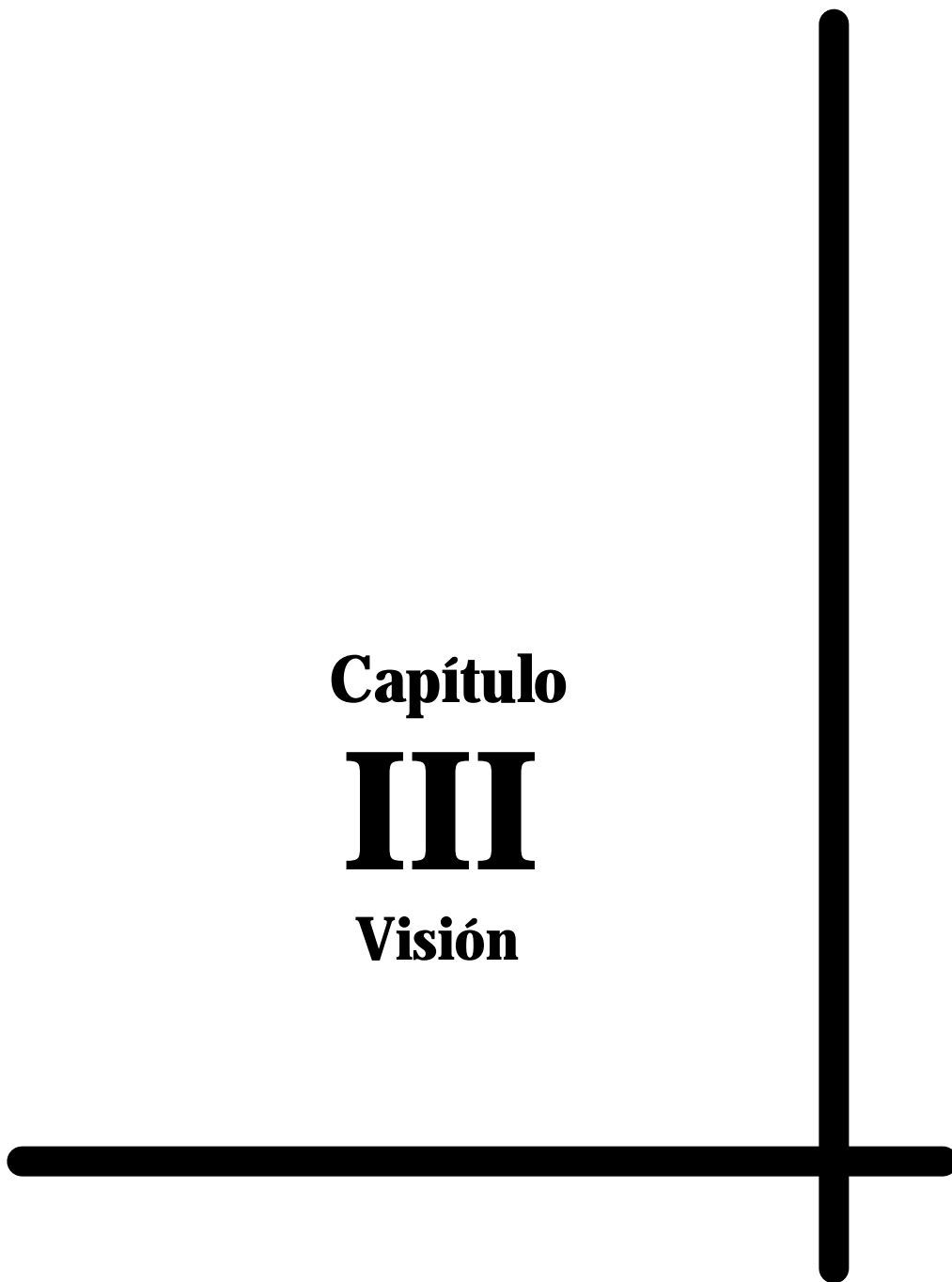


Capítulo

III

Visión



III

3.1 Procesamiento de Imágenes

Se entiende por procesamiento de imágenes, a la alteración y análisis de la información gráfica.

3.1.1 Sistema de visión humano

El sistema de procesamiento de imágenes más poderoso en la actualidad es aquel conformado por nuestros ojos y cerebro.

Una aproximación a la construcción de robots que desempeñen tareas como el reconocimiento, toma objetos, manipulación y evasión de obstáculos consiste en el estudio del desempeño de las tareas del humano.

Haciendo una analogía del sistema de visión humano con sistemas de visión artificiales, se puede asimilar al ojo con los dispositivos formadores de imagen, donde ambos son los encargados de captar la imagen del ambiente. Los rayos de luz generados por una escena, son colectados por la lente en el caso del ojo la lente es llamado cristalino. El iris sirve como control de la intensidad de luz y en las cámaras el encargado es el diafragma. En el caso del ojo humano los elementos sensores son llamados bastones y conos y realizan el mismo trabajo que los sensores C-MOS en las cámaras digitales modernas.

El encargado de procesar la información de los dispositivos de entrada es la computadora y en el caso del ser humano, el cerebro descifra los impulsos nerviosos para convertirlos en las imágenes que nosotros percibimos.

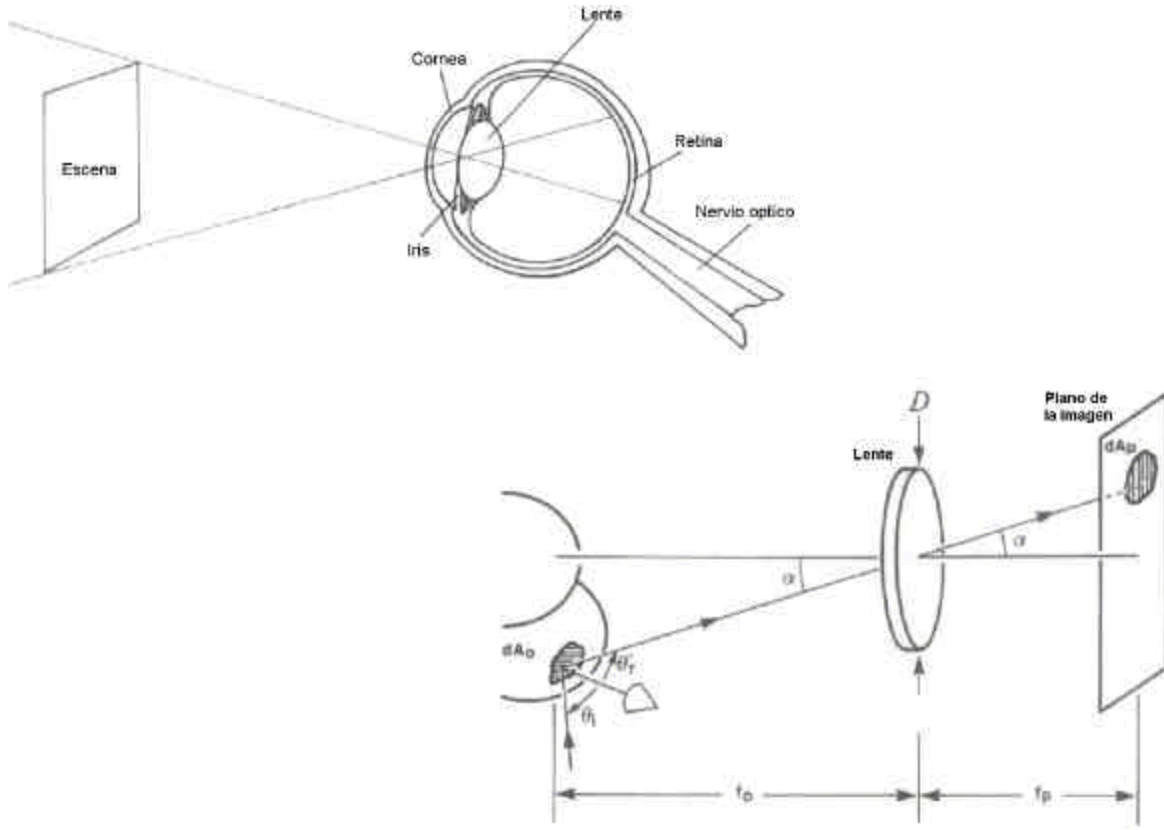


Fig. 3.1 Comparación ojo humano contra sistema formador de imagen

El entendimiento de la operación del sistema de visión humano nos sirve para definir las operaciones a realizar en el procesamiento digital de Imágenes [Baxes '94].

3.1.2 Procesamiento Digital de Imágenes

Gracias a la reciente disponibilidad de sofisticados dispositivos semiconductores y compactas y poderosas computadoras junto a avances en algoritmos de procesamiento de imágenes, se ha atraído la atención del procesamiento de imágenes digital.

Un sistema de procesamiento digital de imágenes es una colección de dispositivos de hardware que provee la digitalización, almacenamiento, procesamiento, despliegue de las imágenes digitales y de los resultados que se obtengan con el procesamiento.

Las rutinas de los sistemas procesadores de imágenes pueden estar ya sea almacenadas en un programa y/o también pueden ser realizadas mediante hardware específico.

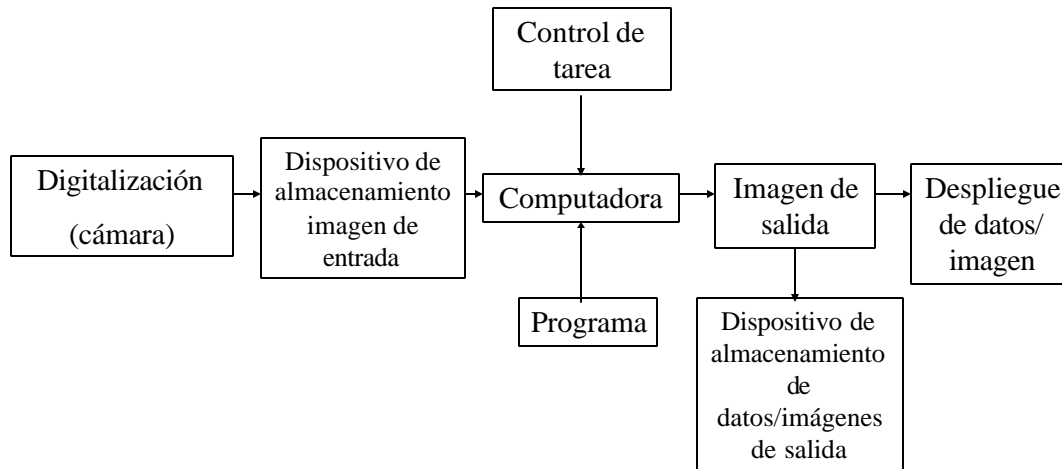


Fig. 3.2 Diagrama a bloques sistema de procesamiento digital de imágenes.

El Procesamiento digital de imágenes puede dividirse en 3 grandes grupos de operaciones que son: operaciones de mejoramiento de la calidad, operaciones de análisis y operaciones de codificación de las imágenes.

Las primeras, subjetiva u objetivamente modifican la apariencia o calidades de la imagen, a su vez, este tipo de operaciones se sub-divide en operaciones de contraste y operaciones espaciales; las operaciones de contraste tratan con la alteración de brillo en una imagen, los negros, blancos y grises pueden ser realzados o suprimidos; las operaciones espaciales modifican el contenido del detalle de una imagen, los bordes por ejemplo pueden ser acentuados con este tipo de operaciones; las operaciones de análisis de la imagen producen información basada en la imagen, es decir la salida de estas operaciones no son de tipo pictórico, en su lugar, la salida es información numérica o gráfica basada en las características originales de la imagen, con el objetivo de describir algún aspecto de la imagen, este tipo de operaciones pueden ser utilizadas para la descripción de características de la

escena, medición automática de objetos y reconocimiento de patrones, la operación más común es el histograma; las operaciones de codificación de la imagen pueden servir para reducir la cantidad de información necesaria para describir una imagen, en este sentido existen dos tipos de operaciones de codificación, la primera codifica la imagen de manera que no hay pérdida de información, la segunda puede codificar la imagen de manera que por ejemplo rompa la imagen en secciones o estructuras codificando únicamente la localización y orientación de las partes, esta forma reduce considerablemente la cantidad de datos, lo cual puede traer como consecuencia pérdida de la representación original de la imagen.

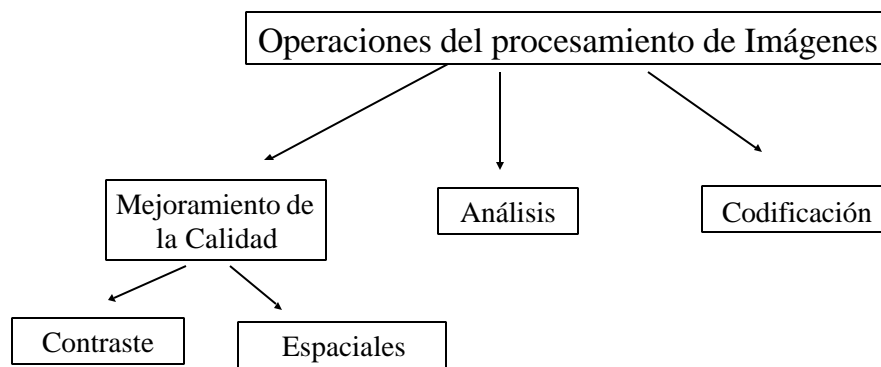


Fig. 3.3 Clasificación de las operaciones del procesamiento de imágenes

Una de las áreas de mayor interés dentro del procesamiento digital de imágenes, es la de mejoramiento de la calidad de imagen, esto es porque en casi todos los casos es necesario aplicar algún tipo de mejora a la imagen antes de continuar con otros procesos en la aplicación.

En el procesado de la imagen total, hay tres grandes categorías de procesamiento, la de punto, la de grupo y una tercera llamada procesamiento de frame, la primera sirve para implementar las operaciones de contraste, mientras que la segunda sirve para realizar las operaciones espaciales; las operaciones de punto se subdividen en operaciones realizadas con una sola imagen o monádicas y operaciones realizadas con dos imágenes o diádicas, en el tercer grupo llamado procesamiento de frame, se han agrupado operaciones como el zoom y el stretch, para las operaciones de punto y de grupo (que también corresponden a las operaciones de mejoramiento de la calidad de las imágenes),

existe equipo especializado que realiza estas operaciones mediante hardware debido a que se utilizan frecuentemente; al hacer esto se reduce el tiempo empleado en dichas operaciones.

[Baxes '94]

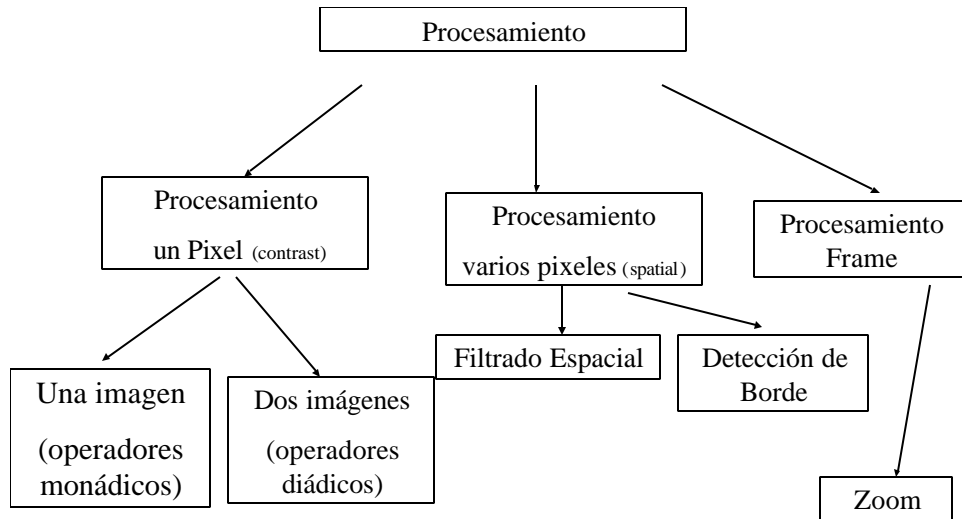


Fig. 3.4 Clasificación del procesamiento de la imagen en su totalidad

3.2 Detección de Borde: Gradiente y Laplaciano

Los bordes son cambios locales significantes en la imagen y son muy importantes para el análisis de las mismas. Los bordes se encuentran típicamente localizados en los límites entre dos diferentes regiones en una imagen. La detección de borde es frecuentemente el primer paso en la recuperación de información de las imágenes. Un borde es un cambio local significativo en la intensidad de la imagen, usualmente asociadas con una discontinuidad en la intensidad de la imagen o primera derivada de la intensidad; Los bordes son pixeles con una gradiente grande; Las discontinuidades pueden ser: discontinuidades de escalón, en donde la intensidad de la imagen abruptamente cambia de un valor de un lado de la discontinuidad a un valor diferente en el lado opuesto, discontinuidad de línea, donde la intensidad de la imagen cambia abruptamente pero regresa al valor inicial en una distancia corta; este tipo de discontinuidades sin embargo, poco existe en la realidad, en donde los escalones son rampas y las discontinuidades de líneas son dobles rampas.

Un algoritmo de detección de borde es aquel que produce un conjunto de puntos de borde o fragmentos de borde de una imagen.

Los algoritmos involucrados en la detección de bordes contienen tres pasos, que son el filtrado, realce y detección propiamente dicha.

La aproximación discreta a la gradiente es muy susceptible al ruido, debido a que está basada en la comparación de variación de intensidades por lo que el filtrado es comúnmente utilizado para mejorar el desempeño de un detector de borde, sin embargo, un filtrado excesivo contribuye a una pérdida en la calidad del borde, es decir, existe un compromiso entre la eliminación del ruido y pérdida de los bordes.

El realce consiste en enfatizar los píxeles cuando exista un cambio local en el valor de la intensidad y usualmente es hecho al computar la magnitud de la gradiente.

La detección más sencilla consiste en fijar un umbral que permita discernir entre las detecciones que pueden ser consideradas bordes y las que no.

Algunos algoritmos incluyen un cuarto paso que consiste en la localización del borde, es decir: la dirección en la cual se encuentra un cambio de contraste, el detectar los bordes puede ser de gran ayuda en el momento de toma de decisiones.

3.2.1 Gradiente

En una dimensión, un borde de escalón es asociado con un pico local en la primera derivada. La gradiente es una medida de cambio en una función, una imagen puede ser considerada como un arreglo de muestras de alguna función continua de la intensidad de una imagen, Cambios significativos en los niveles de gris, pueden ser detectados usando una aproximación discreta a la gradiente. La gradiente es el equivalente bi-dimensional de la primera derivada y es definida como el vector:

$$G[f(x, y)] = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix}$$

Ec. 3.1 Gradiente.

Dos propiedades importantes son asociadas con la gradiente: el vector $G[f(x,y)]$ que apunta en la dirección del cambio máximo de la función $f(x,y)$ y la magnitud de la gradiente (ver ec. 3.2) que es la “fuerza” del borde.

$$G[f(x, y)] = \sqrt{G_x^2 + G_y^2}$$

Ec. 3.2 Magnitud de la gradiente.

La magnitud de la gradiente es el máximo rango de cambio por unidad de distancia en la dirección G; es decir, la pendiente del cambio de la función. Para propósitos prácticos, la gradiente puede ser aproximada a las funciones como se muestran en la ec. 3.3

$$G[f(x, y)] = |G_x| + |G_y|$$

$$G[f(x, y)] = \max(|G_x|, |G_y|)$$

Ec. 3.3 Aproximación de la gradiente.

La dirección de la gradiente, para el análisis vectorial es definida como: (ver ec. 3.4):

$$\theta(x, y) = \text{atan}\left(\frac{G_y}{G_x}\right)$$

Ec. 3.4 Dirección de la gradiente

Para imágenes digitales, las derivadas pueden ser aproximadas como se muestran en la ecuación 3.5.

$$G_x = f[i, j+1] - f[i, j]$$

$$G_y = f[i, j] - f[i-1, j]$$

Ec. 3.5 Aproximaciones digitales más simples de la Gradiente.

Los detectores de borde basados en la gradiente son los métodos Roberts, Sobel y Prewitt que a continuación se describen.

Operador Roberts

El operador Roberts en cruz provee una simple aproximación a la magnitud de la gradiente:

$$G[f[i, j]] = |f[i, j] - f[i+1, j+1]| + |f[i-1, j] - f[i, j+1]|$$

Ec. 3.6 Operador Roberts

Usando máscaras de convolución es:

$$G[f(x, y)] = |G_x| + |G_y|$$

Ec. 3.7 Aproximación del operador Roberts

Donde G_x y G_y pueden ser calculadas mediante las siguientes máscaras:

$$G_x = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \quad G_y = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$$

Fig. 3.5 Máscaras Roberts

En realidad las diferencias son computadas en el punto interpolado $[i + 1/2, j + 1/2]$. Y no en el punto $[i, j]$ como debería de esperarse.

Operador Prewitt

Para evitar cálculos en puntos interpolados no existentes, se considera el uso de máscaras 3x3 para el cálculo de la gradiente.

El operador Prewitt es la magnitud de la gradiente calculada por:

$$M = \sqrt{S_x^2 + S_y^2}$$

Ec. 3.8 Magnitud de la Gradiente operador Prewitt

Donde las derivadas parciales son computadas mediante (ver ec. 3.9):

$$S_x = (a_2 + ca_3 + a_4) - (a_0 + ca_7 + a_6)$$

$$S_y = (a_0 + ca_1 + a_2) - (a_6 + ca_5 + a_4)$$

Ec. 3.9 Derivadas parciales operador Prewitt.

Y donde c es una constante igual a 1.

Los coeficientes de las ecuaciones anteriores pueden ser determinados al utilizar la siguiente referencia de nombramiento de los pixeles cercanos (ver Fig. 3.6)

a0	a1	a2
a7	[i,j]	a3
a6	a5	a4

Fig. 3.6 Nombramiento vecinos operadores Prewitt y Sobel

Como otros operadores gradiente, S_x y S_y pueden ser implantados usando máscaras de convolución.

$$\mathbf{S_x} = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix} \quad \mathbf{S_y} = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix}$$

Fig. 3.7 Máscaras operador Prewitt

Operador Sobel

El operador Sobel usa las mismas ecuaciones que el operador Prewitt, excepto que la constante c tiene un valor igual a 2

$$\mathbf{S_x} = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad \mathbf{S_y} = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

Fig. 3.8 Máscaras operador Sobel

Hay que hacer notar que este método pone especial énfasis en los pixeles más cercanos al centro de la máscara. El operador Sobel es uno de los métodos más comúnmente utilizados.

[Jain '95] Cuando las imágenes originales tienen un buen contraste y con poco ruido, este es el mejor método adaptado para la detección de bordes, además de que su tiempo de cómputo es reducido.

[Ibanes '98]

3.2.2 Operadores de la Segunda Derivada

Cuando se utiliza un valor umbral para determinar si existe o no un borde, se puede llegar a detectar muchos puntos, por lo que se producen bordes gruesos; una forma de eliminar esto es encontrar únicamente los puntos que tienen un máximo local en los valores de gradiente y considerar entonces éstos como puntos de borde.

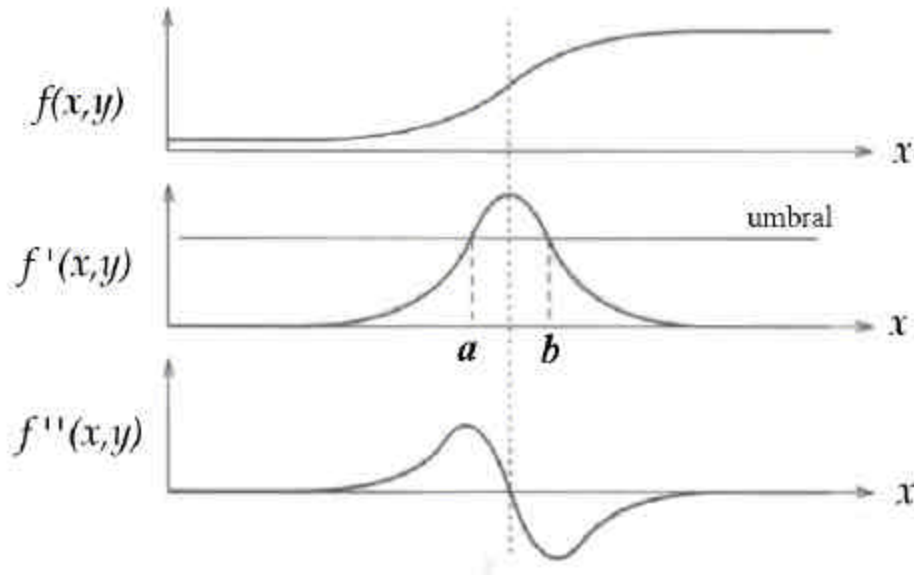


Fig. 3.9 Representación gráfica 1ª y 2ª derivada

Lo anterior significa que los bordes serán un pico en la primera derivada y mediante la segunda derivada de la intensidad de la imagen, los bordes pueden ser detectados al encontrar los cruces por cero.

Una desventaja de la segunda derivada es que es muy sensible al ruido, por lo que el filtrado es realizado previo a la detección del borde.

Existen dos operadores en dos dimensiones que corresponden a la segunda derivada: El Laplaciano y la segunda derivativa direccional.

Operador Laplaciano

El Laplaciano es el equivalente bidimensional de la segunda derivada, la fórmula de la función Laplaciana $f(x,y)$ es:

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$

Ec. 3.10 Ecuación del Laplaciano

Al remover los puntos que no corresponden a un máximo local en la primera derivada, los bordes pueden ser detectados mas eficientemente.

La segunda derivada a lo largo de x y y puede ser aproximada usando las ecuaciones de diferencia en donde:

$$\frac{\partial^2 f}{\partial x^2} \approx \frac{f[i, j+1] - 2f[i, j] + f[i, j-1]}{\Delta x^2}$$

Ec. 3.11 Segunda derivada parcial con respecto a x (aproximación digital)

sustituyendo $j-1$ por j

$$\frac{\partial^2 f}{\partial x^2} \approx \frac{f[i, j+1] - 2f[i, j] + f[i, j-1]}{\Delta x^2}$$

Similarmente se obtiene:

$$\frac{\partial^2 f}{\partial y^2} \approx \frac{f[i-1, j] - 2f[i, j] + f[i+1, j]}{\Delta y^2}$$

Ec. 3.12 Segunda derivada parcial con respecto a y (aproximación digital)

La siguiente máscara puede ser usada para la aproximación al Laplaciano:

0	1	0
1	-4	1
0	1	0

Fig. 3.10 Máscara Laplaciana

La siguiente máscara da más énfasis en los pixeles del centro:

1	4	1
4	-20	4
1	4	1

Fig. 3.11 Máscara Laplaciana con énfasis en el centro

Operador Segunda Derivativa Direccional

La segunda derivativa direccional es la segunda derivada calculada en la dirección de la gradiente, el operador es implementado usando la fórmula:

$$\frac{\partial^2}{\partial n^2} = \frac{f_x^2 - f_{xx} + 2f_x f_y f_{xy} - f_y^2 - f_{yy}}{f_x^2 + f_y^2}$$

El Laplaciano y derivativa direccional de segundo orden, son poco usados en visión por computadora debido a que son muy susceptibles al ruido más que los operadores involucrando la primera derivada. En su lugar son usados algoritmos que combinan el filtrado Gaussiano con la segunda derivada para la detección de borde.

Laplaciano de Gaussiano

El Laplaciano del Gaussiano combina el filtrado Gaussiano con el Laplaciano para la detección de bordes. Este método consiste en tres pasos, los cuales son: filtrado, realce y detección.

El filtrado para suavizar la imagen es Gaussiano

El paso de realce es la segunda derivada

El criterio de detección es la presencia de un cruce por cero en la segunda derivada con el correspondiente pico en la primera derivada

La localización del borde puede ser estimada a nivel subpixel con interpolación lineal

Al hacer el filtrado se eliminan las pequeñas estructuras debido al ruido, pero los bordes son ensanchados, por lo que son considerados como bordes aquellos que producen un máximo local; esto es logrado al detectar el cruce por cero en la segunda derivada, para evitar la detección de bordes insignificantes, únicamente los cruces por cero que se cuentan también en su primera derivada con un valor por encima del valor umbral, son seleccionados como borde. La dirección del borde es obtenida usando la dirección en la que el cruce por cero ocurre.

El filtrado Gaussiano es una clase de filtrado lineal suave en el que los valores son escogidos de acuerdo a la forma de una función Gaussiana, las características principales consisten en que este filtro es efectivo como pasa-bajas desde el punto de vista espacial y dominio de las frecuencias; es fácil de

implementar, puede ser usado efectivamente en aplicaciones de visión, es rotacionalmente simétrico, tiene un solo lóbulo, es decir: cada valor es reemplazado por un promedio ponderado de los pixeles vecinos tal que el peso dado a cada vecino decrece monótonicamente con la distancia del pixel central; La transformada Fourier de un Gaussiano tiene un solo lóbulo en el dominio de la frecuencia, la transformada Fourier de un Gaussiano es en si mismo un Gaussiano. La anchura del filtrado es directamente proporcional a su σ , Filtros grandes Gaussianos pueden ser implementados en un orden lineal al separar sus funciones bidimensionales como dos uni-dimensionales.

La siguiente es una máscara clásica de la aplicación de un filtro del tipo Gaussiano

1	1	2	2	2	1	1
1	2	2	4	2	2	1
2	2	4	8	4	2	2
2	4	8	16	8	4	2
2	2	4	8	4	2	2
1	2	2	4	2	2	1
1	1	2	2	2	1	1

Fig. 3.12 Máscara filtro Gaussiano

La salida del Laplaciano del Gaussiano (LoG) es obtenida mediante la operación de convolución

$$LoG = \nabla^2 [g(x, y) * f(x, y)]$$

$$LoG = \nabla^2 g(x, y) * f(x, y)$$

Donde:

$$\nabla^2 g(x, y) = \left\{ \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} \right\} \frac{e^{-\frac{(x^2+y^2)}{2\sigma^2}}}{\sigma^4}$$

es llamado el operador del sombrero

Para poder hacer lo anterior, es posible realizarlo mediante dos métodos matemáticamente equivalentes, el primero consiste en aplicar un filtro del tipo Gaussiano y a la salida de éste aplicar el Laplaciano; el segundo método consiste en aplicar máscaras que implementen directamente el Laplaciano del Gaussiano, un ejemplo de éste tipo de máscaras se muestra a continuación:

0	0	-1	0	0
0	-1	-2	-1	0
-1	-2	16	-2	-1
0	-1	-2	-1	0
0	0	-1	0	0

Fig. 3.13 Máscara Laplaciano del Gaussiano

Un buen método que conserva un buen rechazo al ruido mientras que detecta de manera muy efectiva los bordes, es aplicar un filtrado Gaussiano y aplicar el método de la gradiente. Sin embargo; esto no es rotacionalmente simétrico, siendo su antisimetría perpendicular a el borde.

[Jain '95]

Detector de Borde Canny

El detector de Borde Canny es la primera derivada de la Gaussiana y se aproxima al operador que optimiza el producto de señal a ruido y localización. El resultado de la convolución de la imagen con un filtro pasabajas Gaussiano que es un filtrado separable, es un arreglo de datos suavizados

$$S[i, j] = G[i, j; \sigma] * I[i, j]$$

Donde σ es la amplitud del Gaussiano y controla el grado de suavizamiento. La Gradiente del arreglo suavizado $S[i, j]$ puede ser calculado usando una aproximación de primera diferencia de 2×2 , lo cual produce dos arreglos $P[i, j]$ y $Q[i, j]$, para la derivada parcial x y y .

$$P[i, j] = \frac{(S[i, j+1] - S[i, j]) - (S[i-1, j+1] - S[i-1, j])}{2}$$

$$Q[i, j] = \frac{(S[i, j] - S[i-1, j]) - (S[i, j-1] - S[i-1, j-1])}{2}$$

La orientación y la magnitud pueden ser calculadas mediante las siguientes fórmulas:

$$M[i, j] = \sqrt{P[i, j]^2 + Q[i, j]^2}$$

$$\theta[i, j] = \arctan2(Q[i, j], P[i, j])$$

El arreglo de Magnitud M , contiene valores grandes donde la gradiente de la imagen es grande, así que podría aplicarse un umbral para obtener los bordes, sin embargo, el Algoritmo Canny tiene una forma mas sofisticada para encontrar los bordes, un punto de borde es definido como un punto cuya magnitud es localmente máxima en la dirección de la Gradiente, es decir, el pico del borde; esto conduce a tener bordes de un pixel de ancho y por lo tanto mas delgados y el proceso es llamado supresión de los no máximos (nonmáxima supresión).

La supresión de los no máximos adelgaza los bordes de la gradiente al suprimir los valores a lo largo de la línea de la gradiente que no son valores pico de un borde. El algoritmo inicia al reducir el ángulo de la gradiente $\theta[i,j]$ a uno de cuatro sectores.

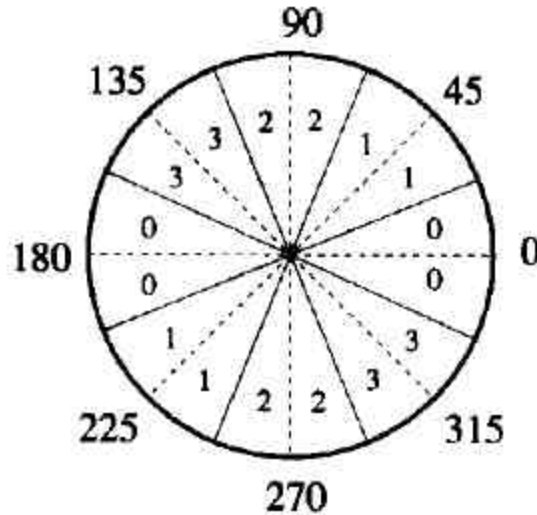


Figura 3.14 Sectores

$\theta[i, j] = \text{Sector}(\theta[i, j])$

En cada punto, el elemento central $M[i,j]$ es comparado con sus 2 vecinos a lo largo de la línea de la gradiente dada por el valor del sector. Si el valor del arreglo magnitud $M[i,j]$ en el centro no es mayor que sus vecinos, entonces $N[i,j]$ es cero y viceversa, Este proceso adelgaza los anchos bordes en bordes que son únicamente un pixel de ancho.

$N[i, j] = \text{nms}(M[i, j], \theta[i, j])$

$N[i,j]$ contendrá varios bordes falsos causados por el ruido y texturas finas.

Estos falsos fragmentos deben ser reducidos, esto puede ser realizado al aplicar un umbral, debido a que el contraste de los falsos bordes es pequeño; todos los valores por debajo del umbral son iguales a cero y aquellos por encima son uno; el resultado es un arreglo $I[i,j]$ de los bordes detectados en la imagen, el hallar un umbral propio es difícil y normalmente es el resultado de prueba y error, debido a esta dificultad, la imagen final puede tener aun pequeños fragmentos si el umbral es muy bajo o resultan en bordes faltantes si el umbral es alto, sin embargo una manera más eficiente de hallar este borde es el uso de dos umbrales.

Con dos umbrales, dos imágenes son producidas, una con contornos discontinuos, pero casi sin bordes falsos (umbral alto, T2) y otra (hecha con un umbral de la mitad del de la primera T1) con varios bordes falsos. Con la imagen hecha a partir del umbral alto (T2) los bordes son encadenados, cuando se encuentra el fin de un contorno, el algoritmo busca en T1 los 8 pixeles vecino por bordes que pueden ser encadenados al contorno, El algoritmo continúa hasta que la discontinuidad es encadenada a un borde.

De esta forma se resuelve el problema de tener que escoger un umbral.

Algoritmo Hough

Este algoritmo es usado para obtener en coordenadas polares los vectores correspondientes a las características más importantes que se encuentran en una imagen, en el caso de este trabajo las características vectoriales que se desean encontrar corresponden a los bordes.

El algoritmo es el siguiente: se escoge el centro de la imagen como punto de partida para realizar todas las operaciones posteriores; ahora bien, se toma la primera característica, con coordenada (x,y) a este punto se le asigna un ángulo θ (que irá variando) lo que se trata de encontrar es una r tal que la coordenada polar (r, θ) (siendo $\theta = \theta + 90^\circ$) sea un punto perteneciente a la ecuación dada por (x,y,θ) , existe un arreglo A de dimensiones $[\text{radio_maximo}, 360]$, el cual en su posición (r, θ) se incrementará en uno para indicar la presencia de una característica cuya ecuación (x,y,θ) pasa por esa coordenada polar.

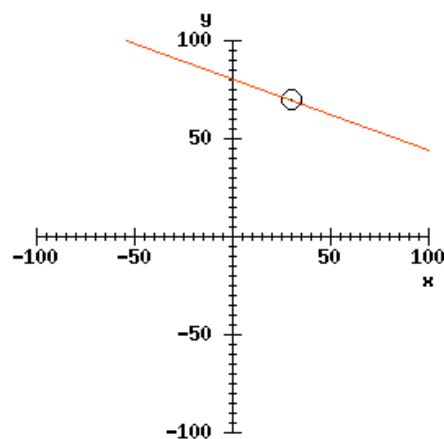


Fig. 3.15 Un punto x,y con una θ

El ángulo θ tendrá varios valores para un mismo punto (x,y) , al hacer esto, se obtienen varios pares de coordenadas (r, θ) para un mismo punto.

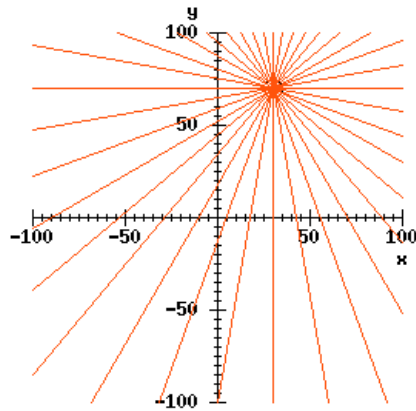


Fig. 3.16 Varias θ para un punto x,y

Lo anterior se repite para otro punto de borde o característica. Ahora bien, si existen varios puntos cuyos pares (r,θ) resultan iguales, entonces se encuentran determinando un mismo borde y es, por tanto, este borde representativo, es decir, si el arreglo A en $[r, \theta]$ contiene valores por encima de cierto umbral, será entonces que ese r y θ determinarán las coordenadas del vector que representa una característica.

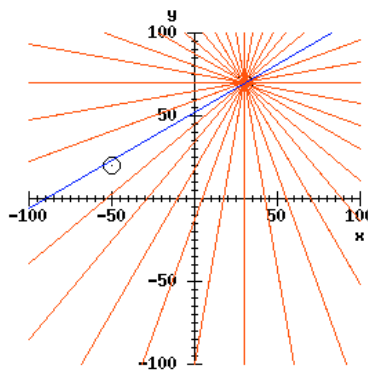


Fig. 3.17 Par r, θ que determina un vector

3.3 Algoritmos de Emparejamiento Estéreo

Los algoritmos de emparejamiento estéreo se basan en el hecho de que cada punto en la escena se observa idéntico en las dos imágenes. Al encontrar el punto correspondiente de una imagen en la otra es posible determinar su posición en el espacio tridimensional.

El problema del emparejamiento estéreo, supone el hacer pares con los píxeles de la línea de una imagen que se escanea contra los píxeles en la línea de la otra imagen, asumiendo que las dos cámaras están alineadas. La correspondencia es alcanzada mediante una secuencia de emparejamientos de las imágenes izquierda y derecha del mismo punto de interés, los píxeles sin emparejamientos tienen algún tipo de oclusión.

Para determinar que dos puntos forman un par conjugado, es necesario medir la similitud de los puntos, claramente, los puntos a ser emparejados, deben ser diferentes de los píxeles que le rodean.

En los algoritmos estéreo es necesario localizar características emparejables; las características de borde y de región han sido usadas por varios autores para el emparejamiento estéreo.

Existen varias maneras de medir la similitud de las características, por ejemplo, algunos sistemas hacen uso de la orientación del borde. La mayoría sin embargo adoptan el uso de la similitud basada en la escala de grises en la imagen en una máscara de píxeles alrededor de la característica; se supone que las regiones a emparejar tienen una estructura local similar y un nivel de gris general; para evitar diferencias positivas y negativas, normalmente se obtiene el cuadrado de cada diferencia antes de realizar la adición, se buscan en cada uno de los lados de la característica y se acepta un emparejamiento si ambas diferencias son pequeñas, los bordes horizontales no son buenos para el emparejamiento y es mejorado si es usada la información de la orientación de borde además de la información de la escala de gris.

La profundidad es computada únicamente con los pares obtenidos y los demás puntos de la escena son calculados mediante interpolación.

Verdaderas discontinuidades ocurren en cambios de varios píxeles o más.

Lo anterior es el procedimiento del algoritmo estándar del matching entre dos escenas; existen pequeñas variantes sobre este proceso y un derivado es descrito a continuación

Pixel by pixel birchfield

El objetivo de este método, es detectar las discontinuidades mas que computar un mapa exacto de profundidad; prefiere asignar una constante de disparidad a cada objeto, aun cuando la profundidad de ese objeto varia ligeramente como es en el caso de un cilindro

Con cada línea que se escanea, cada objeto contiene al menos una ligera variación de la intensidad.

El costo de una secuencia de emparejamiento M es una penalización por cada oclusión, una ganancia por cada emparejamiento bueno y la suma de sus disimilitudes entre los pixeles emparejados.

El basarse únicamente en la intensidad de grises puede verse afectada por el muestreo de los sensores en los dispositivos de entrada; una manera de enfrentar este problema es el utilizar el emparejamiento a través de ventanas. Es propuesto el uso de la función de la intensidad linealmente interpolada de los dos pixeles adyacentes para medir la similitud de los pixeles. Esto es, al haber errores en el sensor, aunque el error en el sensor alrededor del punto de interés sea cóncavo o convexo, al trabajar con los pixeles adyacentes se aproxima al uso de una función lineal. Esto en la práctica causa que los puntos de inflexión no causen problemas debido a que las regiones circundantes son aproximadamente lineales.

Existen dos algoritmos de búsqueda de emparejamiento, el de búsqueda hacia atrás y el de búsqueda hacia delante, en el primero cuando se ha encontrado un par de puntos conjugados se realiza una búsqueda hacia todos sus posibles emparejamientos, para determinar cual conduce a la mejor secuencia de emparejamiento; el de búsqueda hacia delante consiste en tomar la menor diferencia entre dos argumentos, y este se compara contra un tercer valor, de esta comparación se obtiene el menor y se compara contra un cuarto elemento y así sucesivamente hasta que al final del algoritmo se obtiene el mejor par conjugado de puntos con la menor diferencia.

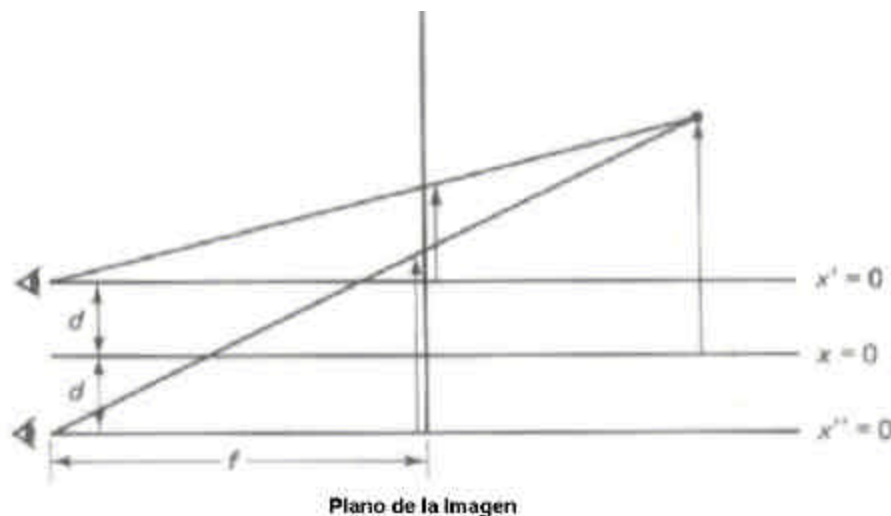
A cada objeto se le asigna una sola disparidad, con este comportamiento se sacrifica una reconstrucción exacta de la profundidad.

[Birchfield '96]

3.4 Visión Estéreo

Haciendo la analogía con el sistema de visión humana, los sistemas de visión estereoscópica en robótica utilizan dos cámaras de características similares montadas comúnmente a una distancia horizontal de 7 cms. Aunque también son comunes los sistemas donde las cámaras son montadas a 10, 12, 14 y 20 cms.

Ver una escena desde dos o más posiciones diferentes simultáneamente, permite hacer inferencias acerca de la estructura 3-D de lo que se observa, esto es logrado al detectar puntos de referencia en las imágenes. Los sistemas de visión de humanos y animales hacen uso de algo muy similar para poder percibir la tridimensionalidad del mundo, por lo que la investigación en esta línea es muy importante en los intentos por desarrollar sistemas de visión por computadora.



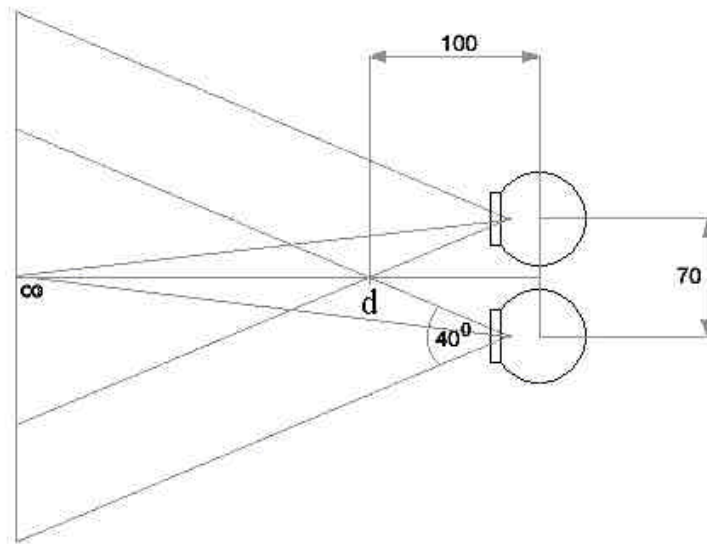


Fig. 3.18 Diagramas visión estereoscópica en robótica

Al analizar dos imágenes obtenidas como se describió anteriormente, es posible observar diferencias o cambio en la posición de los objetos de interés respecto al resto del ambiente, este cambio es el que nos ayuda a tener un conocimiento tridimensional de la escena de interés.

Hay dos consideraciones importantes computacionalmente hablando, la primera es cómo las diferencias en imagen son trasladadas a información 3-D y la segunda cómo las imágenes pueden ser emparejadas una a otra para que las diferencias entre ellas puedan ser medidas.

La geometría asociada a esta forma de detectar la tridimensionalidad de los objetos es descrita a continuación.

Se asume cámaras ideales con una imagen plana sin distorsión, la relación entre un punto en el espacio y la captada por la cámara es:

$$x = \frac{(X)(f)}{Z}$$

$$y = \frac{(Y)(f)}{Z}$$

Ec. 3.13 Ecuaciones proyección perspectiva

Esto parte de la idea de que una cámara con su lente se comporta aproximadamente como una cámara de abertura, con excepción de que produce una imagen más brillante. Se dibuja una línea recta del

punto en el espacio a través del centro de la lente a la imagen producida, para un punto con un nivel $y=0$. El diagrama resultante es mostrado a continuación.

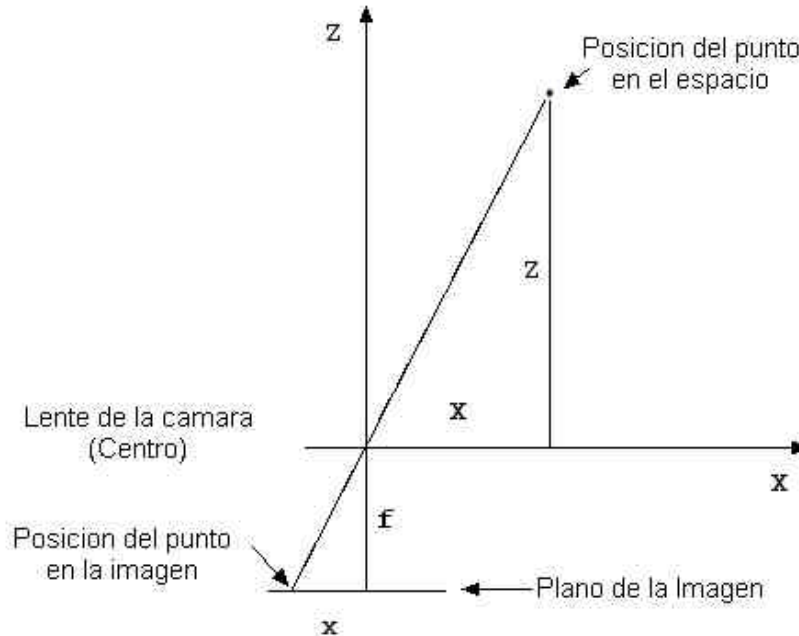


Fig. 3.19 Proyeccion Perspectiva

Nota: Se ha adoptado por posicionar la imagen a la misma distancia que f pero entre la imagen y la lente, debido a que como se había planteado originalmente lo que se obtenía era una imagen invertida. Es decir, se forma una imagen virtual la cual está frente a la lente en lugar de atrás y la imagen rotada al derecho.

La relación entre x y X nos conduce a triángulos similares se tiene que:

$$\frac{x}{f} = \frac{X}{Z}$$

Se asume que se tienen 2 cámaras apuntando a una escena de interés, es más simple si se realizan las ecuaciones para las cámaras cuando sus ejes se encuentran paralelos entre sí y sus coordenadas Y son las mismas. La posición en el espacio puede ser descrita por cualquiera de los dos sistemas de las cámaras, pero es más conveniente si se describe a partir de un sistema colocado justo en medio de las 2 cámaras llamado eje de coordenadas ojo de cíclope.

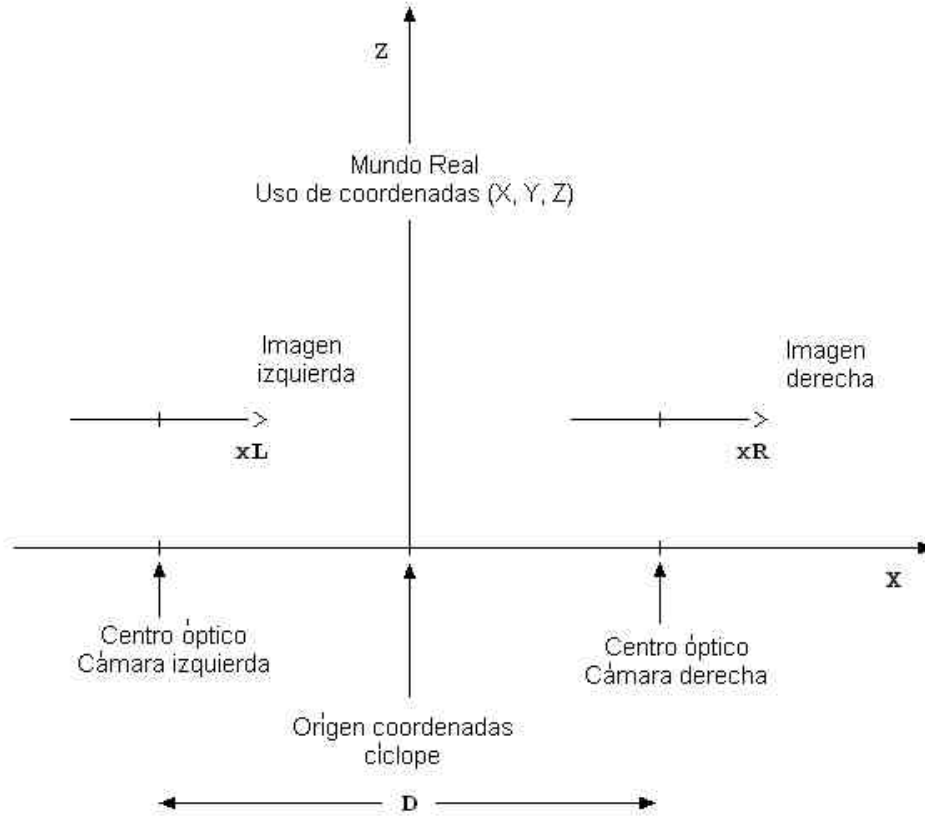


Fig. 3.20 Cámaras paralelas

Donde:

xL y xR son las posiciones x en la imagen izquierda y derecha respectivamente.

D es la distancia de separación entre las cámaras.

Por lo que un punto X, Y, Z , puede ser determinado mediante dos imágenes usando las ecuaciones que a continuación de muestran:

$$X = \frac{D(xL - xR)}{2(xL + xR)}$$

$$Y = \frac{(D)(y)}{xL + xR}$$

$$Z = \frac{(D)(f)}{xL + xR}$$

Ec. 3.14 Ecuaciones para un punto en el espacio

La cantidad $x_L - x_R$ es llamada disparidad estéreo o simplemente disparidad y es la diferencia horizontal de posiciones de un punto en una imagen.

Z es la posición relativa del objeto respecto a la cámara, la disparidad es inversamente proporcional a la profundidad, es decir: pequeñas disparidades conducen a largas profundidades, mientras que un cambio grande en la disparidad corresponde a un objeto cercano.

Nota: El desglose completo de la obtención de las coordenadas tridimensionales a partir de 2 puntos se encuentra en el Apéndice A..

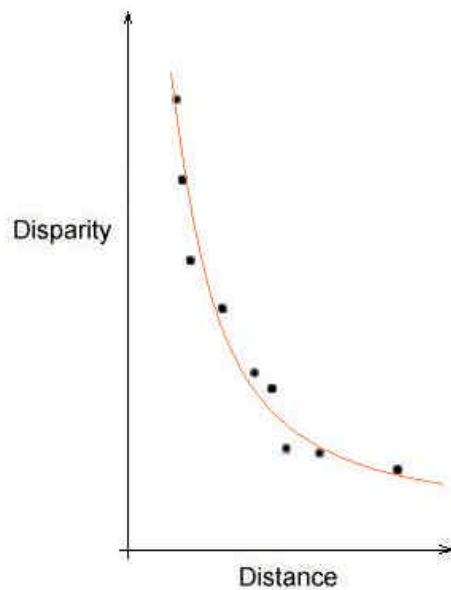


Fig. 3.21 Ejemplo de Graficacion Distancia vs Disparidad

[Sussex '94]