

COPIA PRIVADA CON FINES DOCENTES

VISIÓN POR COMPUTADOR

ARTURO DE LA ESCALERA HUESO



6. Extracción de características.

Una vez que la imagen ha sido procesada se está en disposición de detectar los objetos presentes en ella. Para ello se buscan en la imagen aquellas características que definen al objeto que se desea encontrar. Algunas ya han sido vistas como el nivel de gris o el color; ahora veremos algoritmos para la detección de bordes, líneas, esquinas, texturas y movimiento.

6.1. Detección de bordes.

Una de las informaciones más útiles que se encuentran en una imagen la constituyen los bordes ya que al delimitar los objetos definen los límites entre ellos y el fondo y entre los objetos entre sí. La información extraída por los detectores de bordes puede ser utilizada posteriormente en la etapa de segmentación. Las técnicas usadas en la detección de bordes tienen por objeto la localización de los puntos en que se produce una variación de intensidad -dónde se pasa del nivel de gris del fondo al del objeto- (figura 6.1), empleándose para ello métodos basados en los operadores derivada. Básicamente se tienen dos posibilidades: aplicar la primera (gradiente) o la segunda derivada (laplaciana). En el primer caso se buscarán grandes picos y en la segunda pasos de respuesta positiva a negativa o viceversa (paso por cero o *zero-crossing* en la literatura anglosajona). Como puede verse en la figura 6.1 se supone que el nivel de gris en el interior de los objetos es constante y, como ya se vio cuando se estudió la eliminación de ruido, es una situación que nunca se dará. Es por ello que después de la fase de detección de bordes viene otra que distingue lo que son los bordes del objeto de lo que es ruido. Para ello se supondrá que la respuesta de este último es mucho menor en amplitud que la de los bordes de los objetos por lo que se definirá un umbral mínimo para considerar un pixel como perteneciente a un borde. Ello llevará a que en el caso de los métodos basados en el gradiente la respuesta sea bordes de anchura mayor de un pixel, lo que introduce una incertidumbre en la verdadera localización de los bordes. Este inconveniente no se encuentra en los métodos que se basan en la segunda derivada ya que los bordes vienen definidos por el paso por cero. Sin embargo encontrar ese paso por cero no es algo inmediato por lo que los algoritmos serán más lentos que los que utilizan únicamente la primera derivada. Dependerá por tanto el que se necesite precisión o exactitud el que se elija un detector u otro.

6.1.1 Técnicas basadas en el gradiente.

Para poder utilizar operadores de este tipo sobre una imagen muestreada es necesario obtener una aproximación del concepto de derivada para espacios discretos. La generalización comúnmente usada se basa en el cálculo de diferencias entre pixeles vecinos. Estas diferencias, según la relación entre los pixeles considerados, pueden dar lugar a derivadas unidimensionales o bidimensionales, así como aplicarse en alguna dirección determinada de la imagen o en todas las direcciones de forma global. Existen otras aproximaciones diferenciales que han probado ser de gran utilidad para la extracción de contornos, es el caso de la de Roberts, la de Sobel, que se verán después.

Se define el operador gradiente G aplicado sobre una imagen $f(x,y)$ como:

$$\nabla f(x,y) = \begin{bmatrix} G_x & G_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} & \frac{\partial f}{\partial y} \end{bmatrix}$$

El vector gradiente representa la variación máxima de intensidad para el punto (x,y). Es por ello interesante conocer su módulo y dirección que vendrán dados por:

$$|\nabla f| = \sqrt{G_x^2 + G_y^2}$$

$$\angle \nabla f = \arctan\left(\frac{G_y}{G_x}\right)$$

siendo la dirección del gradiente perpendicular al borde.

Debido al coste computacional el módulo a veces se simplifica por:

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

Como las imágenes digitales no son señales continuas sino discretas el gradiente se aproxima por:

$$\nabla f(x,y) = \begin{bmatrix} G_x & G_y \end{bmatrix} = \begin{bmatrix} \frac{\Delta f}{\Delta x} & \frac{\Delta f}{\Delta y} \end{bmatrix}$$

que se puede representar por las máscaras:

- $G_x = \frac{\Delta f}{\Delta x}$

$$\begin{bmatrix} -1 & 1 \end{bmatrix} * f(x,y)$$

- $G_y = \frac{\Delta f}{\Delta y}$

$$\begin{bmatrix} -1 \\ 1 \end{bmatrix} * f(x,y)$$

Sin embargo estas máscaras no suelen utilizarse debido a que son muy sensibles al ruido al tener en cuenta solamente la información de dos píxeles. Con el tiempo han aparecido otros filtros que además de calcular el gradiente tengan cierto efecto de suavizado, o lo que es lo mismo, sean menos sensibles al ruido.

Operador de Roberts.

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

Operador de Prewitt.

Expande la definición de gradiente a un entorno de 3x3 para ser más inmune al ruido.

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

Operador de Sobel

Sobel da más importancia a los píxeles centrales que el operador de Prewitt.

$$\begin{bmatrix} -1 & 0 & 1 \\ -1 & -2 & -1 \end{bmatrix} \quad \begin{bmatrix} -1 & -2 & -1 \end{bmatrix}$$

-2	0	2
-1	0	1

0	0	0
1	2	1

Operador Isotrópico (u operador de Frei-Chen).

Mientras que Prewitt detecta mejor los bordes verticales, Sobel lo hace en los diagonales. El operador isotrópico intenta llegar a un equilibrio entre ellos.

-1	0	1
$-\sqrt{2}$	0	$\sqrt{2}$
-1	0	1

-1	$-\sqrt{2}$	-1
0	0	0
1	$\sqrt{2}$	1

En la figura 6.2 se muestra la aplicación de estos operadores sobre la misma imagen. Quitando el operador de Roberts, claramente distinto a los otros tres, no se puede apreciar grandes diferencias en el resultado que proporcionan Prewitt, Sobel o Frei-Chen. Así, en la figura 6.3 se han tomado como puntos de borde aquellos pixeles superiores a cierto valor, siendo el de Roberts mucho menor que el de los otros tres. Aunque ahora sí que pueden verse resultados distintos aunque las diferencias no son muy grandes. Meramente por cuestión de costumbre el operador de Sobel es el que más se utiliza.

6.1.2 Operadores de segundo orden.

La laplaciana es la segunda derivada de una función y representa la derivada de esta respecto a todas las direcciones:

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

Las dos definiciones que de forma más frecuente se utilizan para el operador Laplaciana son:

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

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

donde el pixel central toma el valor negativo de la suma de todos los que le rodean.

Como puede verse en la figura 6.4, este operador es muy sensible al ruido por lo que nunca se utiliza. Sin embargo si que es un paso intermedio para el operador de Marr-Hildreth que se considera a continuación.

6.1.3. Métodos basados en las derivadas de gaussianas.

El operador de Marr-Hildreth.

Según el desarrollo propuesto por Marr y Hildreth, el filtro que se utilice debe de cumplir dos condiciones.

1. Como se deben detectar los bordes, el filtro debe ser local y tomar solamente en cuenta la información de puntos cercanos al que se quiere analizar.

2. Desde el punto de vista frecuencial, debe de reducir el número de frecuencias que analiza, debe ser por tanto paso-banda.

Estas condiciones son contradictorias y el mejor compromiso es tomar la laplaciana de una gaussiana.

Si se convoluciona la imagen con una gaussiana de expresión:

$$G(x, y) = e^{-\frac{(x+y)^2}{2\sigma^2}}$$

se disminuye la influencia del ruido.

Si después se realiza la segunda deriva se obtiene el operador de Marr-Hildreth, también conocido como el operador laplaciana de una gaussiana o LoG:

$$H(x, y) = \nabla^2 (G(x, y) * I(x, y))$$

El operador así expresado primero haría la convolución de la imagen con la gaussiana para después obtener la laplaciana. Esto puede acelerarse ya que por propiedades de la gaussiana es lo mismo que:

$$H(x, y) = (\nabla^2 G(x, y)) * I(x, y)$$

siendo:

$$\begin{aligned} \nabla^2 &= \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} \\ \frac{\partial G}{\partial x} &= -\frac{x}{\sigma^2} e^{-\frac{r^2}{2\sigma^2}} \\ \frac{\partial^2 G}{\partial x^2} &= -\frac{1}{\sigma^2} e^{-\frac{r^2}{2\sigma^2}} + \frac{x^2}{\sigma^4} e^{-\frac{r^2}{2\sigma^2}} = \frac{x^2 - \sigma^2}{\sigma^4} e^{-\frac{r^2}{2\sigma^2}} \end{aligned}$$

Luego:

$$\begin{aligned} \nabla^2 G(x, y) &= \frac{r^2 - 2\sigma^2}{\sigma^4} e^{-\frac{r^2}{2\sigma^2}} \\ r^2 &= x^2 + y^2 \end{aligned}$$

En la figura 6.5 se tiene la imagen original, la convolución con una gaussiana de desviaciones típicas 1.8, 2.8 y 3.8 píxeles y la detección de bordes. Se puede ver la imagen original, la convolución con las gaussianas, la laplaciana y los bordes tras obtener el paso por cero. Obviamente cuanto más estrecha sea la gaussiana más bordes se obtendrán. De ello se desprende que si se tiene un juego de convoluciones se pueda ir de los bordes más importante de la imagen a los bordes que definen los objetos más pequeños o lo detalles de los objetos. Este operador tiene los inconvenientes de encontrar más bordes de los necesarios si la gaussiana es estrecha (efecto *plato de espagueti*) y de redondear las esquinas.

De la figura 6.6 puede deducirse que la diferencia entre dos gaussianas (DoG) tiene una forma aproximada a la laplaciana. Por ello otra forma de obtener los bordes es convolucionar la imagen con dos gaussianas distintas, obtener su diferencia y buscar los puntos de paso por cero (figura 6.7).

Detector de Canny.

El detector de bordes de Canny se obtiene a partir de la optimización de una serie de condiciones:

- Error: Se deben detectar todos y solo los bordes.

- Localización. La distancia entre el pixel señalado como borde y el borde real debe de ser tan pequeña como se pueda.
 - Respuesta. No debe identificar varios pixeles como bordes cuando sólo exista uno.
- Estas tres condiciones pueden ser expresadas de forma matemática como:

$$SNR = \frac{A \left| \int_{-W}^0 f'(x) dx \right|}{n_0 \sqrt{\int_{-W}^W f^2(x) dx}}$$

$$\text{Localización} = \frac{A |f'(0)|}{n_0 \sqrt{\int_{-W}^W f^2(x) dx}}$$

$$\text{Distancia} = \pi \left(\frac{\int_{-\infty}^{\infty} f^2(x) dx}{\int_{-\infty}^{\infty} f'^2(x) dx} \right)^{\frac{1}{2}}$$

Canny busca la optimización del producto de la relación señal ruido por la localización y el tercero como una condición. Con ello se llega a que el operador óptimo es la derivada de una gaussiana.

Para obtener los bordes se siguen los siguientes pasos:

- Se tiene la imagen I.
- Se tiene una gaussiana unidimensional G
- Se obtienen las derivadas (unidimensionales) de la gaussiana Gx y Gy
- Se convolucionan las derivadas de G con la imagen obteniendo Ix Iy (figura 6.8a) (figura 6.8b)
- Se obtiene la magnitud M como la raíz cuadrada de la suma de los cuadrados de Ix Iy (figura 6.8c).
- El siguiente paso es la supresión de aquellos pixeles que no sean máximos. Para ello se compara el valor de la magnitud para cada pixel con sus vecinos. Solo aquellos que son máximos en su entorno se dejan como están (figura 6.8d).
- Para detectar los bordes se utilizarán dos umbrales T1 y T2, siendo este último el mayor. Si se usase el menor (figura 6.8e) se detectarían bordes poco importantes, si se usase solo el mayor (figura 6.8f) se dejarían de detectar puntos pertenecientes a los bordes. Por ello la condición para pertenecer a un borde es ser mayor de T2 o mayor que T1 siempre que uno de sus vecinos sea mayor que T2 (figura 6.8g).

6.1.4. Operadores de ajuste al modelo.

Los operadores de ajuste al modelo permiten realizar una medida del gradiente en las direcciones seleccionadas. Se puede considerar las siguientes direcciones y sus máscaras asociadas:

Prewitt

Kirsch

Robinson

Robinson

Este	<table border="1"><tr><td>1</td><td>1</td><td>-1</td></tr><tr><td>1</td><td>-2</td><td>-1</td></tr><tr><td>1</td><td>1</td><td>-1</td></tr></table>	1	1	-1	1	-2	-1	1	1	-1	<table border="1"><tr><td>5</td><td>-3</td><td>-3</td></tr><tr><td>5</td><td>0</td><td>-3</td></tr><tr><td>5</td><td>-3</td><td>-3</td></tr></table>	5	-3	-3	5	0	-3	5	-3	-3	<table border="1"><tr><td>1</td><td>0</td><td>-1</td></tr><tr><td>1</td><td>0</td><td>-1</td></tr><tr><td>1</td><td>0</td><td>-1</td></tr></table>	1	0	-1	1	0	-1	1	0	-1	<table border="1"><tr><td>1</td><td>0</td><td>-1</td></tr><tr><td>2</td><td>0</td><td>-2</td></tr><tr><td>1</td><td>0</td><td>-1</td></tr></table>	1	0	-1	2	0	-2	1	0	-1
1	1	-1																																						
1	-2	-1																																						
1	1	-1																																						
5	-3	-3																																						
5	0	-3																																						
5	-3	-3																																						
1	0	-1																																						
1	0	-1																																						
1	0	-1																																						
1	0	-1																																						
2	0	-2																																						
1	0	-1																																						
Noreste	<table border="1"><tr><td>1</td><td>-1</td><td>-1</td></tr><tr><td>1</td><td>-2</td><td>-1</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table>	1	-1	-1	1	-2	-1	1	1	1	<table border="1"><tr><td>-3</td><td>-3</td><td>-3</td></tr><tr><td>5</td><td>0</td><td>-3</td></tr><tr><td>5</td><td>5</td><td>-3</td></tr></table>	-3	-3	-3	5	0	-3	5	5	-3	<table border="1"><tr><td>0</td><td>-1</td><td>-1</td></tr><tr><td>1</td><td>0</td><td>-1</td></tr><tr><td>1</td><td>1</td><td>0</td></tr></table>	0	-1	-1	1	0	-1	1	1	0	<table border="1"><tr><td>0</td><td>-1</td><td>-2</td></tr><tr><td>1</td><td>0</td><td>-1</td></tr><tr><td>2</td><td>1</td><td>0</td></tr></table>	0	-1	-2	1	0	-1	2	1	0
1	-1	-1																																						
1	-2	-1																																						
1	1	1																																						
-3	-3	-3																																						
5	0	-3																																						
5	5	-3																																						
0	-1	-1																																						
1	0	-1																																						
1	1	0																																						
0	-1	-2																																						
1	0	-1																																						
2	1	0																																						
Norte	<table border="1"><tr><td>-1</td><td>-1</td><td>-1</td></tr><tr><td>1</td><td>-2</td><td>1</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table>	-1	-1	-1	1	-2	1	1	1	1	<table border="1"><tr><td>-3</td><td>-3</td><td>-3</td></tr><tr><td>-3</td><td>0</td><td>-3</td></tr><tr><td>5</td><td>5</td><td>5</td></tr></table>	-3	-3	-3	-3	0	-3	5	5	5	<table border="1"><tr><td>-1</td><td>-1</td><td>-1</td></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table>	-1	-1	-1	0	0	0	1	1	1	<table border="1"><tr><td>1</td><td>-2</td><td>-1</td></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>1</td><td>2</td><td>1</td></tr></table>	1	-2	-1	0	0	0	1	2	1
-1	-1	-1																																						
1	-2	1																																						
1	1	1																																						
-3	-3	-3																																						
-3	0	-3																																						
5	5	5																																						
-1	-1	-1																																						
0	0	0																																						
1	1	1																																						
1	-2	-1																																						
0	0	0																																						
1	2	1																																						
Noroeste	<table border="1"><tr><td>-1</td><td>-1</td><td>1</td></tr><tr><td>-1</td><td>-2</td><td>1</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table>	-1	-1	1	-1	-2	1	1	1	1	<table border="1"><tr><td>-3</td><td>-3</td><td>-3</td></tr><tr><td>-3</td><td>0</td><td>5</td></tr><tr><td>-3</td><td>5</td><td>5</td></tr></table>	-3	-3	-3	-3	0	5	-3	5	5	<table border="1"><tr><td>-1</td><td>-1</td><td>0</td></tr><tr><td>-1</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>1</td></tr></table>	-1	-1	0	-1	0	1	0	1	1	<table border="1"><tr><td>-2</td><td>-1</td><td>0</td></tr><tr><td>-1</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>2</td></tr></table>	-2	-1	0	-1	0	1	0	1	2
-1	-1	1																																						
-1	-2	1																																						
1	1	1																																						
-3	-3	-3																																						
-3	0	5																																						
-3	5	5																																						
-1	-1	0																																						
-1	0	1																																						
0	1	1																																						
-2	-1	0																																						
-1	0	1																																						
0	1	2																																						
Oeste	<table border="1"><tr><td>-1</td><td>1</td><td>1</td></tr><tr><td>-1</td><td>-2</td><td>1</td></tr><tr><td>-1</td><td>1</td><td>1</td></tr></table>	-1	1	1	-1	-2	1	-1	1	1	<table border="1"><tr><td>-3</td><td>-3</td><td>5</td></tr><tr><td>-3</td><td>0</td><td>5</td></tr><tr><td>-3</td><td>-3</td><td>5</td></tr></table>	-3	-3	5	-3	0	5	-3	-3	5	<table border="1"><tr><td>-1</td><td>0</td><td>1</td></tr><tr><td>-1</td><td>0</td><td>1</td></tr><tr><td>-1</td><td>0</td><td>1</td></tr></table>	-1	0	1	-1	0	1	-1	0	1	<table border="1"><tr><td>-1</td><td>0</td><td>1</td></tr><tr><td>-2</td><td>0</td><td>2</td></tr><tr><td>-1</td><td>0</td><td>1</td></tr></table>	-1	0	1	-2	0	2	-1	0	1
-1	1	1																																						
-1	-2	1																																						
-1	1	1																																						
-3	-3	5																																						
-3	0	5																																						
-3	-3	5																																						
-1	0	1																																						
-1	0	1																																						
-1	0	1																																						
-1	0	1																																						
-2	0	2																																						
-1	0	1																																						
Sudoeste	<table border="1"><tr><td>1</td><td>1</td><td>1</td></tr><tr><td>-1</td><td>-2</td><td>1</td></tr><tr><td>-1</td><td>-1</td><td>1</td></tr></table>	1	1	1	-1	-2	1	-1	-1	1	<table border="1"><tr><td>-3</td><td>5</td><td>5</td></tr><tr><td>-3</td><td>0</td><td>5</td></tr><tr><td>-3</td><td>-3</td><td>-3</td></tr></table>	-3	5	5	-3	0	5	-3	-3	-3	<table border="1"><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>-1</td><td>0</td><td>1</td></tr><tr><td>-1</td><td>-1</td><td>0</td></tr></table>	0	1	1	-1	0	1	-1	-1	0	<table border="1"><tr><td>0</td><td>1</td><td>2</td></tr><tr><td>-1</td><td>0</td><td>1</td></tr><tr><td>-2</td><td>-1</td><td>0</td></tr></table>	0	1	2	-1	0	1	-2	-1	0
1	1	1																																						
-1	-2	1																																						
-1	-1	1																																						
-3	5	5																																						
-3	0	5																																						
-3	-3	-3																																						
0	1	1																																						
-1	0	1																																						
-1	-1	0																																						
0	1	2																																						
-1	0	1																																						
-2	-1	0																																						
Sur	<table border="1"><tr><td>1</td><td>1</td><td>1</td></tr><tr><td>1</td><td>-2</td><td>1</td></tr><tr><td>-1</td><td>-1</td><td>-1</td></tr></table>	1	1	1	1	-2	1	-1	-1	-1	<table border="1"><tr><td>5</td><td>5</td><td>5</td></tr><tr><td>-3</td><td>0</td><td>-3</td></tr><tr><td>-3</td><td>-3</td><td>-3</td></tr></table>	5	5	5	-3	0	-3	-3	-3	-3	<table border="1"><tr><td>1</td><td>1</td><td>1</td></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>-1</td><td>-1</td><td>-1</td></tr></table>	1	1	1	0	0	0	-1	-1	-1	<table border="1"><tr><td>1</td><td>2</td><td>1</td></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>-1</td><td>-2</td><td>-1</td></tr></table>	1	2	1	0	0	0	-1	-2	-1
1	1	1																																						
1	-2	1																																						
-1	-1	-1																																						
5	5	5																																						
-3	0	-3																																						
-3	-3	-3																																						
1	1	1																																						
0	0	0																																						
-1	-1	-1																																						
1	2	1																																						
0	0	0																																						
-1	-2	-1																																						
Sudeste	<table border="1"><tr><td>1</td><td>1</td><td>1</td></tr><tr><td>1</td><td>-2</td><td>-1</td></tr><tr><td>1</td><td>-1</td><td>-1</td></tr></table>	1	1	1	1	-2	-1	1	-1	-1	<table border="1"><tr><td>5</td><td>5</td><td>-3</td></tr><tr><td>5</td><td>0</td><td>-3</td></tr><tr><td>-3</td><td>-3</td><td>-3</td></tr></table>	5	5	-3	5	0	-3	-3	-3	-3	<table border="1"><tr><td>1</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>-1</td></tr><tr><td>0</td><td>1</td><td>-1</td></tr></table>	1	1	0	1	0	-1	0	1	-1	<table border="1"><tr><td>2</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>-1</td></tr><tr><td>0</td><td>-1</td><td>-2</td></tr></table>	2	1	0	1	0	-1	0	-1	-2
1	1	1																																						
1	-2	-1																																						
1	-1	-1																																						
5	5	-3																																						
5	0	-3																																						
-3	-3	-3																																						
1	1	0																																						
1	0	-1																																						
0	1	-1																																						
2	1	0																																						
1	0	-1																																						
0	-1	-2																																						

Se ha de tener en cuenta que de las ocho máscaras anteriores sólo cuatro son linealmente independientes. Ello conduce a la posibilidad de definir cuatro matrices de 3x3 que sean mutuamente ortogonales y medir el espacio cubierto por ellas

6.1.5 Detección de líneas.

Las líneas se pueden considerar como una extensión dentro de lo visto anteriormente para los bordes. Así se tiene un juego de cuatro filtros para detectar líneas en diferentes direcciones.

E-O	NE-NO	N-S	No-SE																																				
<table border="1"><tr><td>-1</td><td>-1</td><td>-1</td></tr><tr><td>2</td><td>2</td><td>2</td></tr><tr><td>-1</td><td>-1</td><td>-1</td></tr></table>	-1	-1	-1	2	2	2	-1	-1	-1	<table border="1"><tr><td>-1</td><td>-1</td><td>2</td></tr><tr><td>-1</td><td>2</td><td>-1</td></tr><tr><td>2</td><td>-1</td><td>-1</td></tr></table>	-1	-1	2	-1	2	-1	2	-1	-1	<table border="1"><tr><td>-1</td><td>2</td><td>-1</td></tr><tr><td>-1</td><td>2</td><td>-1</td></tr><tr><td>-1</td><td>2</td><td>-1</td></tr></table>	-1	2	-1	-1	2	-1	-1	2	-1	<table border="1"><tr><td>2</td><td>-1</td><td>-1</td></tr><tr><td>-1</td><td>2</td><td>-1</td></tr><tr><td>-1</td><td>-1</td><td>2</td></tr></table>	2	-1	-1	-1	2	-1	-1	-1	2
-1	-1	-1																																					
2	2	2																																					
-1	-1	-1																																					
-1	-1	2																																					
-1	2	-1																																					
2	-1	-1																																					
-1	2	-1																																					
-1	2	-1																																					
-1	2	-1																																					
2	-1	-1																																					
-1	2	-1																																					
-1	-1	2																																					

La primera máscara proporciona las líneas de orientación horizontal. Para el caso de tener un fondo constante, se obtiene el máximo valor de respuesta cuando la línea pase por la fila central de la máscara. Lo mismo ocurrirá para el caso de la segunda máscara de cara a la detección de líneas que formen 45°, en la tercera para líneas verticales y finalmente la última para detección de líneas que formen -45° respecto a la referencia. Algunos resultados pueden verse en la figura 6.9. En ella se compara los resultados si se hubiera aplicado a la imagen un detector de bordes. Puede verse como el detector de bordes resalta un mayor número de líneas y que en el caso de coincidir, da una respuesta doble.

6.1.6 Detección de esquinas.

Las esquinas presentes en una imagen pueden ser debidas a dos razones:

- Corresponden a la intersección de dos bordes de un objeto.
- Representan una discontinuidad en los niveles de gris debido a la textura de los objetos.

Una manera de detectarlos es encontrar primero los bordes de los objetos y buscar más adelante puntos en los que la curvatura del borde varíe rápidamente. Aquí se mostraran dos métodos que trabajan directamente con los niveles de gris de la imagen.

El método propuesto por Tomasi y Kanade consiste en determinar qué zonas de la imagen tienen una respuesta elevada al gradiente horizontal y vertical al mismo tiempo. Para ello se analiza en el entorno de un punto la siguiente expresión:

$$C = \begin{bmatrix} \sum E_x^2 & \sum E_x E_y \\ \sum E_x E_y & \sum E_y^2 \end{bmatrix}$$

Si de la matriz anterior se obtienen los autovalores y autovectores se puede deducir cómo es la distribución de los niveles de gris dentro del entorno del punto. Toda matriz simétrica (y la anterior lo es) puede diagonalizarse mediante una rotación de los ejes de coordenadas. Así se podría diagonalizar la matriz anterior a:

$$C = \begin{bmatrix} C_1 & 0 \\ 0 & C_2 \end{bmatrix}$$

siendo C1 y C2 los autovalores. Si los niveles de gris de la imagen fueran constantes C1 y C2 valdrían cero. Si se tuviese un escalón ideal uno de los dos valores sería cero y el otro indicaría el salto de nivel de gris que hay presente en la imagen. El autovector correspondiente a este último valor tendría la misma dirección que el gradiente. En resumen los autovectores indican la dirección del gradiente y los autovalores el valor alto o bajo de este. Para detectar las esquinas habrá por tanto que obtener C1 y C2 y comprobar que el menor de ellos está por encima de un umbral prefijado.

Kitchen y Rosenfeld también se basan en la variación de la dirección de los gradientes. La fórmula que utilizan es el producto de la curvatura horizontal por el gradiente de la imagen:

$$C = \frac{I_{xx}I_y^2 - 2I_{xy}I_xI_y + I_{yy}I_x^2}{I_x^2 + I_y^2}$$

Resultados de ambos métodos pueden verse en la figura 6.10. Los resultados no son idénticos ya que dependen de varios factores: el tamaño del entorno que se toma en el primer método y el umbral que se fija en ambos para determinar qué se considera esquina.

6.2. Texturas

Al observar cierto tipo de materiales (figura 6.11) se nota que presentan un aspecto homogéneo a pesar de que no tienen un nivel de gris constante. Así se puede definir las texturas como patrones visuales homogéneos que se observan en cierto tipo de materiales (telas, piedras, madera). La importancia de su estudio radica en que forma parte de las características que definen los objetos y que al observar su deformación se puede obtener su forma tridimensional. Hay que hacer notar que la textura depende del grado de resolución de la imagen. Lo que para una distancia aparece como patrones repetitivos desaparece si la imagen está tomada mucho más cerca, apareciendo tal vez otro tipo de textura distinto.

A la hora de estudiar la textura se pueden seguir dos aproximaciones: estadística y frecuencial. En el primer caso se analizan los valores estadísticos de primer orden o superior de los niveles de gris en una área o el número de picos, valles, etc. En el segundo caso se tiene que al ser la textura la repetición de un patrón esto debe manifestarse en la aparición de picos en la transformada de Fourier.

6.2.1. Análisis estadístico de texturas.

El análisis estadístico de texturas analiza la distribución de alguna propiedad para cada uno de los píxeles de la imagen. Dependiendo del número de puntos que definen la textura se clasifican en estadísticas de primer, segundo orden u órdenes superiores. A partir de este instante nos centraremos en el análisis de los niveles de gris de una imagen aunque algo similar se puede realizar para los bordes, picos, valles, u otra propiedad espacial.

Estadísticos de primer orden.

Ya se vio en un capítulo anterior lo que era el histograma de la imagen. Si a cada valor se le divide por el número total de píxeles se tiene que el nuevo histograma representa la probabilidad de que un nivel de gris determinado aparezca en la imagen. Las propiedades que se pueden obtener son (figura 6.12):

- Media.

$$\mu = \sum_{i=1}^n i h(i)$$

- Desviación típica.

$$\sigma^2 = \sum_{i=1}^n (i - \mu)^2 h(i)$$

- Tercer momento o coeficiente de asimetría.

$$\mu_3 = \frac{1}{\sigma^3} \sum_{i=1}^n (i - \mu)^3 h(i)$$

- Apuntamiento o curtosis:

$$\mu_4 = \frac{1}{\sigma^4} \sum_{i=1}^n (i - \mu)^4 h(i)$$

- Entropía.

$$-\sum_{i=1}^n h(i) \log h(i)$$

Donde la media es la estimación del nivel de gris de la textura; la desviación típica indica la media de la dispersión respecto de ese valor. El tercer momento es la medida de la asimetría del histograma. Así si el histograma siguiese perfectamente una distribución gaussiana (por lo tanto simétrico) su tercer momento sería cero. Según el histograma esté mas inclinado a un lado o a otro sus valores serán negativos o positivos. La curtosis o apuntamiento indica como se reparte el histograma entre la parte central y los extremos. Cuanto mayor sea este valor menos importancia tendrá la parte central. La entropía mide la uniformidad del histograma.

En la figura 6.13 se tiene la imagen con su histograma. Si se obtiene la media (6.14-a) puede verse en el nuevo histograma que tiene dos picos (6.14-a). Uno correspondiente al aeropuerto y otro al resto de la imagen (como separarlos se verá en mayor detalle cuando se hable de la segmentación). Si se toma un valor o umbral intermedio entre los dos picos, puede separarse el aeropuerto del resto de los objetos; los pixeles con un valor menor que el umbral son aeropuerto y los mayores no (6.14-c). Puede verse en la siguiente imagen (6.14-d) el resultado de haber aplicado el mismo umbral a la imagen original. Aunque se habrían clasificado bien todos los puntos del aeropuerto se habría clasificado erróneamente puntos pertenecientes al campo. Si se analiza la desviación típica (6.14-e) se tiene la siguiente imagen con su histograma (6.14-f) en donde se pueden separar la zona de edificios (6.14-g), del campo y el aeropuerto (6.14-h).

Otra manera de comparar dos histogramas es utilizar el test de Kolmogorov-Smirnov. Para ello se toma la distribución acumulativa del histograma:

$$H(k) = \sum_{i=1}^k h(k)$$

y la similitud viene dada por:

$$\max_k |H_1(k) - H_2(k)|$$

o por:

$$\sum_{i=1}^n |H_1(k) - H_2(k)|$$

donde $H_2(k)$ vendría dado por el histograma ideal de la textura.

Estadísticos de segundo orden.

Al estar basados en el histograma, los estadísticos de primer orden tiene el inconveniente de no captar toda la información espacial. Así se obtendría el mismo resultado para un tablero de ajedrez que para otro con los colores blanco y negro intercambiados. Es por ello que se desarrollaron los estadísticos de segundo orden.

Matrices de coocurrencia.

Dado un vector en coordenadas polares, $\delta = (r, \theta)$, se puede calcular la probabilidad condicional de que dos propiedades aparezcan separados por la distancia δ , P_δ . Como encontrar matrices de coocurrencia para cada δ es prohibitivo se suelen limitar a una serie de ángulos, $0^\circ, 45^\circ, 90^\circ$ y 135° , así como la distancia r al valor de 1 pixel.

Así, por ejemplo se tiene la imagen:

0	1	2	3	1
2	1	3	1	1
0	0	2	2	1
1	2	0	3	1
0	0	0	0	3

Si r es igual a 1 y δ a 0° , entonces $P_\delta(3,1)=3$ ya que se tienen las parejas:

0	1	2	3	1
2	1	3	1	1
0	0	2	2	1
1	2	0	3	1
0	0	0	0	3

Por lo que $P_\delta(i,j)$ es:

4	0	1	0
1	1	2	3
1	2	1	0
2	1	1	0

Algunas propiedades que pueden extraerse a partir de dicha matriz son:

- Energía o segundo momento angular.

$$\sum_{i=1}^n \sum_{j=1}^n P_\delta(i,j)^2$$

- Varianza.

$$\sum_{i=1}^n \sum_{j=1}^n (i - \mu)^2 P_\delta(i,j)^2$$

- Contraste.

$$\sum_{k=0}^{n-1} k^2 \sum_{i=1}^n \sum_{\substack{j=1 \\ |i-j|=k}}^n P_\delta(i,j)$$

- Entropía:

$$-\sum_{i=1}^n \sum_{j=1}^n P_\delta(i,j) \log P_\delta(i,j)$$

- Correlación.

$$\frac{\sum_{i=1}^n \sum_{j=1}^n i j P_\delta(i,j) - \mu_x \mu_y}{\sigma_x \sigma_y}$$

donde:

$$\mu_x = \sum_{i=1}^n i \sum_{j=1}^n P_\delta(i, j)$$

$$\mu_y = \sum_{j=1}^n j \sum_{i=1}^n P_\delta(i, j)$$

$$\sigma_x = \sum_{i=1}^n (i - \mu_x)^2 \sum_{j=1}^n P_\delta(i, j)$$

$$\sigma_y = \sum_{j=1}^n (j - \mu_y)^2 \sum_{i=1}^n P_\delta(i, j)$$

- Homogeneidad:

$$\frac{\sum_{j=1}^n \sum_{i=1}^n P_\delta(i, j)}{1 + |i - j|}$$

- Tendencia del agrupamiento (*cluster*):

$$\sum_{j=1}^n \sum_{i=1}^n (i + j - 2\mu)^k P_\delta(i, j)$$

- Máxima probabilidad:

$$\max_{i, j} P_\delta(i, j)$$

El segundo momento angular indica la homogeneidad de la imagen. El contraste la variación que existe dentro de ella y la correlación la linealidad de la imagen. La entropía indica la uniformidad de la matriz.

Estadísticos de las diferencias (*difference statistics*).

Viene dada por la distribución de probabilidad $P_\delta(k)$ de los valores pertenecientes a los píxeles intermedios entre los que están separados una distancia δ . Son una variación de lo anteriormente visto y viene definido por:

$$P_\delta(k) = \sum_{i=1}^n \sum_{\substack{j=1 \\ |i-j|=k}}^n P_\delta(i, j)$$

Las propiedades de esta distribución son:

- Segundo momento angular.

$$\sum_{k=0}^{n-1} P_\delta(k)^2$$

- Contraste.

$$\sum_{k=0}^{n-1} k^2 P_\delta(k)$$

- Entropía.

$$-\sum_{k=0}^{n-1} P_\delta(k) \log P_\delta(k)$$

- Media.

$$\sum_{k=0}^{n-1} k P_\delta(k)$$

6.2.2. Análisis frecuencial de texturas.

También de la transformada de Fourier (figura 6.15) se pueden analizar sus texturas. Si la transformada de Fourier de una imagen $f(x,y)$ es $F(u,v)$ su módulo será:

$$P(u, v) = |F(u, v)|^2$$

si se toma en coordenadas polares se podrá obtener dos distribuciones que son:

$$P(r) = 2 \sum_{\theta=0}^{\pi} P(r, \theta)$$

$$P(\theta) = \sum_{r=0}^{n/2} P(r, \theta)$$

El primero indicará el tamaño de la textura dominante mientras que el segundo su dirección (figura 6.16).

6.3. Detección de movimiento.

Otro tipo de característica es la debida al movimiento. Puede ayudar a la segmentación ya que los puntos del mismo objeto presentan unas velocidades semejantes. Los distintos tipos de movimiento que existen son:

- Movimiento de la cámara.
- Movimiento de los objetos.
- Cambios en la iluminación.
- Cambios en la estructura, forma o tamaño del objeto.

Se aplicarán las siguientes restricciones (aunque luego las dos primeras se eliminarán):

- No existen cambios de iluminación entre una imagen y otra.
- Respecto al movimiento se supondrá que la cámara está fija y que son los objetos los que se mueven.
- Todos los objetos son rígidos.

El método más sencillo es aquel que se basa en las diferencias de imágenes. Aunque estos algoritmos son menos precisos, son mucho más simples y por tanto pueden procesar la imagen en poco tiempo, por lo que su utilidad es mayor.

Los pasos a la hora de analizar el movimiento son tres:

- Determinar en que zonas de la imagen existe algún objeto que esté moviéndose. En esta primera etapa no es importante la exactitud, sólo detectar las futuras zonas de interés, en las que se realizará un análisis más detallado.
- En la segunda etapa la atención sólo está en aquellas zonas donde se haya detectado actividad y se extrae información que pueda ser usada para el reconocimiento de objetos, análisis del movimiento, etc.
- La última etapa tiene en cuenta el conocimiento previo del problema en concreto. Así por ejemplo la información obtenida de un objeto pequeño que se mueve lentamente cerca de la cámara de otro con igual forma pero mayor que se mueve rápidamente pero lejos de ella es la misma.

Ahora se tendrá no una imagen sino una secuencia representada por $f(x,y,t)$. Como se indicó el método para determinar el movimiento será la diferencia entre dos

imágenes. Luego la imagen que indicará el campo de velocidades entre los instantes i y j será:

$$V_{ij}(x, y) = \begin{cases} 1 & \text{si } |f(x, y, i) - f(x, y, j)| \geq T \\ 0 & \text{en cualquier otro caso} \end{cases}$$

El sentido de aplicar un valor de umbralización es para minimizar la influencia del ruido en ambas imágenes. Obviamente de igual manera movimientos pequeños no serán detectados. También deberá aplicarse un filtro ya que la influencia del ruido provoca pixeles aislados que hay que eliminar. Puede tomarse para ello la condición de considerar solamente aquellos pixeles que este conectados a sus cuatro u ocho vecinos. En la figura 6.17 se observan dos imágenes donde queda reflejado el movimiento de un coche y el de los arbustos. La imagen diferencia sufre una umbralización para después eliminar aquellos pixeles que no estén conectados a sus cuatro o a sus ocho vecinos.

Este método presenta la gran ventaja de su inmunidad a los cambios de iluminación ya que el tiempo transcurrido entre la toma de las dos imágenes suele ser muy pequeño. El gran inconveniente que presenta es que los cambios detectados son de dos tipos: la parte del fondo tapado por el objeto en la segunda imagen y la parte del fondo que estaba ocluida en la primera imagen y que aparece en la segunda. Puede así verse en la figura 6.17 cómo aparecen grupos de pixeles por parejas representando estas diferencias. No se tiene así información (con solo dos imágenes) de hacia donde se dirige el objeto. Si se tuviese una imagen en la que todos los objetos que apareciesen fueran estáticos se podría tomar como referencia y así estar seguros que todos los pixeles marcados como movimiento fueran del objeto que se mueve. Ahora el problema que surge son los cambios de iluminación que hacen que la imagen de referencia pierda validez.

Se puede seguir entonces un enfoque como el de la figura 6.18. Se toma una imagen base donde se esté seguro de que todos los objetos son estáticos. Cuando llega una nueva imagen se compara con la referencia tal como se vio en el caso anterior. Si la diferencia está por encima de un umbral es que existe movimiento. Si no el cambio es debido a la iluminación y ese valor sustituye al anterior de la imagen base. Resultados pueden verse en la imagen 6.19.

Los métodos presentados hasta ahora tienen como gran ventaja su simplicidad pero también sufren varios inconvenientes:

1. La cámara debe estar fija.
2. La imagen resultado indica en qué zonas existe el movimiento, pero no da ninguna información sobre la magnitud o dirección de la velocidad.

Existen dos grandes grupos de métodos para la obtención del campo de velocidades que tratan de solucionar estos dos problemas. En el primero se buscan una serie de características en una imagen (esquinas, bordes) y se encuentra su correspondencia en la segunda. La dificultad estriba en encontrar qué característica de la segunda imagen corresponde a una dada de la primera. Aunque son métodos rápidos no dan mucha información ya que el número de características elegidas suele ser bajo (si se compara con el número total de pixeles de la imagen).

El segundo grupo de métodos busca obtener un mapa denso del campo de velocidades (flujo óptico u optical flow en la literatura anglosajona). Para ello se tendrán en cuenta todos los pixeles de las dos imágenes. Como idea general se busca en el entorno a un punto cuál es la zona más parecida en la segunda imagen dependiendo de sus variaciones espaciales y temporales. Es un método local y sufre del denominado *problema de la apertura*. Así si se tiene el caso de la figura 6.20, rápidamente se observa que el objeto oscuro se está desplazando arriba a la derecha. Sin embargo este

sería un análisis global ya que observamos toda la imagen. El problema de la apertura se ve con claridad en las dos imágenes inferiores. Representan el mismo caso anterior pero ahora la única información es la que rodea a un punto. Solo se podrá concluir que el objeto se está desplazando hacia arriba. Si se podría conocer perfectamente el movimiento de la figura cuadrada si se observase a través del agujero una de las esquinas del mismo, tal como se aprecia en las dos últimas imágenes. Como conclusión no se puede calcular la velocidad verdadera de los puntos que constituyen la imagen, si solamente se tiene información en una pequeña vecindad en torno a un punto, ya que la única información obtenible directamente a partir de las vecindades de cada punto es la componente de la velocidad normal al contorno o borde que aparezca en la imagen (es decir sólo la componente en la dirección de los gradientes presentes en la imagen). Como caso extremo si la parte de la imagen que observamos hubiera estado situada justo en una zona donde no existieran gradientes de intensidad, el centro de cuadrado por ejemplo, no se podría determinar su velocidad. Se deben por tanto imponer restricciones adicionales, que de alguna manera lo que se está haciendo es introducir información adicional al planteamiento del problema.

Dado un punto P en el mundo se proyecta sobre un punto de la imagen $I(x,y)$ tomada en el tiempo t . Este mismo punto P se proyectará posteriormente, para un incremento de tiempo δt , en la posición $(x + \delta x, y + \delta y)$. Matemáticamente se expresa:

$$I(x, y, t) = I(x + \delta x, y + \delta y, t + \delta t)$$

La parte derecha de la ecuación puede ser desarrollada usando las series de Taylor en torno al punto (x,y) :

$$I(x, y, t) = I(x + \delta x, y + \delta y, t + \delta t) = I(x, y, t) + \frac{\partial I}{\partial x} \delta x + \frac{\partial I}{\partial y} \delta y + \frac{\partial I}{\partial t} \delta t + \varepsilon$$

Donde ε contiene términos de segundo orden y superior en δx , δy y δt . Restando $I(x,y,t)$ de ambos lados de la ecuación δt se obtiene:

$$\frac{\partial I}{\partial x} \delta x + \frac{\partial I}{\partial y} \delta y + \frac{\partial I}{\partial t} \delta t = \theta(\delta t)$$

dividiendo por δt y tomándolo el límite cuando $\delta t \rightarrow 0$ se obtiene la *ecuación de restricción del Flujo óptico* (figura 6.21):

$$I_x u + I_y v + I_t = 0$$

Donde:

- I_x es el gradiente de la imagen respecto al eje horizontal.
- I_y es el gradiente de la imagen respecto al eje vertical
- I_t es el gradiente temporal de las imágenes tomada en tiempos distintos.
- u es la componente horizontal de la velocidad.
- v es la componente vertical de la velocidad.

Puede observarse como se tienen dos incógnitas (u,v) y una sola ecuación por lo que se necesitarían dos puntos del mismo objeto para determinar las componentes de la velocidad. Sin embargo esto no es del todo cierto, como se explicó, debido al problema de la apertura ya que realmente la ecuación anterior indica la componente normal del campo de velocidades, es decir la componente de la velocidad en la dirección del gradiente ya que:

$$I_x u + I_y v + I_t = \begin{pmatrix} I_x & I_y \end{pmatrix} \begin{pmatrix} u \\ v \end{pmatrix} + I_t = 0$$

siendo la componente normal (figura 6.21):

$$V_n = \frac{-I_t}{\sqrt{I_x^2 + I_y^2}}$$

Se debe poner por tanto una nueva restricción, aquí se explicará la propuesta por Horn y Schunk: la suavidad del campo de velocidades (puntos cercanos pertenecientes al objeto poseerán velocidades similares). La estrategia que utilizan Horn y Schunk para obtener las fórmulas del algoritmo es la minimización de una función que contiene la condición de conservación y la condición de suavidad.

Se define el error de suavidad en el punto (i,j) como:

$$s(i, j) = \left(\frac{\partial u(i, j)}{\partial x} \right)^2 + \left(\frac{\partial u(i, j)}{\partial y} \right)^2 + \left(\frac{\partial v(i, j)}{\partial x} \right)^2 + \left(\frac{\partial v(i, j)}{\partial y} \right)^2$$

Además el error en la ecuación de restricción del flujo óptico es:

$$c(i, j) = (I_x(i, j)u(i, j) + I_y(i, j)v(i, j) + I_t(i, j))^2$$

Constituyendo finalmente el error total, que debe ser minimizado:

$$e = \sum_i \sum_j s(i, j) + \lambda c(i, j)$$

Donde λ es una constante que pondera el error en la ecuación del movimiento de la imagen relativo a la desviación de la condición de suavidad. El valor de este parámetro tiene una gran influencia sobre los resultados obtenidos del algoritmo. Si las imágenes son precisas y están libres de ruido el valor de λ será mayor que para el caso en el que la presencia de ruido es significativa.

Diferenciando el error e respecto de las variables u y v se obtiene:

$$\frac{\partial e}{\partial u} = 2(u - \bar{u}) + 2\lambda(I_x u + I_y v + I_t)I_x$$

$$\frac{\partial e}{\partial v} = 2(v - \bar{v}) + 2\lambda(I_x u + I_y v + I_t)I_y$$

donde \bar{u} y \bar{v} son las medias locales de u y v . Estas ecuaciones tendrán el mínimo cuando sus valores sean nulos, por lo que pueden ser reescritas de la forma:

$$(1 + \lambda I_x^2)u + \lambda I_x I_y v = \bar{u} - \lambda I_x I_t$$

$$(1 + \lambda I_y^2)v + \lambda I_x I_y u = \bar{v} - \lambda I_y I_t$$

Si se expresa este sistema de ecuaciones de forma lineal:

$$(A) \begin{pmatrix} u_{kl} \\ v_{kl} \end{pmatrix} = (B)$$

Donde el determinante de A puede ser escrito como :

$$|A| = 1 + \lambda(I_x^2 + I_y^2)$$

De tal manera que al final queda:

$$(1 + \lambda(I_x^2 + I_y^2))u = (1 + \lambda I_y^2)\bar{u} - \lambda I_x I_y \bar{v} - \lambda I_x I_t$$

$$(1 + \lambda(I_x^2 + I_y^2))v = (1 + \lambda I_x^2)\bar{v} - \lambda I_x I_y \bar{u} - \lambda I_y I_t$$

Resolviendo estas ecuaciones para u y v se obtienen, finalmente, las fórmulas recursivas que constituyen el algoritmo:

$$u^{n+1} = \bar{u}^n - \frac{I_x \bar{u}^n + I_y \bar{v}^n + I_t}{1 + \lambda(I_x^2 + I_y^2)} I_x$$

$$v^{n+1} = \bar{v}^n - \frac{I_x \bar{u}^n + I_y \bar{v}^n + I_t}{1 + \lambda(I_x^2 + I_y^2)} I_y$$

La interpretación geométrica puede verse en la figura 6.22. El valor del campo de velocidades en un punto es igual al valor medio del entorno que le rodea menos un ajuste que se produce en la dirección del gradiente de los niveles de gris. La obtención del flujo óptico para un caso real se presenta en la imagen 6.23.

6.4 Bibliografía.

- Argenti, F. Alparone L., Benelly G. *Fast algorithms for texture analysis using co-occurrence matrices*. IEEE Proceedings, Part F: Radar and Signal Processing, 137 (6): 443-448, 1990
- Borgefors, G. Distance transformations in digital images". *Computer Vision, Graphics and Image Processing*, vol 34, pp 344-371, 1986.
- Canny, J. A computational approach to edge detection. *IEEE Transaction on pattern analysis and Machine Intelligence*, vol PAMI- 8 (6), pp 676-698, 1986.
- Gotlieb, G.C. Kreyszig, H.E. *Texture descriptors based on co-occurrence matrices*. *Computer Vision Graphics and Image Processing* 51 (1): 70-86, 1990
- Haralick, R.; Statistical and Structural Approaches to Textures. *Proceedings of the IEEE*, vol 67, pp 786-804, may 1979.
- Horn B.K.P.; Schunk, B.G. Determining optical flow. *Artificial Intelligence*, vol 17, mayo 1981, pp 185-203
- Kitchen, L., Rosendfeld, A. Gray-level corner detection. *Pattern Recognition Letters*, 1 pp 95-102, 1982
- Marr, D. *Vision: a computational investigation into the human representation and processing of visual information*. 1982.
- Marr, D.; Hildreth, E. Theory of edge detection. *Proceedings of the Royal Society of London*. Series B, vol 207, pp 187-217, 1980.
- Prewitt, J., *Object enhancement and extraction*. *Picture processing and psychopictorics* (B. Lipkin and A. Rosendfeld Eds). New York, Academic Press, 1970.
- Roberts, L.G., "Machine Perception of Three-dimensional Solids". *Optical and Electro-optical information processing* (J. Tippett ed.) pp 159-197. Cambridge, MA, MIT Press, 1965.
- Robinson, G.S. edge detection by compass gradient masks. *Computer Graphics and Image Processing* 6 (5): 492-501, 1977.
- Schunk, B.G. The image flow constraint equation, *Computer vision, graphics and image processing* 35, 20-46, 1986
- Sobel, I.; *Camera Models and Machine Perception*. PhD Thesis, Stanford University, Stanford, CA, 1970.
- Tomasi, C. Kanade, T. Shape and motion from image streams under orthography: a factorization method, *International journal of computer Vision*, vol 9, pp137-154, 1992
- Tomita, F.; Tsuji, S.; *Computer analysis of visual textures*. Kluwer Academic Publishers, 1990

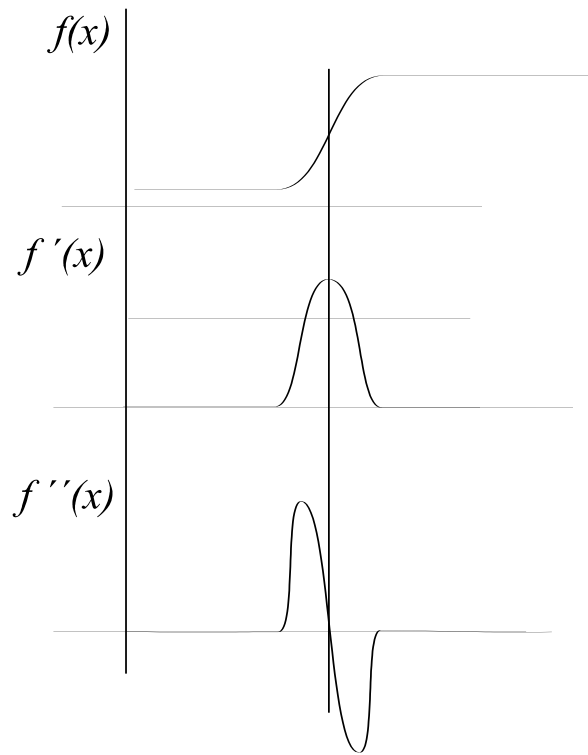


Figura 6.1. Detección de bordes mediante la primera y la segunda derivada.



Roberts



Prewitt



Sobel

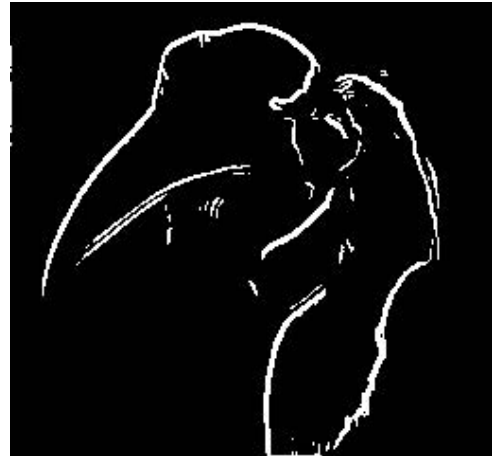


Isotrópico

Figura 6.2. Comparación entre los distintos detectores de bordes.



Roberts



Prewitt



Sobel



Isotrópico

Figura 6.3. Comparación entre los distintos detectores de bordes después de la umbralización.

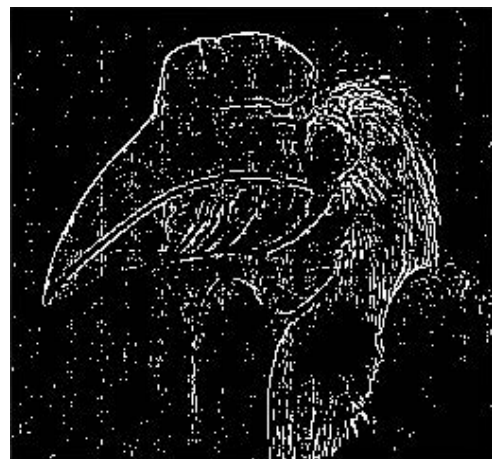


Figura 6.4. Detección de bordes mediante la laplaciana.

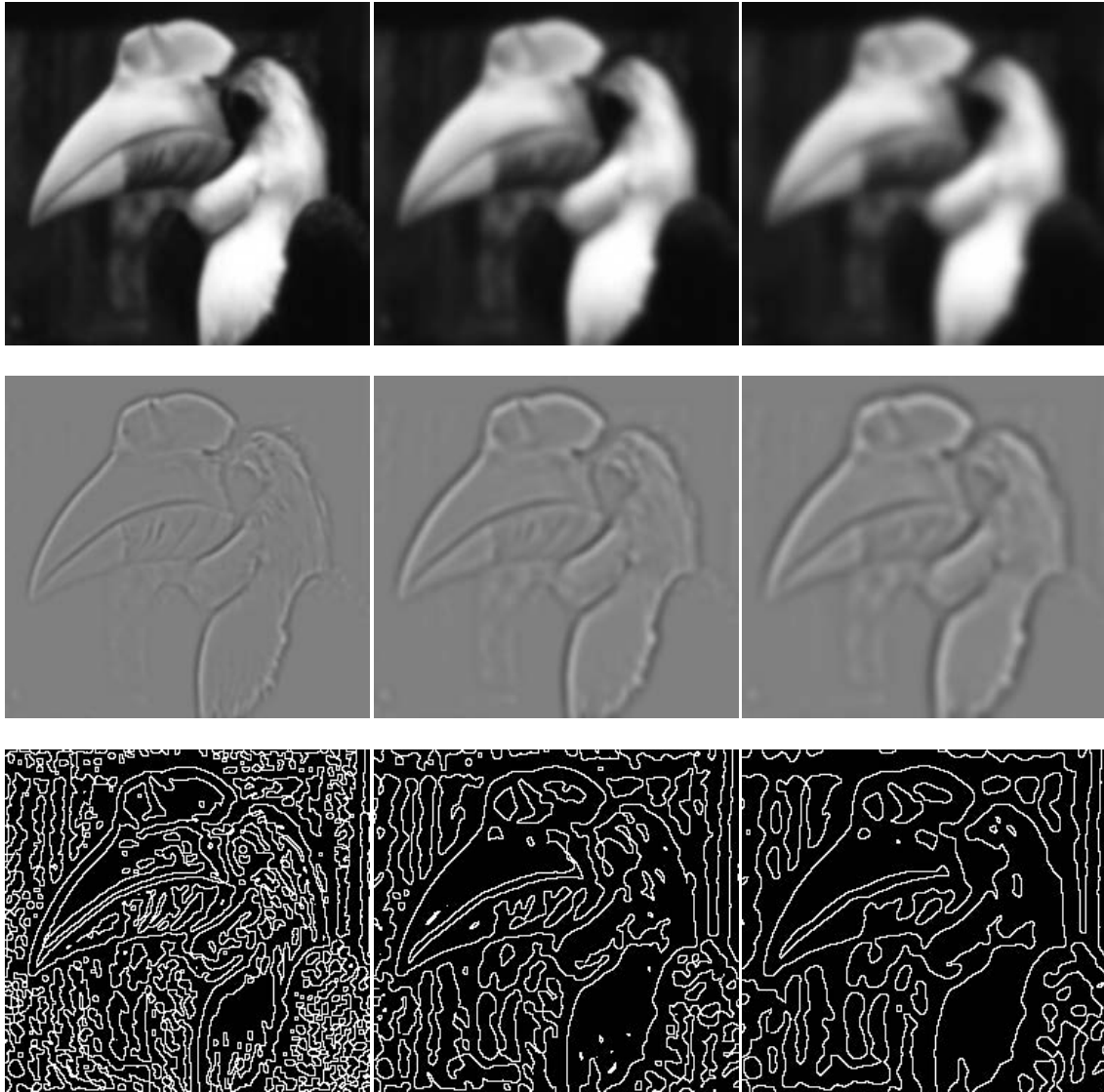


Figura 6.5 Detección de bordes según Marr

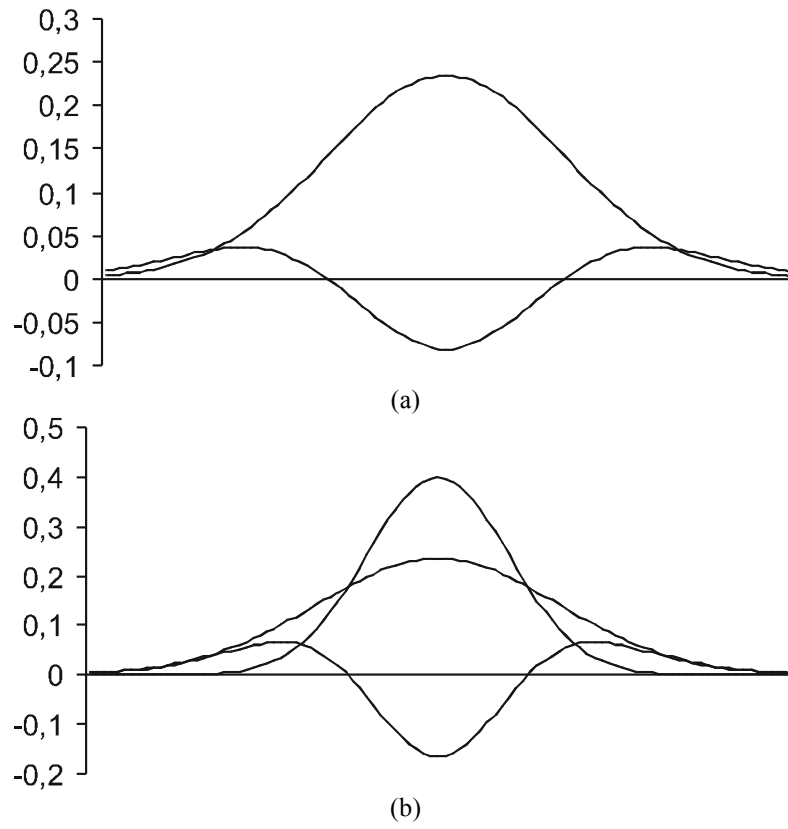


Figura 6.6. (a) Laplaciana de una gaussiana (b) Diferencia de gaussianas.

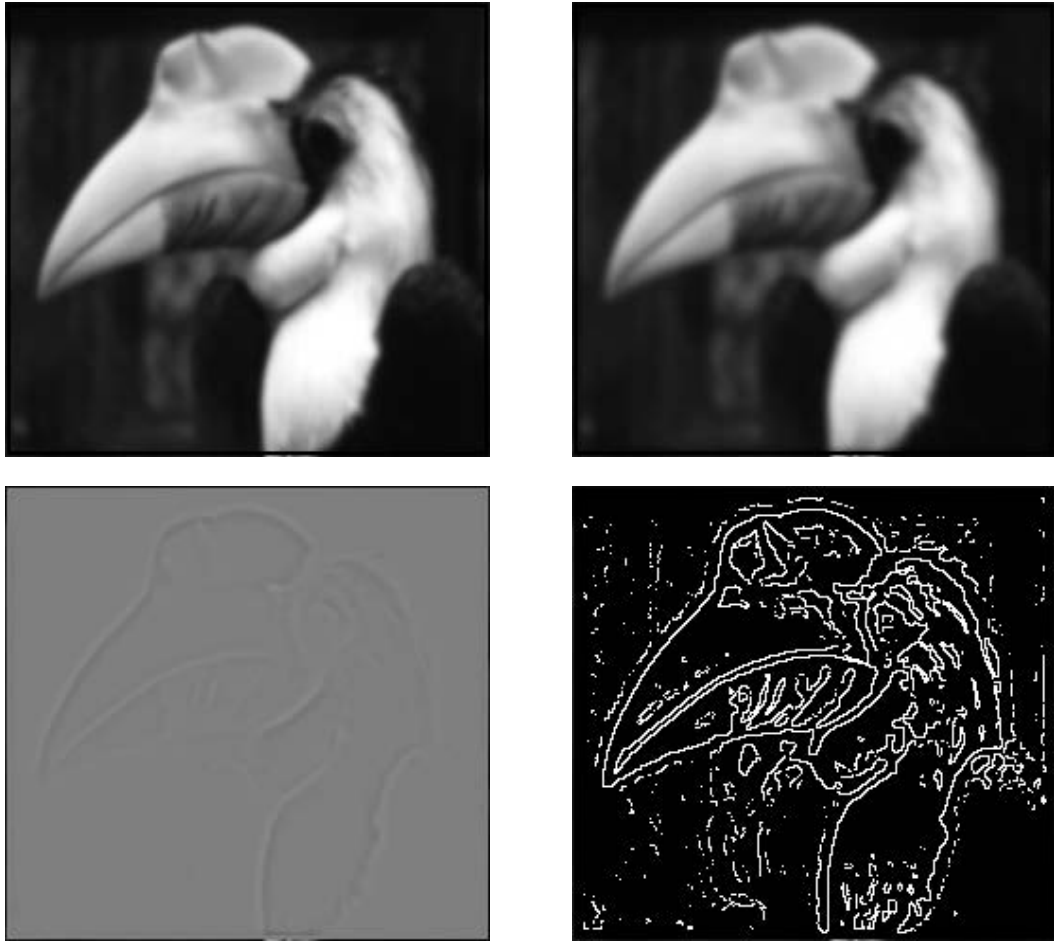
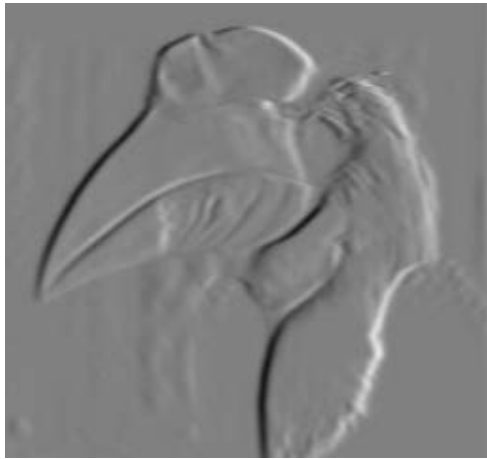
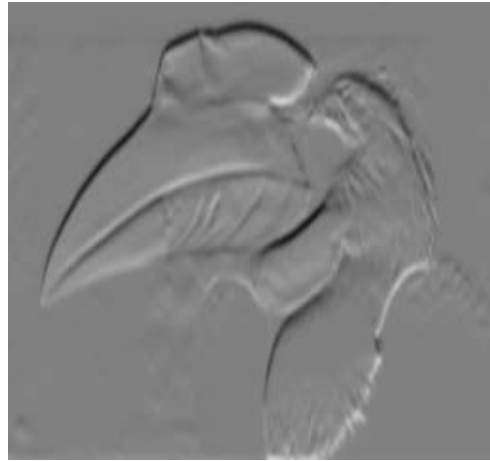


Figura 6.7. Diferencia de gaussianas.



(a)



(b)



(c)



(d)

Figura 6.8 Detector de bordes de Canny.



(e)



(f)



(g)

Figura 6.8 (cont) Detector de bordes de Canny.

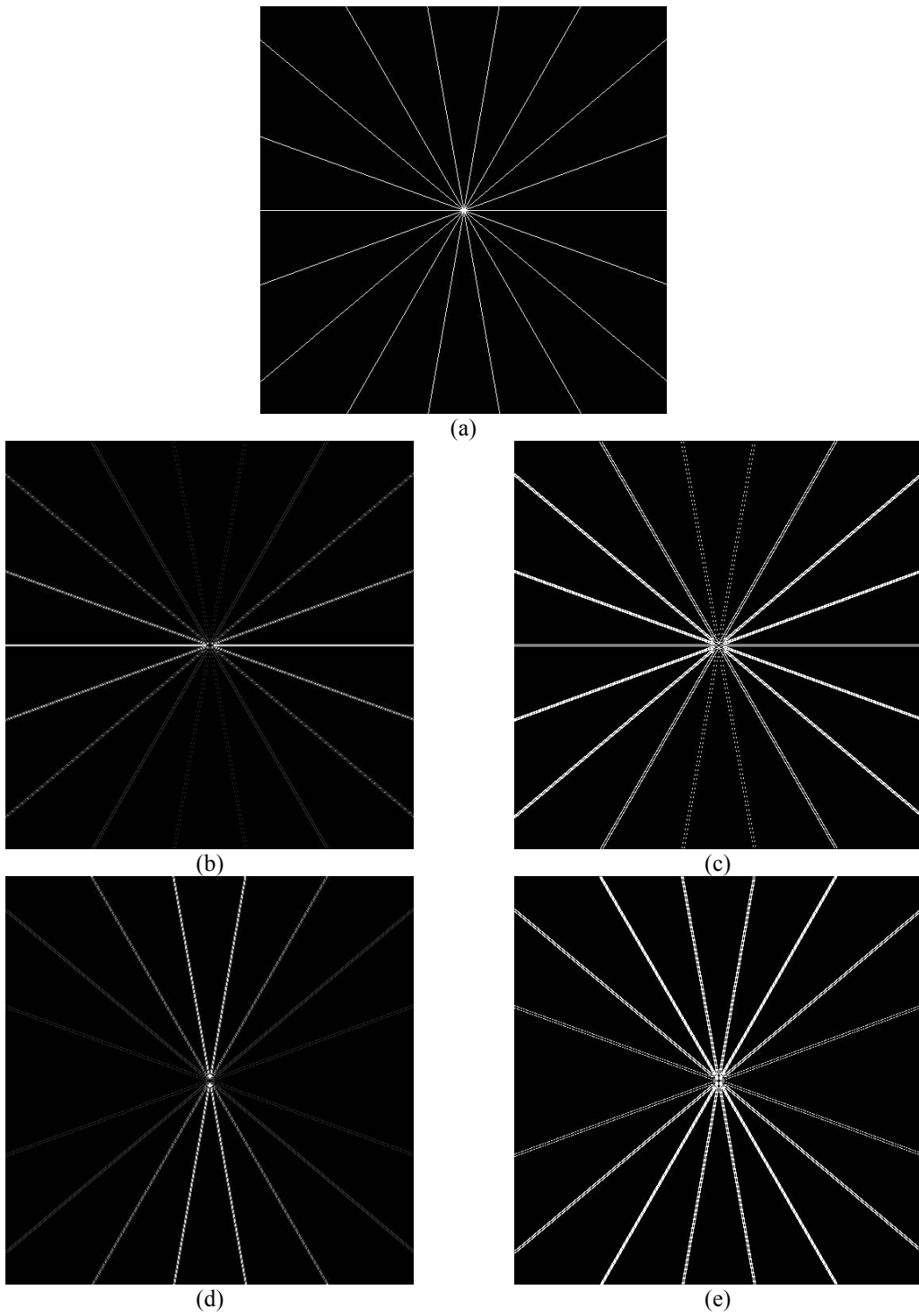
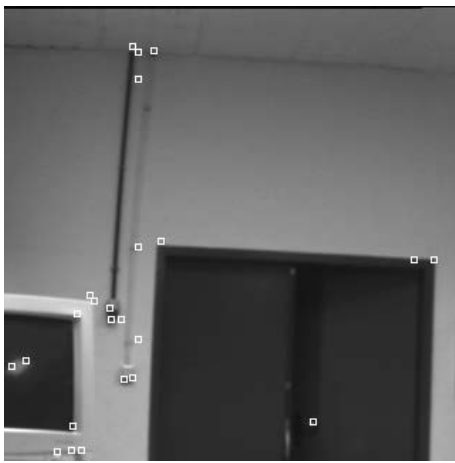


Figura 6.9. Detección de líneas. (a) Imagen original (b) detección de líneas horizontales (c) aplicación del gradiente vertical (d) detección de líneas verticales (e) gradiente horizontal



(a)



(b)



(c)

Figura 6.10. (a) Imagen original, (b) esquinas detectada por el método de Tomasi y Kanade, (c) esquinas detectadas por el método de Kitchen y Rosendfeld

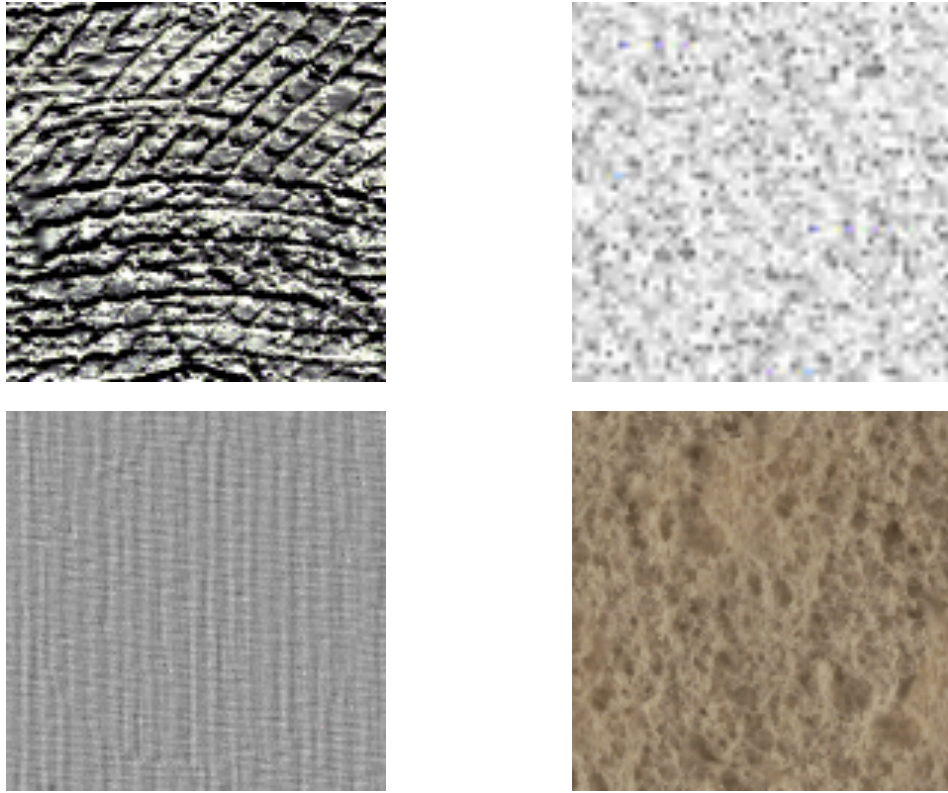


Figura 6.11. Diversas texturas

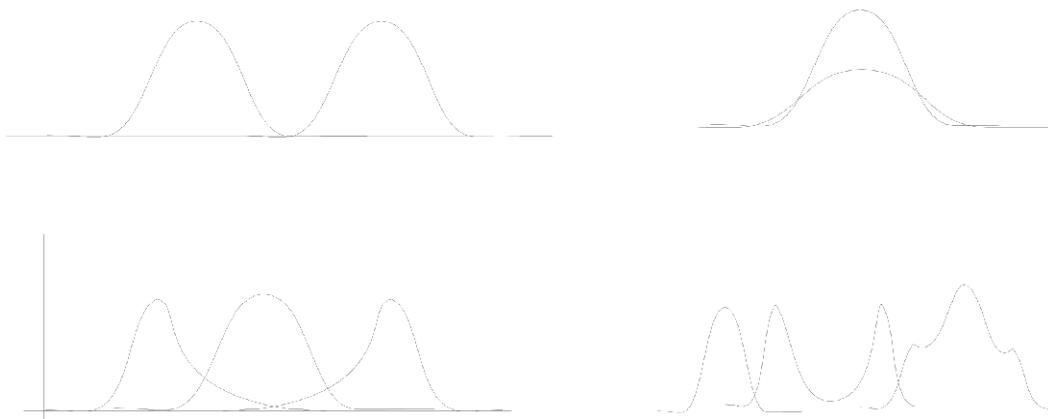


Figura 6.12 Histogramas con distinta media, desviación específica, tercer momento y apuntamiento

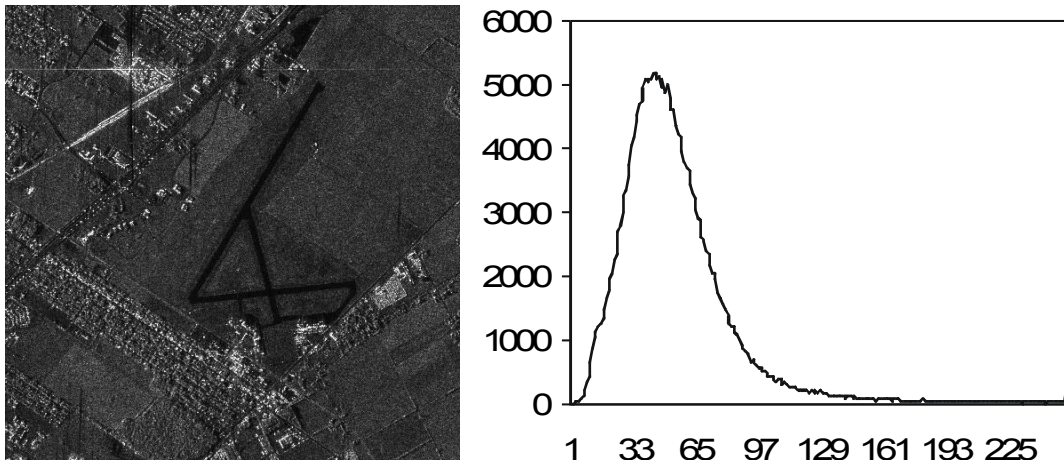


Figura 6.13. Imagen del aeropuerto y su histograma.

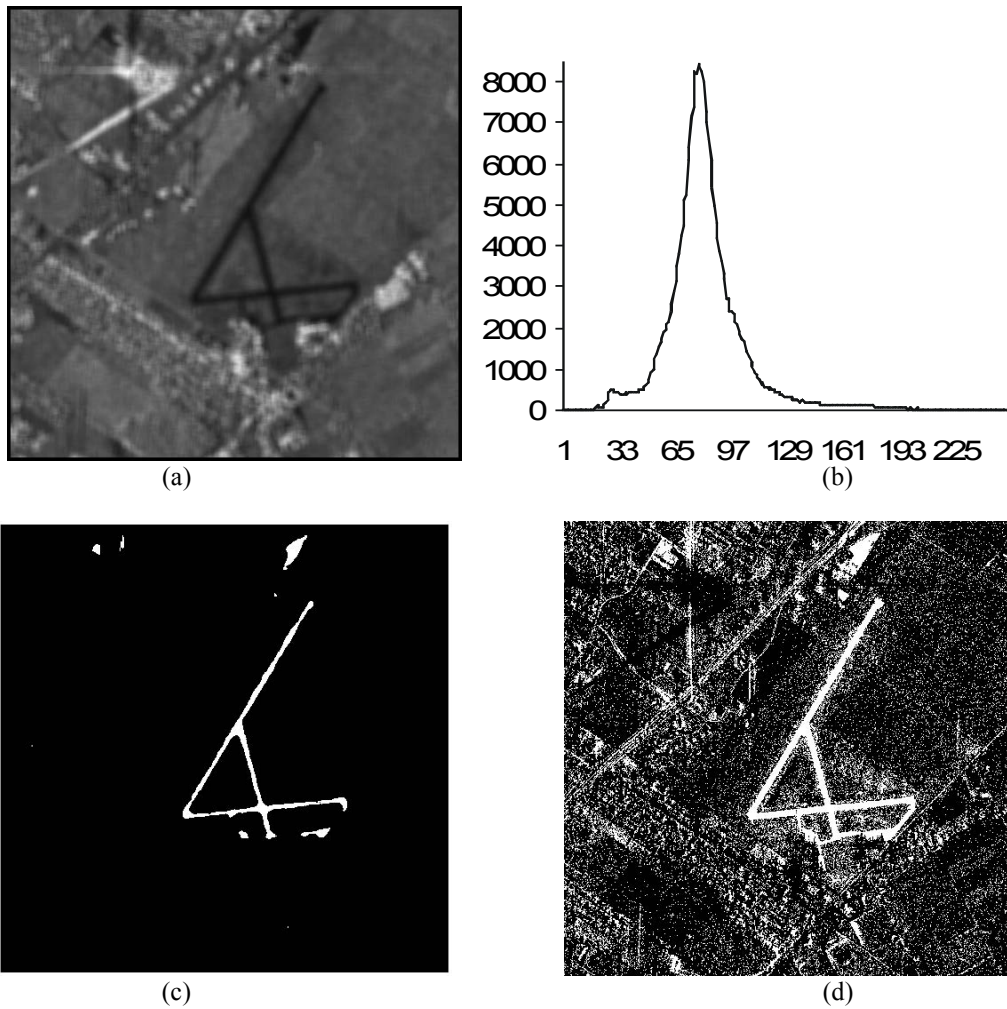


Fig 6.14 Separación entre la zona del aeropuerto, la urbana y el campo mediante texturas

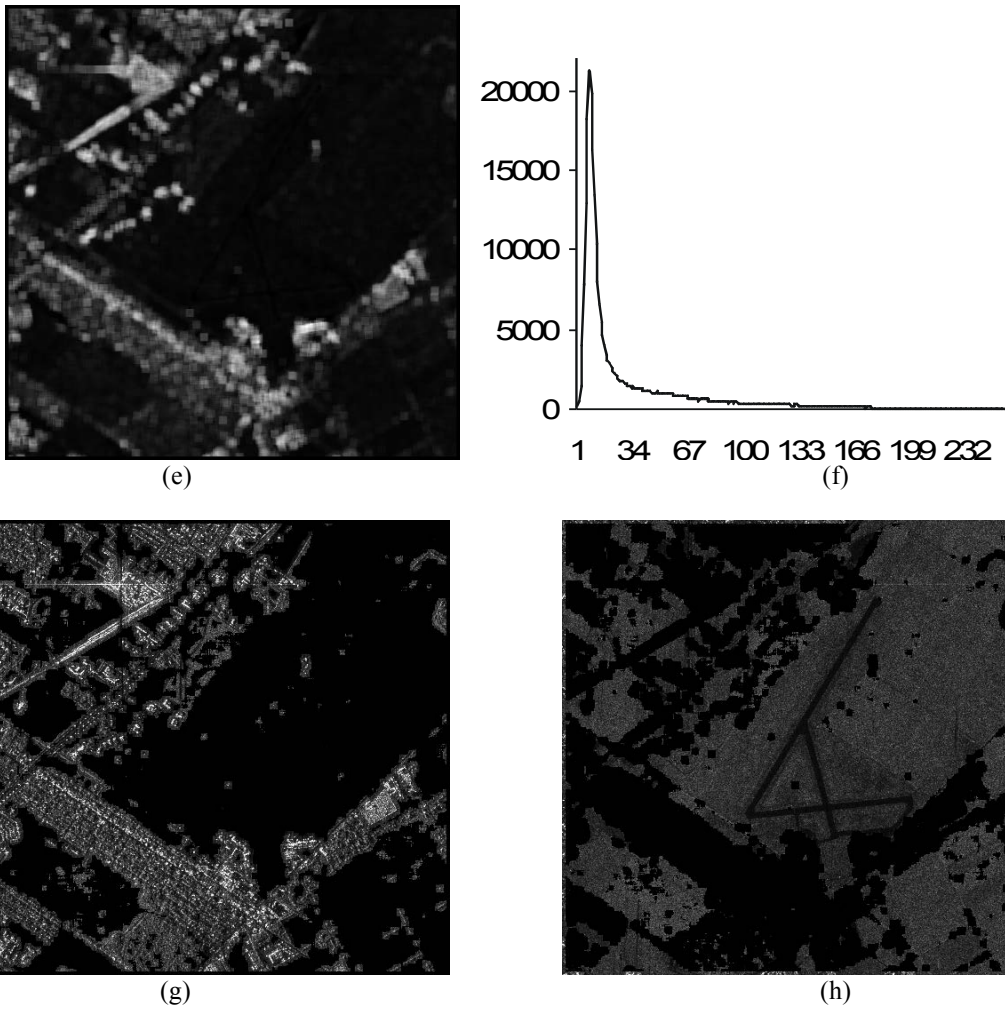


Fig 6.14 (Cont.) Separación entre la zona del aeropuerto, la urbana y el campo mediante texturas

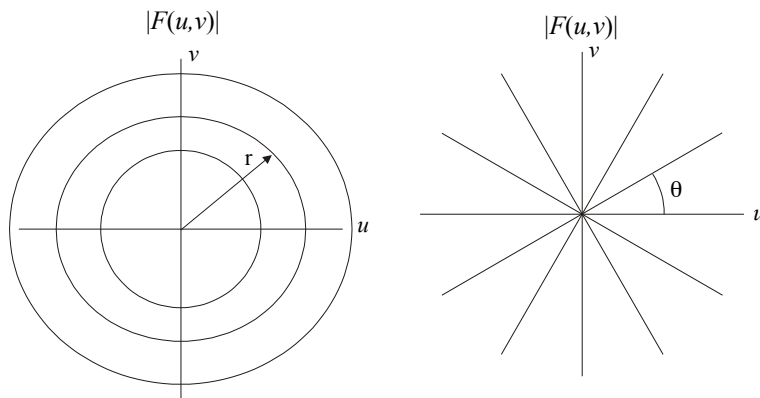
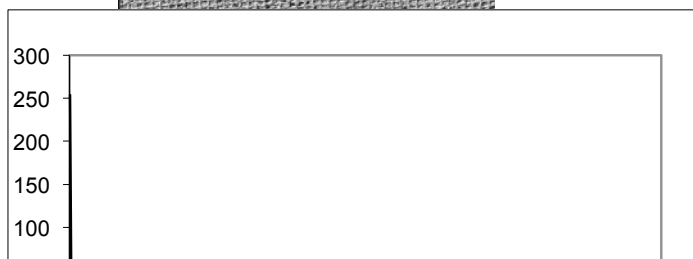
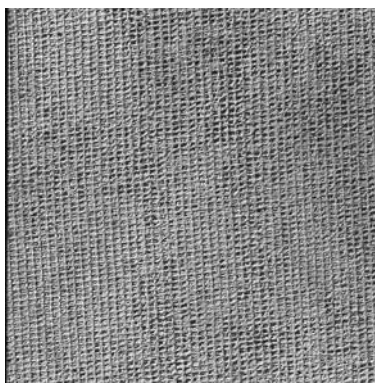
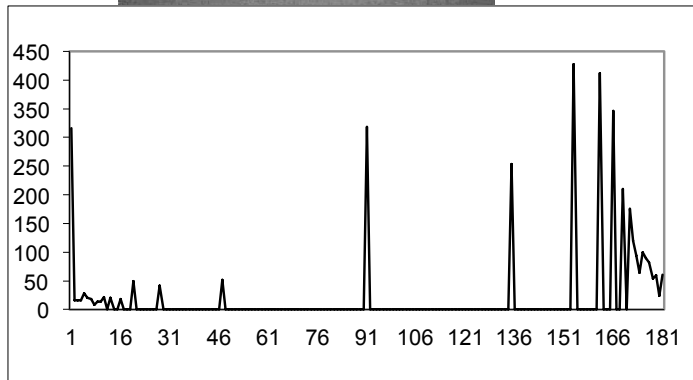
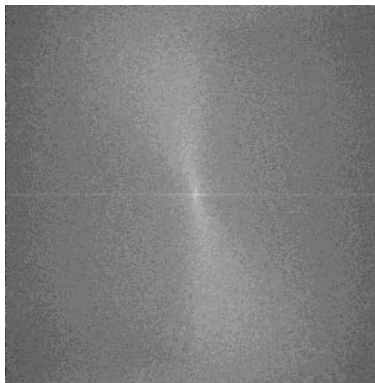
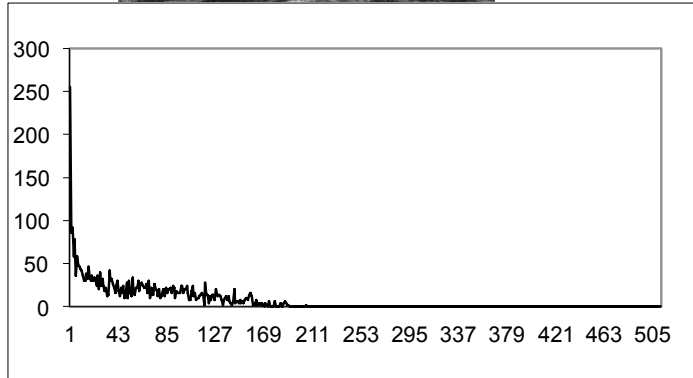
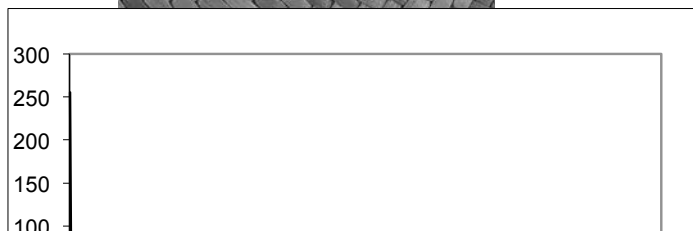
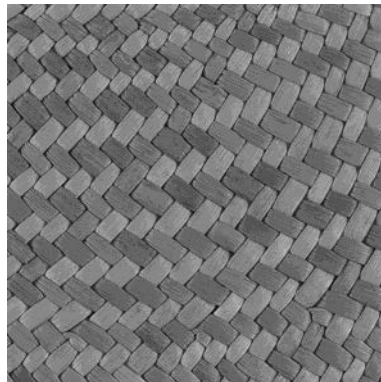
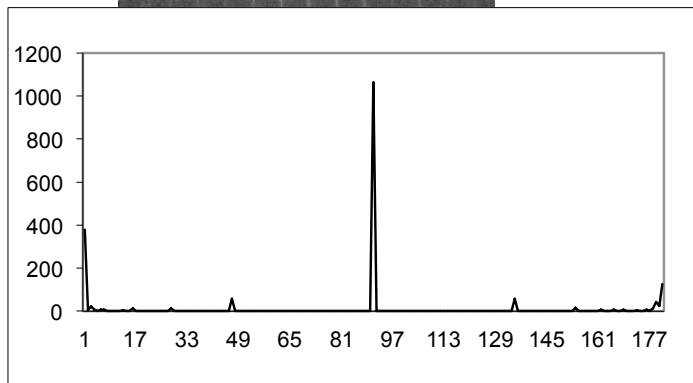
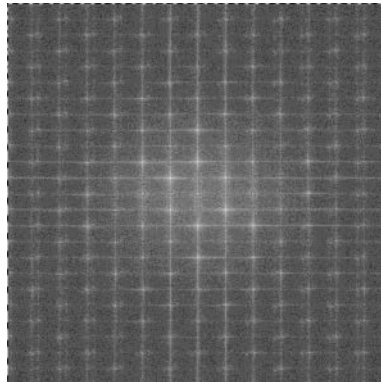
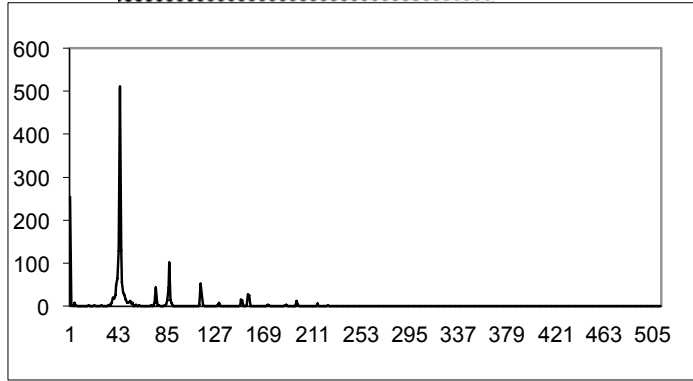
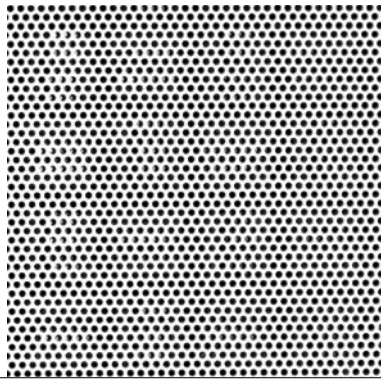


Figura 6.15 Determinación de texturas por Fourier.





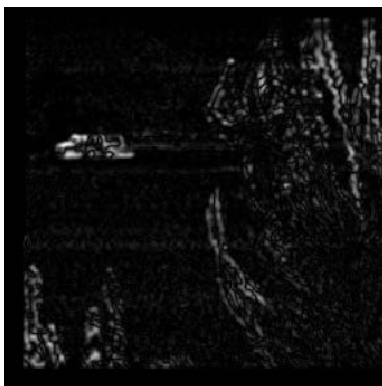


Figura 6.17 Detección de movimiento.

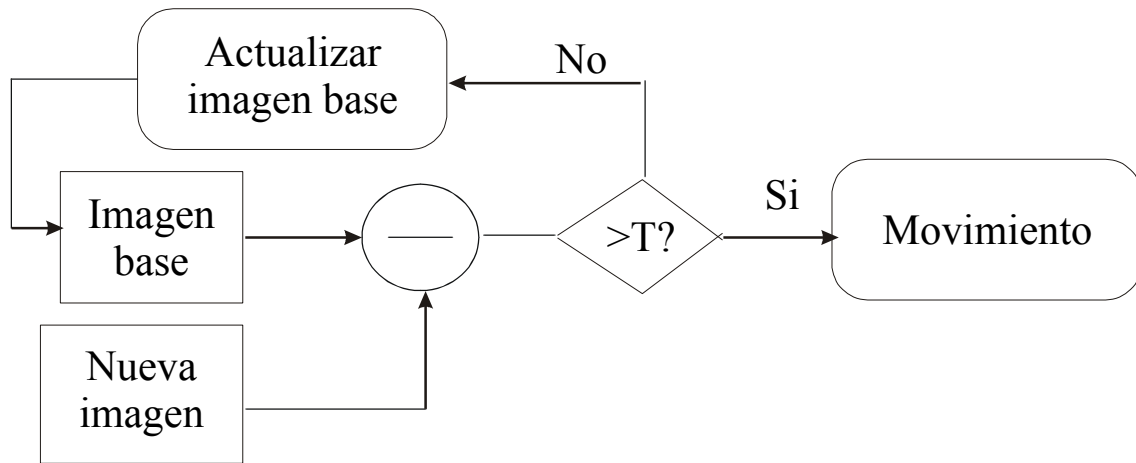


Figura 6.18. Detección de movimiento con actualización del fondo

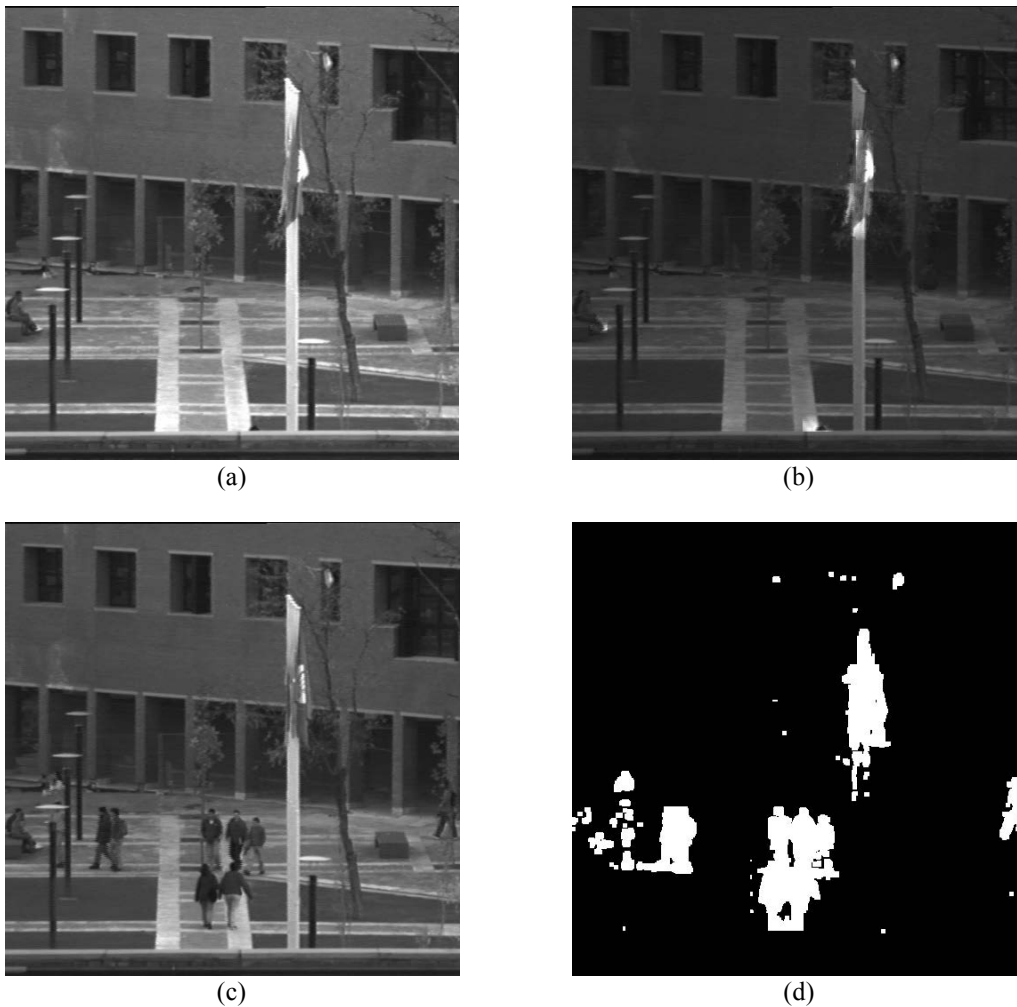


Figura 6.19. (a) Imagen original (b) Imagen base actualizada (c) Imagen (d) movimiento

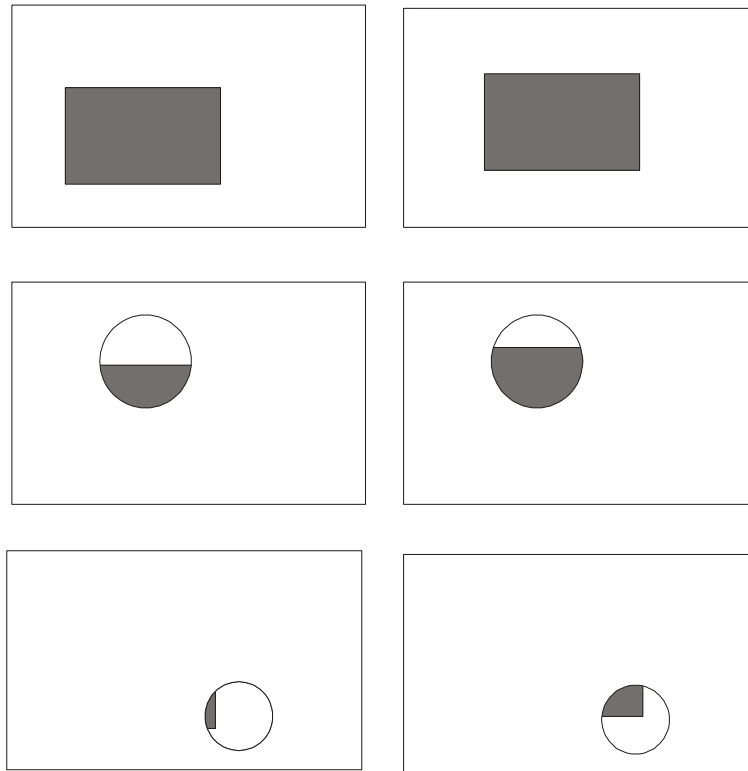


Figura 6.20 Problema de la apertura.

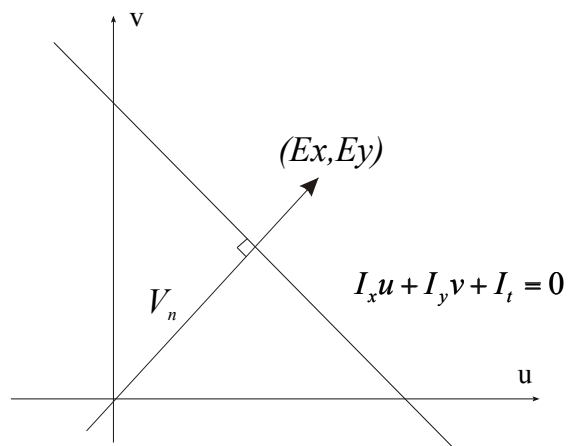


Figura 6.21. Ecuación de restricción del optical flow

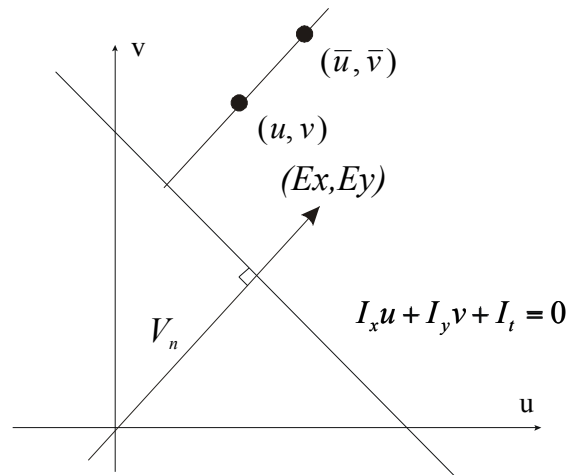


Figura 6.22. Obtención de las dos componentes de la velocidad

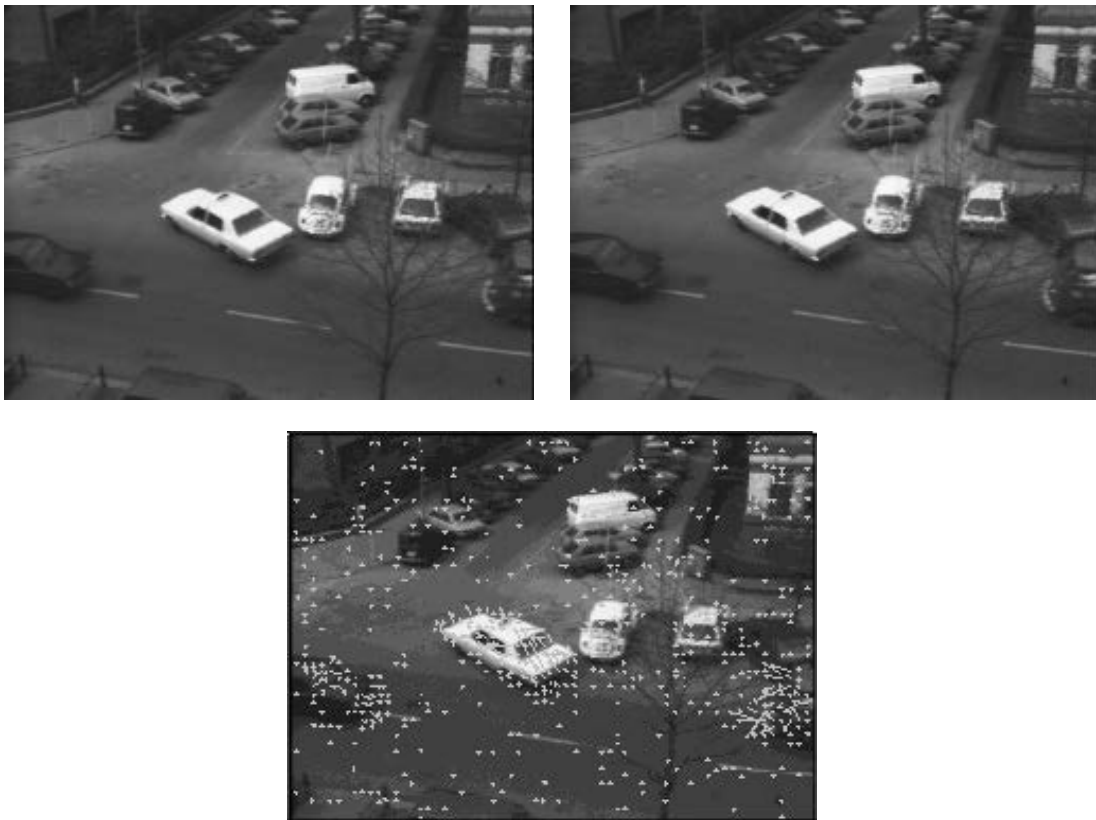


Figura 6.23. Obtención del optical flow.