

Detectores de Características

Guillermo Steiner

Centro de Investigación en Informática para la Ingeniería
Universidad Tecnológica Nacional, F.R.C.

<http://ciiii.frc.utn.edu.ar>

Córdoba, Argentina



Octubre 2011

Detectores de Características

Son partes interesantes de una imagen

Bordes

Puntos que están en la frontera entre dos regiones, se definen generalmente como puntos con gran gradiente.

Esquina o Puntos de interés

Puntos de la imagen con estructura de dos dimensiones basados en principio por un simple detector de borde y luego en la curvatura del gradiente.

Blobs

Descriptores basados en regiones de la imagen.

Detectores de Características

Son partes interesantes de una imagen

Bordes

Puntos que están en la frontera entre dos regiones, se definen generalmente como puntos con gran gradiente.

Esquina o Puntos de interés

Puntos de la imagen con estructura de dos dimensiones basados en principio por un simple detector de borde y luego en la curvatura del gradiente.

Blobs

Descriptores basados en regiones de la imagen.

Detectores de Características

Son partes interesantes de una imagen

Bordes

Puntos que están en la frontera entre dos regiones, se definen generalmente como puntos con gran gradiente.

Esquina o Puntos de interés

Puntos de la imagen con estructura de dos dimensiones basados en principio por un simple detector de borde y luego en la curvatura del gradiente.

Blobs

Descriptores basados en regiones de la imagen.

Detectores de Características

Son partes interesantes de una imagen

Bordes

Puntos que están en la frontera entre dos regiones, se definen generalmente como puntos con gran gradiente.

Esquina o Puntos de interés

Puntos de la imagen con estructura de dos dimensiones basados en principio por un simple detector de borde y luego en la curvatura del gradiente.

Blobs

Descriptores basados en regiones de la imagen.

Detector de Bordes

Se basa en identificar puntos en una imagen cuyo brillo cambia abruptamente (discontinuidad en el brillo)

Captura Eventos o cambios en las propiedades de un imagen como ser:

- Discontinuidades en la profundidad de un objeto.
- Cambios de orientación de sus superficies.
- Cambios de material o textura.

La existencia de un borde implica la fluctuación en el brillo, este hecho introduce el concepto de derivada (primera y segunda)

Detector de Bordes

Se basa en identificar puntos en una imagen cuyo brillo cambia abruptamente (discontinuidad en el brillo)

Captura Eventos o cambios en las propiedades de un imagen como ser:

- Discontinuidades en la profundidad de un objeto.
- Cambios de orientación de sus superficies.
- Cambios de material o textura.

La existencia de un borde implica la fluctuación en el brillo, este hecho introduce el concepto de derivada (primera y segunda)

Detector de Bordes

Se basa en identificar puntos en una imagen cuyo brillo cambia abruptamente (discontinuidad en el brillo)

Captura Eventos o cambios en las propiedades de un imagen como ser:

- Discontinuidades en la profundidad de un objeto.
- Cambios de orientación de sus superficies.
- Cambios de material o textura.

La existencia de un borde implica la fluctuación en el brillo, este hecho introduce el concepto de derivada (primera y segunda)

Detector de Bordes

- Baja Respuesta a regiones uniformes
- Isotrópico, independiente a la orientación del borde
- Baja detección de Falsos Positivos
- Baja pérdida de bordes
- Buena Detección
- Respuesta Única

Detector de Bordes

- Detectores de Primera Derivada.
 - ▶ Diferencias
 - ▶ Roberts
 - ▶ Prewitt
 - ▶ Sobel
- Detectores de Segunda Derivada.
 - ▶ Laplaciano de Gaussiano

Detector de Bordes

- Detectores de Primera Derivada.
 - ▶ Diferencias
 - ▶ Roberts
 - ▶ Prewitt
 - ▶ Sobel
- Detectores de Segunda Derivada.
 - ▶ Laplaciano de Gaussiano

Detectores de Primera Derivada

$$G[f(x, y)] = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \frac{\partial}{\partial x} f(x, y) \\ \frac{\partial}{\partial y} f(x, y) \end{bmatrix} \quad (1)$$

$$\|G\|_{L2} = \sqrt{G_x^2 + G_y^2} \quad (2)$$

$$\|G\|_{L1} = G_x + G_y \quad (3)$$

$$\text{ang}G = \tan^{-1} \left(\frac{G_y}{G_x} \right) \quad (4)$$

Discretizado

$$G_x = f(x + 1, y) - f(x, y) \quad (5)$$

$$G_y = f(x, y + 1) - f(x, y) \quad (6)$$

Detectores de Primera Derivada

a $M_{G_x} =$

-1	1
----	---

$M_{G_y} =$

-1
1

b $M_{G_x} =$

1	0
0	-1

$M_{G_y} =$

0	1
-1	0

c $M_{G_x} =$

-1	0	1
-1	0	1
-1	0	1

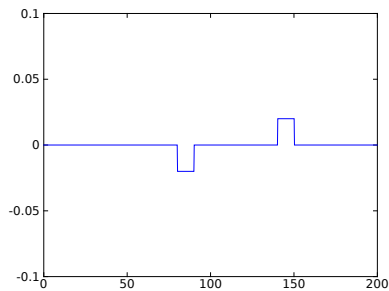
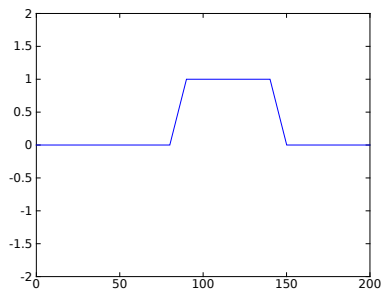
$M_{G_y} =$

1	1	1
0	0	0
-1	-1	-1

Mascara para cálculo de 1era Derivada a)Diferencias b)Roberts c)Prewitt

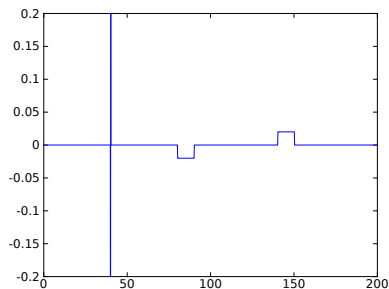
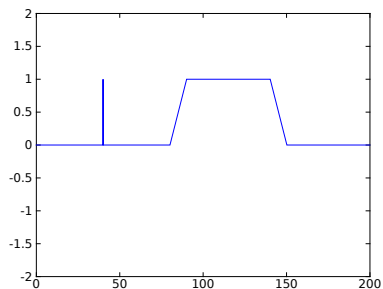
Detección de Borde

$$I' = D(I) \quad (7)$$



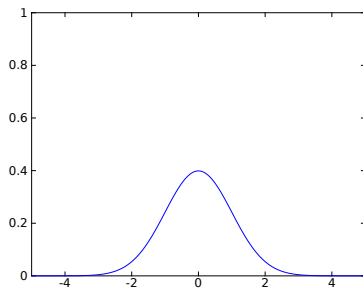
Detección de Borde con Ruido

$$I' = D(I) \quad (8)$$



Filtro Gaussiano

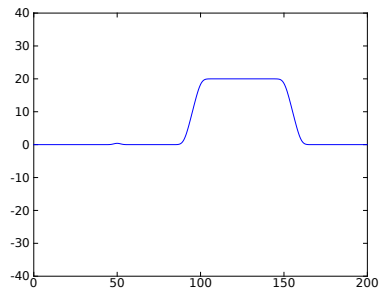
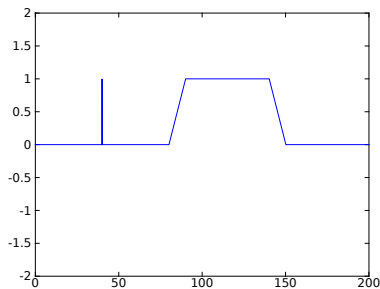
$$F(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}} \quad (9)$$



Filtro Gaussiano

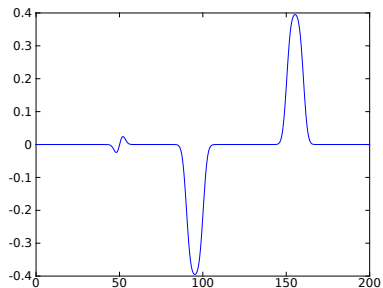
Convolución entre un kernel Gaussiano y la imagen

$$I' = D(F * I) \quad (10)$$



Filtro Gaussiano

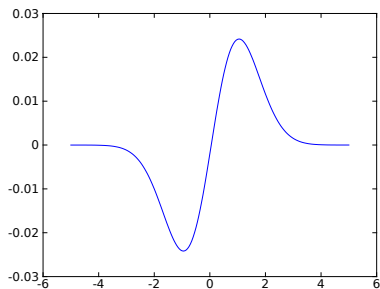
Resultado



Filtro Gaussiano

Derivar primero la gaussiana

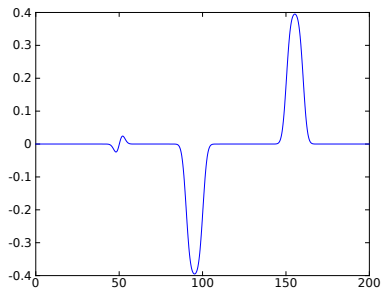
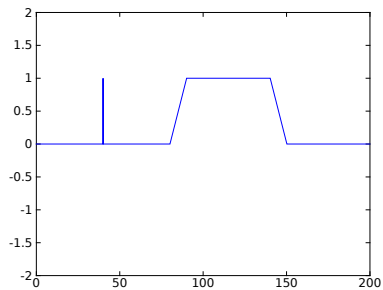
$$I' = D(F) * I$$



Filtro Gaussiano

Resultado

$$I' = D(F) * I$$



Filtro Gaussiano

Filtro de Sobel

a $M_{G_x} =$

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

$M_{G_y} =$

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

b $M_{G_x} =$

-3	0	3
-10	0	10
-3	0	3

$M_{G_y} =$

3	10	3
0	0	0
-3	-10	-3

Figura: a) Sobel b) Schar

Detector de segunda derivada - Laplaciano de Gaussiano

$$\Delta I = \nabla^2 I = \frac{\partial^2 I}{\partial x^2} + \frac{\partial^2 I}{\partial y^2} \quad (11)$$

Aproximaciones de Laplacianas de gaussiana

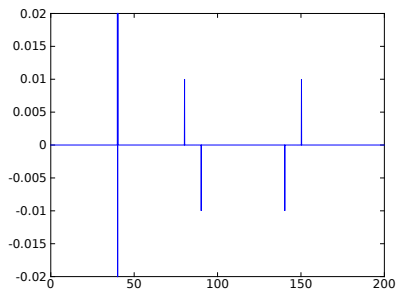
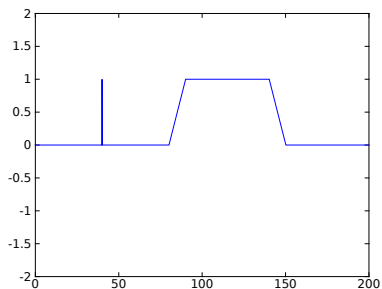
$$M_L = \begin{array}{|c|c|c|} \hline 0 & 1 & 0 \\ \hline 1 & -4 & 1 \\ \hline 0 & 1 & 0 \\ \hline \end{array}$$

$$M_L = \begin{array}{|c|c|c|} \hline 1 & 1 & 1 \\ \hline 1 & -8 & 1 \\ \hline 1 & 1 & 1 \\ \hline \end{array}$$

$$M_L = \begin{array}{|c|c|c|} \hline -1 & 2 & -1 \\ \hline 2 & -4 & 2 \\ \hline -1 & 2 & -1 \\ \hline \end{array}$$

Detección de Borde con Ruido

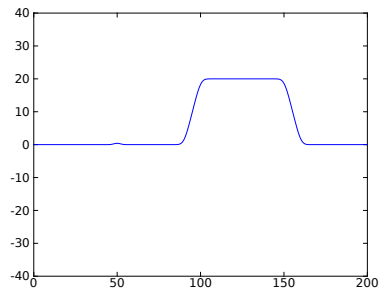
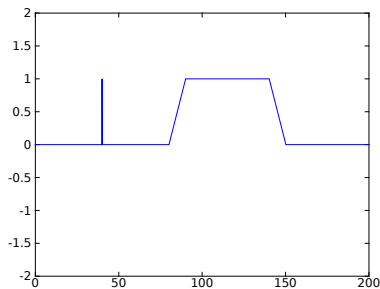
$$I'' = \Delta(I) \quad (12)$$



Filtro Gaussiano

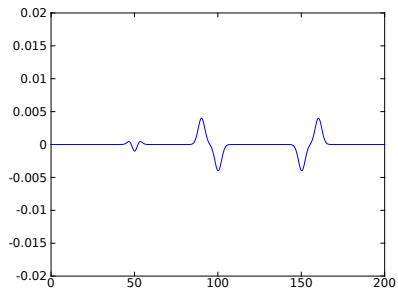
Convolución entre un kernel Gaussiano y la imagen

$$I'' = \Delta(F * I) \quad (13)$$



Filtro Gaussiano

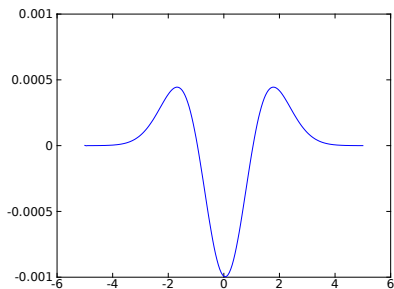
Resultado



Filtro Gaussiano

Derivar dos veces primero la gaussiana

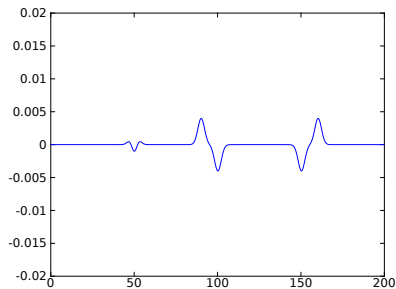
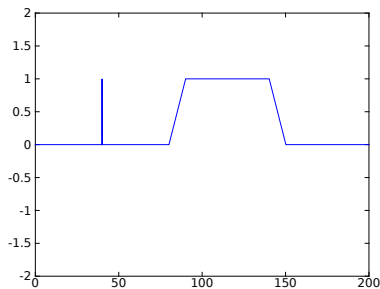
$$I'' = \Delta(F) * I$$



Filtro Gaussiano

Resultado de aplicar el Laplaciano del Gaussiano a la imagen

$$I'' = \Delta(F) * I$$



Detector Canny

Basado en un método simple de detección de borde de primera derivada incorpora mejoras para disminuir el tamaño de los bordes y evitar cortes.

- Calcula las derivadas parciales por Sobel, Prewitt u otro método y luego computa el modulo y ángulo del gradiente.
- Implementa un método de adelgazamiento del borde por el método de supresión no máxima.
- Detecta puntos por medio de un umbral con histéresis.

Detector Canny

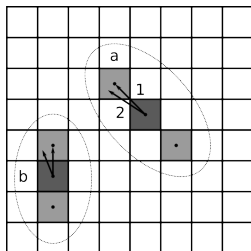
Basado en un método simple de detección de borde de primera derivada incorpora mejoras para disminuir el tamaño de los bordes y evitar cortes.

- Calcula las derivadas parciales por Sobel, Prewitt u otro método y luego computa el modulo y ángulo del gradiente.
- Implementa un método de adelgazamiento del borde por el método de supresión no máxima.
- Detecta puntos por medio de un umbral con histéresis.

Detector Canny

Basado en un método simple de detección de borde de primera derivada incorpora mejoras para disminuir el tamaño de los bordes y evitar cortes.

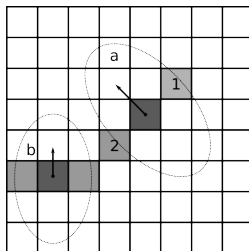
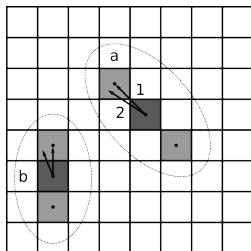
- Calcula las derivadas parciales por Sobel, Prewitt u otro método y luego computa el modulo y ángulo del gradiente.
- Implementa un método de adelgazamiento del borde por el método de supresión no máxima.
- Detecta puntos por medio de un umbral con histéresis.



Detector Canny

Basado en un método simple de detección de borde de primera derivada incorpora mejoras para disminuir el tamaño de los bordes y evitar cortes.

- Calcula las derivadas parciales por Sobel, Prewitt u otro método y luego computa el modulo y ángulo del gradiente.
- Implementa un método de adelgazamiento del borde por el método de supresión no máxima.
- Detecta puntos por medio de un umbral con histéresis.



Detectores de Puntos de Interés / Esquinas

- Detector de Moravec.
- Harris & Stephens / Plessey / Shi-Tomasi.
- Detector de Trajkovic.
- Detector FAST (Features from Accelerated Segment Test).

Detector de Moravec

Punto de interés - regiones en la imágenes que pueden ser localizadas nuevamente en imágenes sucesivas Auto Correlación

$$c(x, y, \Delta x, \Delta y) = \sum_{i=0}^n [I(x + a_i, y + b_i) - I(x + a_i + \Delta x, y + b_i + \Delta y)]^2 \quad (14)$$

Desventaja

- No es isotropico.

Modificaciones

- Ventana circular.
- Suma Ponderada.

Detector de Moravec

Punto de interés - regiones en la imágenes que pueden ser localizadas nuevamente en imágenes sucesivas Auto Correlación

$$c(x, y, \Delta x, \Delta y) = \sum_{i=0}^n [I(x + a_i, y + b_i) - I(x + a_i + \Delta x, y + b_i + \Delta y)]^2 \quad (14)$$

Desventaja

- No es isotropico.

Modificaciones

- Ventana circular.
- Suma Ponderada.

Detector de Moravec

Punto de interés - regiones en la imágenes que pueden ser localizadas nuevamente en imágenes sucesivas Auto Correlación

$$c(x, y, \Delta x, \Delta y) = \sum_{i=0}^n [I(x + a_i, y + b_i) - I(x + a_i + \Delta x, y + b_i + \Delta y)]^2 \quad (14)$$

Desventaja

- No es isotropico.

Modificaciones

- Ventana circular.
- Suma Ponderada.

Harris & Stephens / Plessey / Shi-Tomasi

Elimina el problema de discretización de direcciones

$$c(x, y) = \sum_{i=0}^n w_i [I(x_i, y_i) - I(x_i + \Delta x, y_i + \Delta y)]^2 \quad (15)$$

Aplicamos Taylor al píxel desplazado

$$I(x_i + \Delta x, y_i + \Delta y) \approx I(x_i, y_i) + \begin{bmatrix} I_x(x_i, y_i) & I_y(x_i, y_i) \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} + \varphi^2() \quad (16)$$

Reemplazamos

$$c(x, y) = \sum_{i=0}^n w_i \left[I(x_i, y_i) - \left(I(x_i, y_i) + \begin{bmatrix} I_x(x_i, y_i) & I_y(x_i, y_i) \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} \right) \right]^2 \quad (17)$$

$$= \sum_{i=0}^n w_i \begin{bmatrix} \Delta x & \Delta y \end{bmatrix} \begin{bmatrix} (I_x(x_i, y_i))^2 & I_x(x_i, y_i)I_y(x_i, y_i) \\ I_x(x_i, y_i)I_y(x_i, y_i) & (I_y(x_i, y_i))^2 \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} \quad (18)$$

Harris & Stephens / Plessey / Shi-Tomasi

Elimina el problema de discretización de direcciones

$$c(x, y) = \sum_{i=0}^n w_i [I(x_i, y_i) - I(x_i + \Delta x, y_i + \Delta y)]^2 \quad (15)$$

Aplicamos Taylor al píxel desplazado

$$I(x_i + \Delta x, y_i + \Delta y) \approx I(x_i, y_i) + \begin{bmatrix} I_x(x_i, y_i) & I_y(x_i, y_i) \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} + \varphi^2() \quad (16)$$

Reemplazamos

$$c(x, y) = \sum_{i=0}^n w_i \left[I(x_i, y_i) - \left(I(x_i, y_i) + \begin{bmatrix} I_x(x_i, y_i) & I_y(x_i, y_i) \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} \right) \right]^2 \quad (17)$$

$$= \sum_{i=0}^n w_i \begin{bmatrix} \Delta x & \Delta y \end{bmatrix} \begin{bmatrix} (I_x(x_i, y_i))^2 & I_x(x_i, y_i)I_y(x_i, y_i) \\ I_x(x_i, y_i)I_y(x_i, y_i) & (I_y(x_i, y_i))^2 \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} \quad (18)$$

Harris & Stephens / Plessey / Shi-Tomasi

Elimina el problema de discretización de direcciones

$$c(x, y) = \sum_{i=0}^n w_i [I(x_i, y_i) - I(x_i + \Delta x, y_i + \Delta y)]^2 \quad (15)$$

Aplicamos Taylor al píxel desplazado

$$I(x_i + \Delta x, y_i + \Delta y) \approx I(x_i, y_i) + \begin{bmatrix} I_x(x_i, y_i) & I_y(x_i, y_i) \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} + \varphi^2() \quad (16)$$

Reemplazamos

$$c(x, y) = \sum_{i=0}^n w_i \left[I(x_i, y_i) - \left(I(x_i, y_i) + \begin{bmatrix} I_x(x_i, y_i) & I_y(x_i, y_i) \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} \right) \right]^2 \quad (17)$$

$$= \sum_{i=0}^n w_i \begin{bmatrix} \Delta x & \Delta y \end{bmatrix} \begin{bmatrix} (I_x(x_i, y_i))^2 & I_x(x_i, y_i)I_y(x_i, y_i) \\ I_x(x_i, y_i)I_y(x_i, y_i) & (I_y(x_i, y_i))^2 \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} \quad (18)$$

Harris & Stephens / Plessey / Shi-Tomasi

$$c(x, y) = \begin{bmatrix} \Delta x & \Delta y \end{bmatrix} \begin{bmatrix} \sum_i w_i (I_x(x_i, y_i))^2 & \sum_i w_i I_x(x_i, y_i) I_y(x_i, y_i) \\ \sum_i w_i I_x(x_i, y_i) I_y(x_i, y_i) & \sum_i w_i (I_y(x_i, y_i))^2 \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix}$$

$$= \begin{bmatrix} \Delta x & \Delta y \end{bmatrix} C(x, y) \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix}$$

donde

$$C(x, y) = \begin{bmatrix} \sum_i w_i (I_x(x_i, y_i))^2 & \sum_i w_i I_x(x_i, y_i) I_y(x_i, y_i) \\ \sum_i w_i I_x(x_i, y_i) I_y(x_i, y_i) & \sum_i w_i (I_y(x_i, y_i))^2 \end{bmatrix}$$

$$C(x, y) = \begin{bmatrix} \langle I_x^2 \rangle & \langle I_x I_y \rangle \\ \langle I_x I_y \rangle & \langle I_y^2 \rangle \end{bmatrix}$$

Harris & Stephens / Plessey / Shi-Tomasi

Si se diagonaliza, podemos sacar sus autovalores.

$$RCR^{-1} = R \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} R^{-1} \quad (19)$$

representando una elipse con semiejes

$$\frac{1}{\sqrt{\lambda_1}} \text{ y } \frac{1}{\sqrt{\lambda_2}} \quad (20)$$

y al ángulo por su autovector

$$\alpha = \arctan \left(\frac{y_1}{x_1} \right) \quad (21)$$

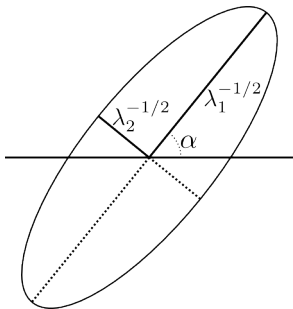
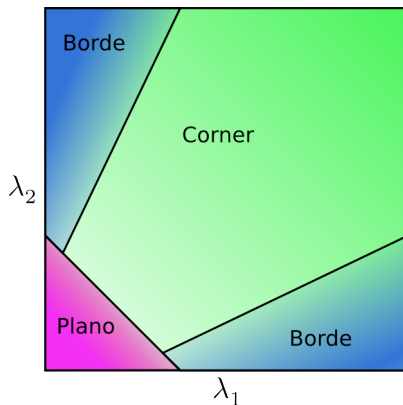


Figura: representación de los autovalores en una elipse

Harris & Stephens / Plessey / Shi-Tomasi

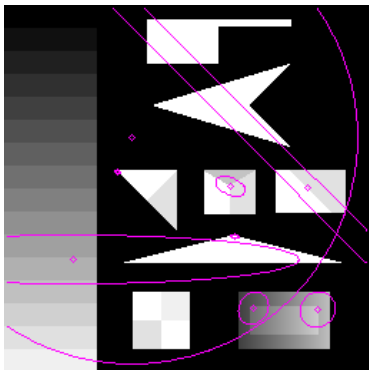
Características de la imagen según los autovalores.

- $\lambda_1 \gg \lambda_2$ o $\lambda_2 \gg \lambda_1$, Borde.
- $\lambda_1 > 0$ y $\lambda_2 > 0$, Esquina o punto de interés.
- $\lambda_1 \simeq 0$ y $\lambda_2 \simeq 0$, Superficie Homogénea.



Harris & Stephens / Plessey / Shi-Tomasi

Respuesta de la segunda derivada en diferentes puntos de la imagen.



Harris & Stephens / Plessey / Shi-Tomasi

- Propuesta de Harris & Stephens / Plessey.
 - ▶ El Determinante (un determinante alto implica ambos autovalores alto)
 - ▶ La Traza (orden de magnitud del autovalor mas grande)

$$Harris(x, y) = \det(C) - k(\text{traza}(C))^2 \quad (22)$$

donde

$$\det(C) = \lambda_1 \lambda_2 = I_x^2 I_y^2 - I_{xy}^2 \quad (23)$$

$$\text{traza}(C) = \lambda_1 + \lambda_2 = I_x^2 + I_y^2 \quad (24)$$

- Propuesta Shi-Tomasi.

$$ST(x, y) = \min(\lambda_1, \lambda_2) \quad (25)$$

Detector de Trajkovic

Aproximación Interpixel en ventana de 3x3

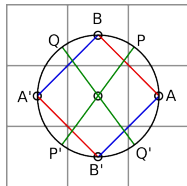


Figura: Circulo del detector de Trajkovic

$$T(x, y) = \min((B - C)^2 + (B' - C)^2, (A - C)^2 + (A' - C)^2) \quad (27)$$

Detector de Trajkovic

Aproximación Interpixel en ventana de 3x3

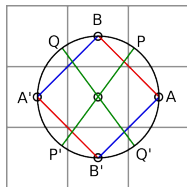


Figura: Circulo del detector de Trajkovic

Si tomamos dos puntos (P P') y (Q Q') podemos calcular sus respuestas

$$r_1(x) = (P_x - C)^2 + (P'_x - C)^2, r_2(x) = (Q_x - C)^2 + (Q'_x - C)^2 \quad (27)$$

$$T(x, y)_{\text{interpixel}} = \min_{x \in (0,1)} (r_1(x), r_2(x)) \quad (28)$$

$$P(x) = (1 - x)A + xB \quad P'(x) = (1 - x)A' + xB' \quad (29)$$

$$Q(x) = (1 - x)A' + xB \quad Q'(x) = (1 - x)A + xB' \quad (30)$$

Detector de Trajkovic

Aproximación Interpixel en ventana de 3x3

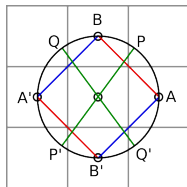


Figura: Círculo del detector de Trajkovic

Calculo del mínimo

$$T(x, y)_{\text{interpixel}} = \begin{cases} r_A - \frac{m_2^2}{m_3} & \text{si } (m_2 < 0) \text{ y } (m_2 + m_3) > 0 \\ T(x, y) & \text{sino} \end{cases} \quad (27)$$

donde

$$m_2 = \min(p_2, q_2) \text{ y } m_3 = r_B - r_A - 2m_2 \quad (28)$$

Detector FAST (Features from Accelerated Segment Test)

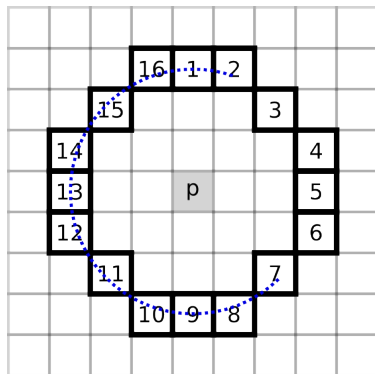
Detección de Puntos Característicos basado en Árboles de Decisión

Método Convencional

- Se busca 12 pixeles contiguos que cumplan con

$$I_{p \rightarrow x} < I_p - t \text{ o } I_{p \rightarrow x} > I_p + t$$

- permite un descarte rápido si hay menos de 3 píxel del conjunto (1,5,9 y 13) que no cumple con lo anterior
- El Problema supone una estructura que respeta los 12 píxeles contiguos.



Detector FAST (Features from Accelerated Segment Test)

Arboles de Decisión

- Por cada posición x se calcula para cada punto $I_{p \rightarrow x}$

$$S_{p \rightarrow x} = \begin{cases} d, & I_{p \rightarrow x} \leq I_p - t & \text{(oscuro)} \\ s, & I_p - t < I_{p \rightarrow x} < I_p + t & \text{(similar)} \\ b, & I_p + t \leq I_{p \rightarrow x} & \text{(brillante)} \end{cases} \quad (29)$$

y se dividen en tres subconjuntos P_d, P_s, P_b

- Se construye un conjunto K_p de tipo booleano, que contendrá por cada p *true* si es punto de interés y *false* si no lo es.
- Se construye el árbol ID3 partiendo del x con la mayor Entropía en K

$$H(Q) = (c + \bar{c}) \log_2(c + \bar{c}) - c \log_2 c - \bar{c} \log_2 \bar{c} \quad (30)$$

$$c = |\{q \mid K_q \text{ es true}\}| \quad (\text{número de corners})$$

$$\bar{c} = |\{q \mid K_q \text{ es false}\}| \quad (\text{número de no corners})$$

Detector FAST (Features from Accelerated Segment Test)

Cálculo del Árbol

- Bucle Recursivo comenzando por el x de mayor entropía

$$H(P) - H(P_{d \rightarrow x}) - H(P_{s \rightarrow x}) - H(P_{b \rightarrow x}) \quad (31)$$

- se repite la operación pero para cada uno de los subconjuntos $P_{d \rightarrow x}, P_{s \rightarrow x}, P_{b \rightarrow x}$, tendremos para $P_{d \rightarrow x} = P'_d \cup P'_s \cup P'_b$
- El procedimiento concluye, cuando la entropía del subconjunto sea nula, indicando que todos los elementos del mismo son esquinas o no lo son, pudiendo etiquetar finalmente.

Detector FAST (Features from Accelerated Segment Test)

Función de Respuesta

$$V = \text{máx} \left(\sum_{x \in S_b} |I_{p \rightarrow x} - I_p| - t, \sum_{x \in S_d} |I_{p \rightarrow x} - I_p| - t \right) \quad (32)$$

donde

$$S_b = \{x \mid I_{p \rightarrow x} \geq I_p + t\} \quad (\text{brillosos})$$

$$S_d = \{x \mid I_{p \rightarrow x} \leq I_p - t\} \quad (\text{oscuros})$$

FIN !!!
¿ Preguntas ?