

Asignatura: Procesamiento Digital de Imágenes

Profesor: Dr.Sc. Gerardo García Gil

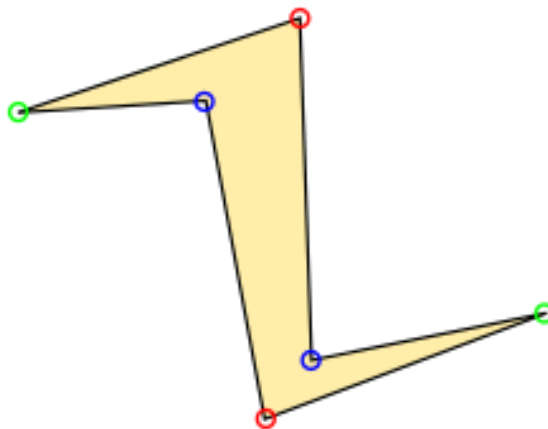
Alumno: René Francisco Coss y León Monterde

Registro: 17310066 **Semestre:** 2020-B
Ingeniería en Desarrollo de Software

INTRODUCCIÓN.

Esquinas

Intersección de dos líneas que forman un ángulo. En el caso de una figura es el punto donde se conectan dos aristas, conocido como vértice.



Detección de esquinas

La detección de esquinas se utiliza para el análisis de imágenes, la detección de movimiento y especialmente el procesamiento 3D (que permite identificar y crear objetos de la vida real).

Sus características principales son:

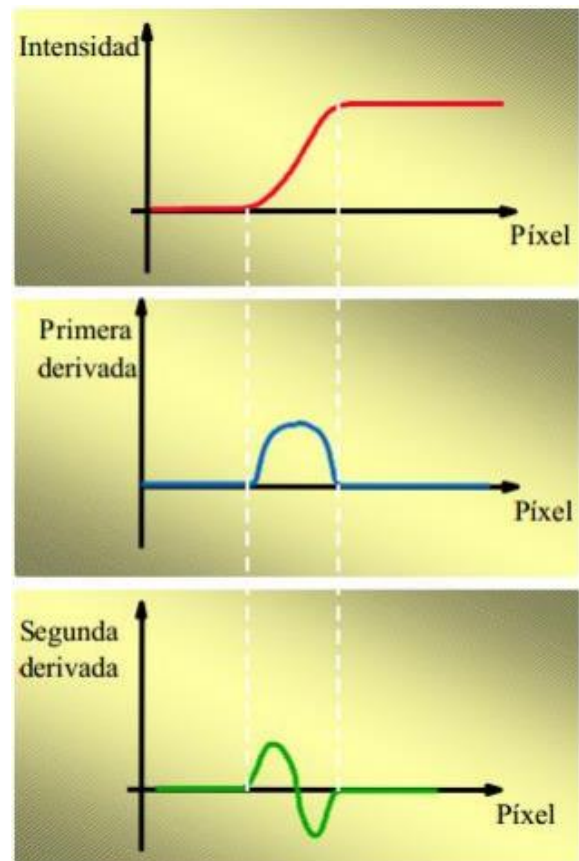
- Todas las esquinas y vértices se deben de detectar.
- No se deben de detectar falsas esquinas.
- Debe de ser robusto frente al ruido.
- La localización debe de ser precisa.

Para lograr esto, primeramente, hay que extraer y seguir los bordes.

Algoritmos de detección de esquinas

Los algoritmos utilizados para detectar esquinas utilizan el criterio de la primera o la segunda derivada sobre la imagen, tales como:

- Umbralización de la primera derivada
- Centro de gravedad de la primera derivada
- Máximo de la primera derivada
- Paso por cero de la segunda derivada



Algoritmo Wang & Brady

Este algoritmo usa un método de borde-relación, basado en la curvatura del gradiente en imagen.

Además, considera a una imagen como una superficie, en la que la dirección de un borde cambia abruptamente, siendo cualquier lugar con esta condición una esquina.

Este algoritmo se considera como una mejora al Kitchen & Rosenfeld, siendo muy utilizado para procesos en tiempo real.

Matemáticamente

Se necesita medir un coeficiente C con:

$$C(x, y) = \nabla^2 I(x, y) + c |\nabla I(x, y)|$$

Donde tenemos que:

$$\nabla I(x, y) = -0.5I(x-1, y) + 0.5I(x+1, y) - 0.5I(x, y-1) + 0.5I(x, y+1)$$

$$\nabla I(x, y) = \begin{bmatrix} 0 & -0.5 & 0 \\ -0.5 & 0 & 0.5 \\ 0 & 0.5 & 0 \end{bmatrix}$$

y

$$\nabla^2 I(x, y) = I(x+1, y) + I(x-1, y) + I(x, y+1) + I(x, y-1) - 4I(x, y)$$

$$\nabla^2 I(x, y) = \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

Y el valor c es una constante que permite calibrar la sensibilidad del algoritmo

DESARROLLO.

Con estos algoritmos se utilizará una imagen en sus 3 dimensiones RGB. La lectura de la imagen es la misma que se ha manejado en prácticas anteriores, pero no se aplicará el escalado a grises que se ha utilizado hasta ahora.

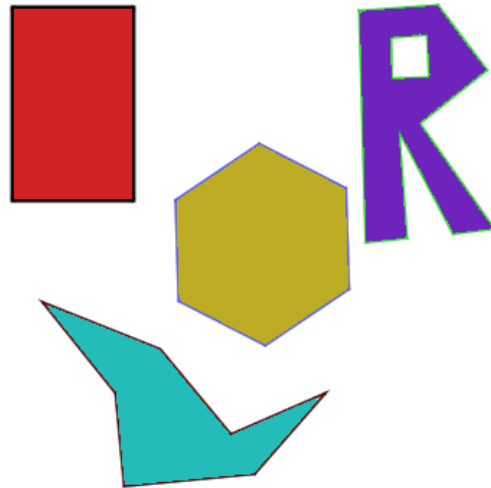


Fig1. Figuras que se van a utilizar para realizar las pruebas. Cuadrado (a), Pentágono (b), Origami (c) y R (d).

Utilizando la codificación que proporcionó el profesor podemos comenzar con las pruebas, simplemente cargaremos la imagen en el programa para visualizarlo de la siguiente forma.

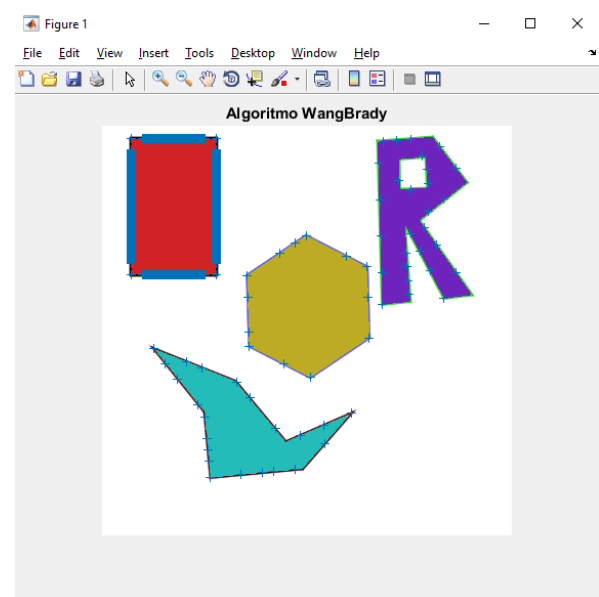


Fig2. Resultados del algoritmo Wang & Brady, las cruces marcan las esquinas detectadas.

Se puede notar que las esquinas de las figuras están marcadas por cruces. Incluso detecta las orillas donde cambia el marco en el que está la imagen y el resto de la ventana

Código

```
clear all; clc;
Iorig = imread('figuras.png');

Im = double(rgb2gray(Iorig));
h=ones(3)/9;
Im = imfilter(Im,h);
d1=[0,-0.5,0;-0.5,0,0.5;0,0.5,0];
d2=[0,1,0;1,-4,1;0,1,0];
I1 = imfilter(Im,d1);
I2 = imfilter(Im,d2);
c= 1;
V = (I2 - c*abs(I1));
V=( 1000/max(max(V)))*V;
V1= (V)>100;
pixel = 10;
[n,m] = size(V1);
res = zeros(n,m);

for r=1:n
for c=1:m
if (V1(r,c))
I1=[r-pixel,1];
I1 =max(I1);
I2=[r+pixel,n];
I2=min(I2);
I3=[c-pixel,1];
I3 =max(I3);
I4=[c+pixel,m];
I4=min(I4);
tmp = V(I1:I2,I3:I4);
maxim = max(max(tmp));
if(maxim == V(r,c))
res(r,c) = 1;
end
end
end
end

imshow(uint8(Iorig));
title("Algoritmo WangBrady");

hold on
[re,co] = find(res);
plot(re,co,'+');
```

CONCLUSIONES

Los valores obtenidos en esta práctica son los esperados en casi su totalidad, viendo la teoría a lado de los resultados prácticos podemos ver dos puntos muy importantes: su similitud con Kitchen & Rosenfeld y la consideración de curva para marcar las orillas.

Este algoritmo puede ser tomado como detección de bordes, las figuras con las que se experimentó tenían un relleno no plano con respecto a las aristas que lo marcan, y tal como vimos con Harris, esto afecta enormemente la eficacia del algoritmo, por lo que, con imágenes con mayor definición y resolución este algoritmo potencia su usabilidad considerablemente.

BIBLIOGRAFÍA

colaboradores de Wikipedia. (2019, 19 diciembre). Esquina. Wikipedia, la enciclopedia libre.
<https://es.wikipedia.org/wiki/Esquina>

Detección de Esquinas y Vértices. Universidad Politécnica de Madrid. Departamento de Automática, Ingeniería Electrónica e Informática Industrial. José María Sebastián
<http://www.dfists.ua.es/tavarca/ponencias/esquinas.pdf>

S. Domínguez. (s. f.). Técnicas Avanzadas de Visión por Computador. UPM-DISAM. Recuperado 14 de noviembre de 2020, de
<https://www.cartagena99.com/recursos/alumnos/apuntes/CaracteristicasPuntuales.pdf>

K. Mikolajczyk, C. Schmid (2005) A performance evaluation on local descriptors, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 27, N. 10, pp. 1615-1630, octubre 2005

Apuntes tomados en clase