculo entre las imágenes de  $A_i=A_2$  y  $A_j=A_1$  y  $D=16$

Rectángulo	Altura [ cm ]	$A_i$ [ pixeles ]	$A_j$ [ pixeles ]	$A(1/A_1-1/A_2)$
C1	15	44	56	0.0730515
C2	30	90	117	0.0769230
C3	5	14	18	0.0744040
C4	10	32	42	0.0744050

De la tabla anterior se obtuvo un promedio del denominador que resultó ser de **0.0746958** obteniendo una longitud focal de **214.201922**.

b) Cálculo entre las imágenes de  $A_i=A_3$  y  $A_j=A_1$  y  $D=48$

Rectángulo	Altura [ cm ]	$A_i$ [ pixeles ]	$A_j$ [ pixeles ]	$A(1/A_1-1/A_2)$
C1	15	32	56	0.2008935
C2	30	64	117	0.2123400
C3	5	10	18	0.2222200
C4	10	22	42	0.2219870

De la tabla anterior se obtuvo un promedio del denominador que resultó ser de **0.2143601** obteniendo una longitud focal de **223.92226**.

c) Cálculo entre las imágenes de  $A_i=A_3$  y  $A_j=A_2$  y  $D=32$



Rectángulo	Altura [ cm ]	Ai [ pixeles ]	Aj [ pixeles ]	$A(1/A1-1/A2)$
C1	15	32	44	0.1278420
C2	30	64	90	0.1354170
C3	5	10	14	0.1428575
C4	10	22	32	0.1420450

De la tabla anterior se obtuvo un promedio del denominador que resultó ser de **0.13670403** obteniendo una longitud focal de **233.50795**.

Ya con estas tres distancias focales se obtiene un promedio final, la cual va a ser nuestra distancia focal utilizada para los cálculos de la distancia real.

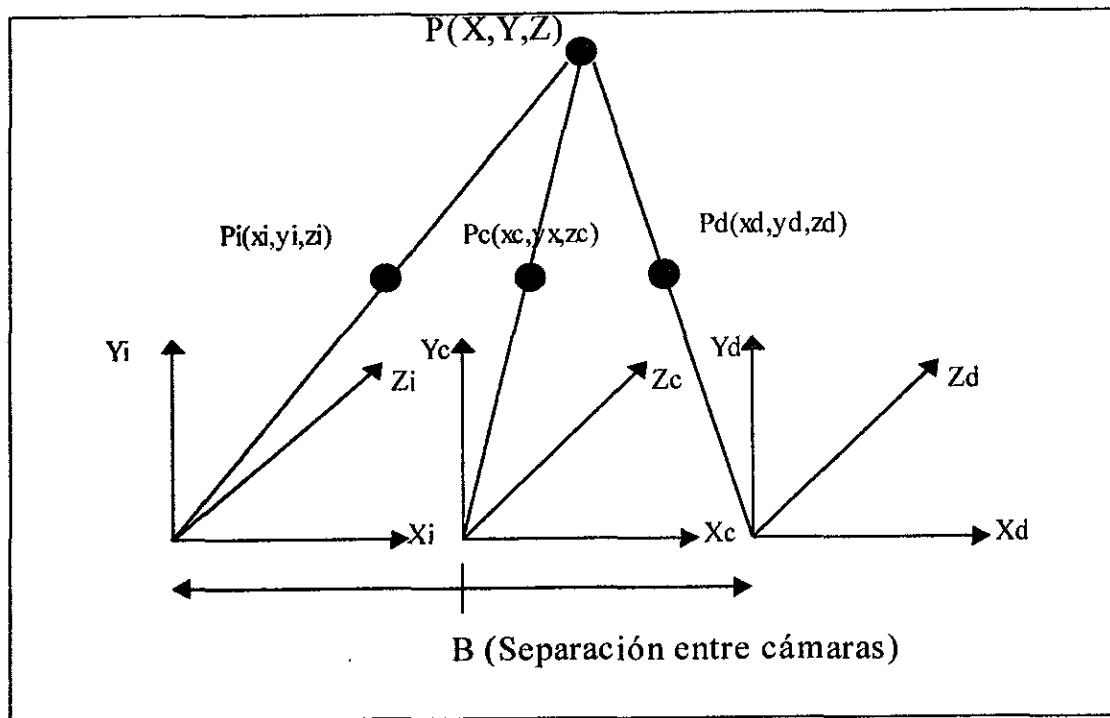
Con los resultados anteriores nos podemos dar cuenta que la longitud focal obtenida entre las muestras varía por lo cual es conveniente obtener un valor promedio de esta, para las muestras tomadas resultó de 223.8774 pixeles la cual se debe cerrar a 224 pixeles para mayor comodidad en imágenes de 160 x 120 pixeles.

Debido a que el algoritmo se desarrolló con diferentes niveles de resolución, es necesario procesar imágenes de 80 x 60 y de 40 x 30 pixeles. Para dichos procesos se requiere un valor de longitud focal según el tamaño de la imagen y debido a que el tamaño de los objetos es proporcional entre las imágenes se puede dividir o multiplicar en la misma proporción dependiendo de las dimensiones de las imágenes procesadas, por tanto para las imágenes de 80 x 60 se obtuvo un valor de  $f$  de 112 y para las imágenes de 40 x 30 un valor de 61.

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

### 4.2.4 Estimación de la Distancia

Partiendo de la existencia de dos sistemas coordenados paralelos (uno por cada cámara) se establece un tercero el cual está simétricamente situado entre los dos anteriores, llamado sistema Ciclopeano.



Geometría Estéreo para Sistemas Coordenados Paralelos

Apoyados en los sistemas coordenados anteriores se pueden establecer las siguientes relaciones:

$$x_c = x_d + \frac{B}{2}$$

$$y_c = y_d$$

$$z_c = z_d$$

$$x_c = x_i - \frac{B}{2}$$

$$y_c = y_i$$

$$z_c = z_i$$

Por cada punto detectado en el sistema Ciclopeano, se tiene la siguiente relación:

$$x_i = \frac{f}{Z} \left( X - \frac{B}{2} \right)$$

$$y_i = \frac{Y f}{Z}$$

$$x_d = \frac{f}{Z} \left( X + \frac{B}{2} \right)$$

$$y_d = y_i$$

Considerando que se conoce la disparidad  $x_d - x_i$  entre puntos se puede recuperar la ubicación real  $(X, Y, Z)$  del punto a partir de las siguientes fórmulas:

$$X = B \frac{x_d + x_i}{Z(x_d - x_i)}$$

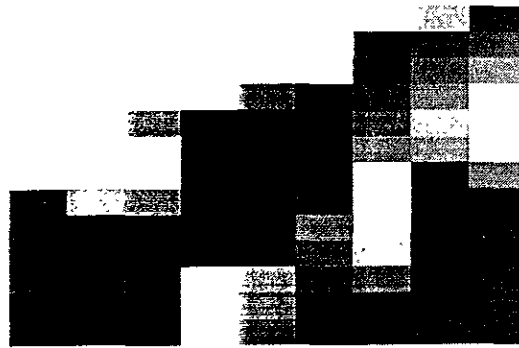
$$Y = B \frac{y}{x_d - x_i}$$

$$Z = B \frac{f}{x_d - x_i}$$

Donde  $B$  es la distancia entre cámaras y  $f$  es la distancia focal.

Para poder ver de manera más clara los resultados obtenidos en el cálculo de la distancia, tomaremos como referencia las siguientes imágenes sintéticas en las que se puede ver la profundidad de los objetos que forman la escena.





Segmento correspondiente a una de las partes más lejanas de la imagen

A continuación se muestran los resultados de las disparidades en pixeles:

										2	
										2	
									2		
											2
3											
					3						
				3							
		3									

Se nota claramente que los valores de disparidad en esta segmento son los menores encontrados, ya que se trata de una región lejana a la cámara.

#### 4.2.5 Exactitud en el Cálculo de la Distancia

Para conocer la exactitud de los diferentes métodos en cuanto al cálculo de la distancia, se obtuvieron 3 pares de imágenes que componen una secuencia, es decir, cada par está a una distancia distinta. Cabe mencionar que nos enfocamos a dos objetos (uno siendo un soldado que es el objeto más cercano a la cámara y el otro un marinero considerablemente más alejado).

Primero mostraremos los resultados para el par de imágenes más cercano en la que el objeto más cercano esta a 26 cm y el otro objeto a considerar estaba a 98 cm.



Imagen izquierda de 160 x 120 pixeles

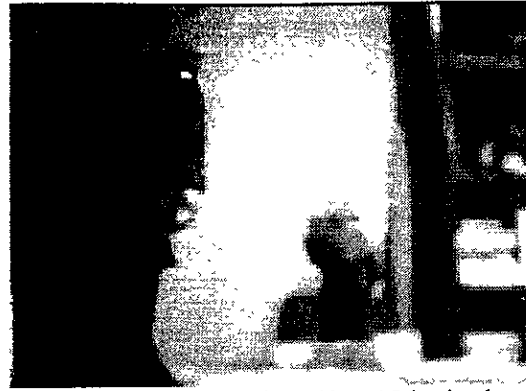


Imagen derecha de 160 x 120 pixeles



Imagen izquierda de 80 x 60 pixeles

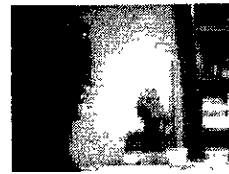


Imagen derecha de 80 x 60 pixeles



Imagen izquierda de 40 x 30 pixeles



Imagen derecha de 40 x 30 pixeles

Las imágenes de correspondencia para cada método son las siguientes:

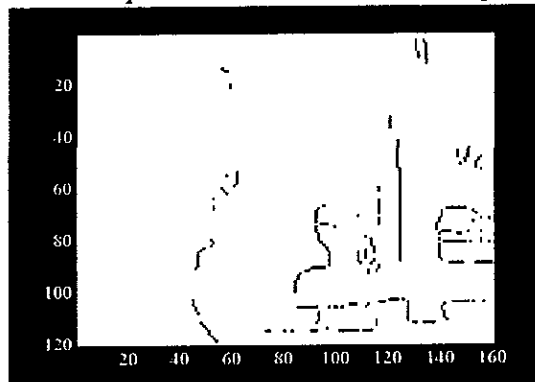


Imagen de Correspondencia de 160 x 120

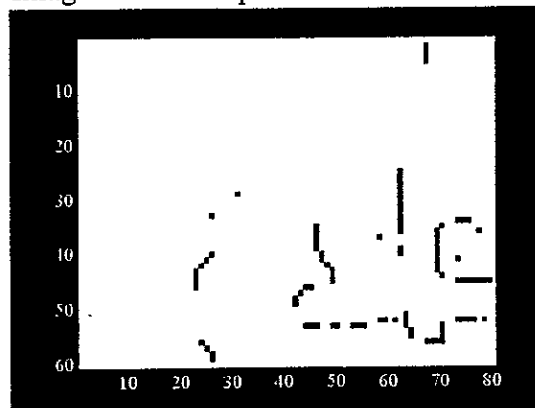


Imagen de Correspondencia de 80 x 60

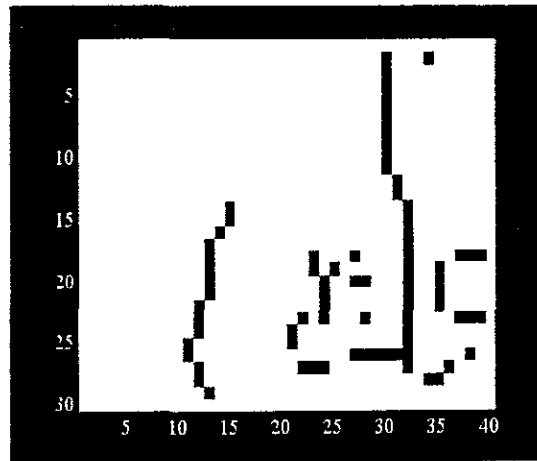


Imagen de Correspondencia de 40 x 30

Método	Distancia obtenida más cercana [cm]	Distancia real más cercana [cm]	Error [cm]	Error[%]
PMF	25.8462	26	0.1538	0.59
PMF con Pirámide Gaussiana	25.8462	26	0.1538	0.59
PMF con 1 nivel de Submuestreo	25.8462	26	0.1538	0.59
PMF con 2 niveles de Submuestreo	24	26	2	7.69

El siguiente par de imágenes que componen la secuencia se muestran a continuación, y el objeto más cercano está a 38 cm de la cámara.



Imagen izquierda de 160 x 120 píxeles



Imagen derecha de 160 x 120 píxeles



Imagen izquierda de 80 x 60 pixeles



Imagen derecha de 80 x 60 pixeles



Imagen izquierda de 40 x 30 pixeles



Imagen derecha de 40 x 30 pixeles

A continuación se muestran las imágenes de correspondencia para 3 tamaños de imágenes utilizados en los métodos.

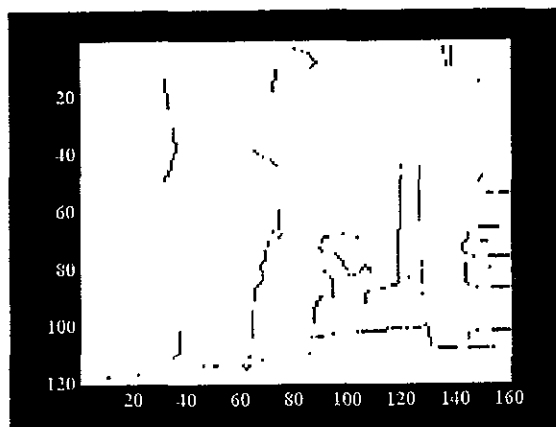


Imagen de correspondencia para 160 x 120 pixeles (objeto a 38 cm)

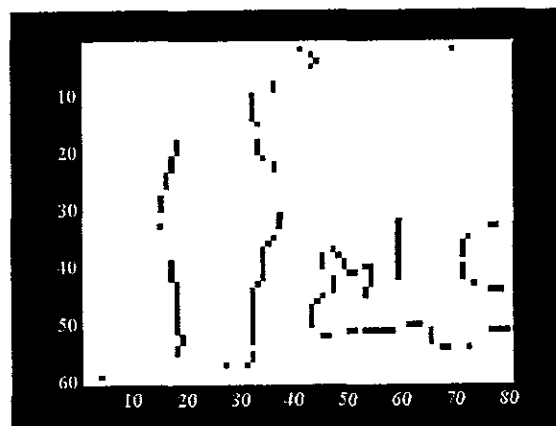


Imagen de correspondencia para 80 x 60 pixeles (objeto a 38 cm)



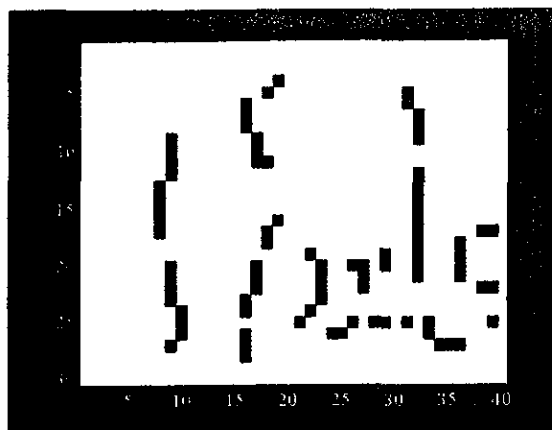


Imagen de correspondencia para 40 x 30 pixeles (objeto a 38 cm)

El cuadro de resultados para los diferentes métodos es :

Método	Distancia obtenida más cercana [cm]	Distancia real más cercana [cm]	Error [cm]	Error[%]
PMF	37.333	38	0.67	1.76
PMF con Pirámide Gaussiana	37.333	38	0.67	1.76
PMF con 1 nivel de Submuestreo	37.333	38	0.67	1.76
PMF con 2 niveles de Submuestreo	33.600	38	4.4	11.57

## Conclusiones.

El tema que abarca el presente trabajo de tesis es el conocido como Sistemas Artificiales de Percepción Visual, dichos sistemas son complejos y no pueden considerarse aún totalmente ideales. La Visión Artificial, como todas las áreas de estudio en la Inteligencia Artificial, tiene como principal objetivo implantar los sentidos y capacidades de los seres biológicos (principalmente del hombre) por medio de métodos y algoritmos ejecutados con la ayuda de la computadora digital, todo esto con el propósito de facilitar cada vez más (a medida que avanza la tecnología) las actividades humanas, para lograrlo es necesario basarse en los sistemas biológicos que nos proporcionan información fundamental para poder desarrollar sistemas con una aplicación específica.

Hoy en día no es posible crear un sistema de Visión Artificial que sea de propósito general y que sea ejecutado en tiempos cercanos al real, por lo que la mayoría de los sistemas que existen están enfocados solo a una tarea concreta. El presente trabajo se enfocó a probar varios métodos utilizando el algoritmo PMF con el fin de saber cual es el más adecuado para obtener una aproximación de las distancias que existen entre los objetos de la escena tomando como origen la cámara que obtuvo las imágenes, es decir, estos métodos no serían útiles si lo que se requiere es obtener la textura, color, iluminación, etc.

Dada la importancia de desarrollar sistemas de visión artificial y la necesidad de optimizar los tiempos de proceso de los algoritmos involucrados, obtuvimos los mapas de disparidades por medio de tres métodos basados en el algoritmo PMF, el primero fue aplicando sólo el algoritmo original, en los otros dos métodos complementamos el algoritmo con procesos de múltiples resoluciones (filtros Gaussianos y Submuestreo o árbol cuádruple). Con los dos últimos métodos obtuvimos ganancia en el tiempo de proceso. La ganancia más significativa es aplicando submuestreo, sin embargo, se sacrifica la exactitud, ya que al aplicar el algoritmo a imágenes de poca resolución se obtienen disparidades muy parecidas en toda la escena y por lo tanto las distancias que se obtienen no son muy confiables para sistemas que requieran gran exactitud en el cálculo de distancias, se recomienda el método que analiza las imágenes a múltiples resoluciones sin aplicar submuestreo, aunque en escenas donde el objetivo está muy cerca de las cámaras el algoritmo con submuestreo puede resultar útil ya que las disparidades entre las imágenes son muy grandes y después de reducir el tamaño de estas se observa que los puntos correspondientes más cercanos si tienen disparidades más grandes a las del resto de la imagen.

Una vez que se obtuvo el mapa de disparidades, es necesario obtener la profundidad (coordenada  $z$ ) en términos de unidades reales, para lo cual fue necesario realizar la calibración de la cámara con un método muy sencillo. Los métodos estudiados en este trabajo pueden ser utilizados en una gran variedad de sistemas de visión que estén enfocados a obtener distancias sin importar otras características de las imágenes que forman la escena.

## **Bibliografía**

Mayhew, J.E.W. and Frisby, J.P., 3D Model Recognition from Estereoscopic Cues, MIT Press.

Pratt, William K., Digital Image Processing, Second Edition, Sun Microsystems Inc. Mountain View California, A Wiley - Interscience Publication, John Wiley & Sons Inc.

Russell, Stewart J., and Norwing Peter, Artificial Intelligence, a Modern Approach, Prentice Hall, Englewood Cliffs: New Jersey Series in Artificial Intelligence.

Sanfeliu, Alberto, Inteligencia Artificial, Conceptos, Técnicas y Aplicaciones, Serie Mundo Electrónico, Marcombo Bioxareo Editores, 1987.

Shalkoof, Robert J., Digital Image Processing and Computer Vision, Department of Electrical and Computer Engineering, Heroson University, John Wiley & Sons Inc.

## **Hemerografía**

Calway, A.D., Knutsson H., and Wilson, R., Multiresolution Estimation of 2-D Disparity using a Frequency Domain Approach, Dept. of Computer Science, university of Warwick, Coventry CV4 7AL, England. Computer Vision Laboratory, Linlöping University, Sweeden.

Jenkin, Michael R.M., Department of Computer Science, York University, Toronto, Ontario, Canada, Jepson, Allan D, Tsotsos John K., Department of Computer Science , University of Toronto, Toronto, Ontario, Canada, Techniques for Disparity Measurment, CVGIP: Image Understanding, Vol 59, No. 1, January, pp 72 – 93 , 1994.

Jenkin, Michael R.M., Department of Computer Science, York University, Toronto, Ontario, Canada, and Jepson, Allan D, Department of Computer Science , University of Toronto, Toronto, Ontario, Canada, Recovering Local Surface Structure through Local Phase Difference Measurements, CVGIP: Image Understanding, Vol 59, No. 1, January, pp 72 – 93, 1994.

Young, David, Octubre 1989, Apuntes de la Maestría en Sistemas basados en el Conocimiento, Universidad de Sussex, Inglaterra, basados en Pollard, S.B, Porrill, J., Mayhew, J.E.W., & Frisby, J.P., “Disparity Gradient, Lipschitz Continuity and Computing Binocular Correspondences, University of Sheffield ( AIVRU ref No 10 ), in Proceedings of the 3<sup>rd</sup> International Symposium of Robotics Research.

Witkin, Andrew P., Scale space filtering: a new approach to multi - scale description, Fairchild Laboratory for Artificial Intelligence Research, 1984.

Escalante, Boris, Noise Reduction in Computed Tomography Images by means of Pollinomial Tranforms, Journal of Visual Communications and Image Representation, Vol. 3, No. 3, Septiembre, pp 272 - 285, 1992.

Burt,P.J., and Adelsen, E.H., The Laplacian Pyramid as a Compact Image Code, IEEE Transactions on Communications, Vol. COM - 31, pp 532 - 540,1983.

## **Páginas de Internet Consultadas**

Pérez de la Blanca Capilla, Nicolas, Fundamentos del Tratamiento de Imágenes. Visión por Computador, 3 de Abril de 1995. Dirección Internet:  
<http://bucanero.ugr.es/wwwcurso/ccordoba/>

Young, David, Sussex Computer Vision Teach Files, February 1993 revised January 1994. Dirección Internet:  
<http://www.cogs.susx.ac.uk/users/davidy/teachvision/>

Schneider Fontán, Miguel, Comportamientos Motores de Visión, Sep 18, 1996.  
Dirección Internet:  
<http://mercurio.iai.csic.es/users/msf/Documentos/Vision/main/>

Stereoscopic Imagery.

Dirección Internet:  
<http://blackhole.tisco.com/3d-web/papers/stereo.htm>