UNIVERDIDAD VERACRUZANA

INSTITUTO DE INVESTIGACIONES EN INTELIGENCIA ARTIFICIAL

VISIÓN ARTIFICIAL

HOMEWORK 6

LIBRO: DIGITAL IMAGE PROCESSING

AUTOR: RAFAEL C. GONZALEZ

GRADIENT OPERATOR

ALUMNO: JOSÉ DAVID VELAZCO MUÑOZ

PROFESOR: DR. HECTOR ACOSTA MESA

XALAPA, VERACRUZ

28 DE MARZO 2023

Objetive

- Develop a program to compute gradient vectors using Sobel operator over the image fig3.5(a).

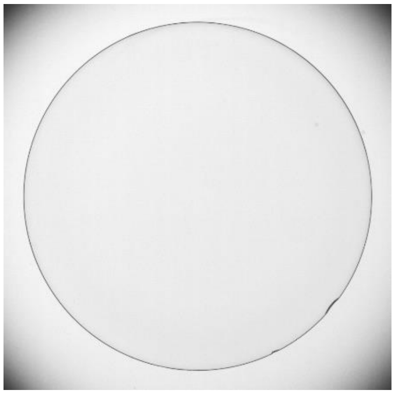


Figure 3.45(a)

- Display Gx, Gy, ∇f, and Ɵ.
- Display the vectors (magnitude and direccion) as an optic flow arrows perpendicular to the edge (use matlab command quiver(Gx,Gy)).

Methodology

Most edge detection methods work on the assumption that the edge occurs where there is a discontinuity in the intensity function or a very steep intensity gradient in the image.
Using this assumption, if one takes the derivative of the intensity value across the image and find points where the derivative is maximum, then the edge could be located. The gradient is a vector, whose components measure how rapid pixel value are changing with distance in the x and y direction.
Use the Sobel Kernel [1]:

$$\nabla x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \qquad \nabla y = \begin{bmatrix} -1 & -2 & 1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

For each pixel in the input image, compute the following:

$$\mathrm{G}x = \sum_{i=-1}^{1} \sum_{j=-1}^{1} f(x+i, y+j) * \nabla x(i,j) \qquad \mathrm{G}y = \sum_{i=-1}^{1} \sum_{j=-1}^{1} f(x+i, y+j) * \nabla y(i,j)$$

The magnitude of the gradient vector represents the strength of the edge or contour at that pixel, while the direction of the gradient vector indicates the orientation of the edge or contour.
The magnitude can be computed using the following formula:

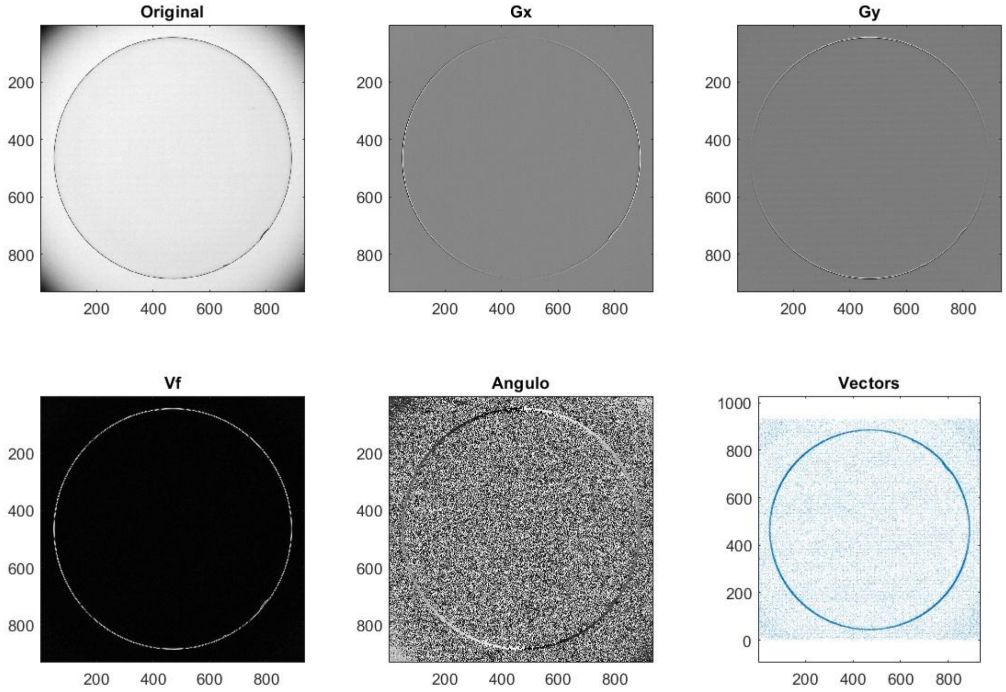
$$\nabla f = \sqrt{Gx^2 + Gy^2}$$

The direction of the gradient vector can be calculated using the atan2 function, which is a four-quadrant inverse tangent function that takes two arguments: the y-component and the x-component of the gradient vector [2].
In the context of image processing, the x-component and y-component of the gradient vector are obtained by applying the Sobel operator or other gradient filters to the image. Once the x and y components of the gradient are obtained, the gradient direction can be calculated as follows:

$$\theta = \tan^{-1} \frac{Gy}{Gx}$$

Results

The resulting gradient vector image shows the direction and magnitude of the gradient at each pixel in the image. This information can be used for many tasks, such as edge detection, object detection, and feature extraction.



Conclusion

In conclusion, the Sobel operator is a widely used technique for computing image gradients. By convolving the image with a pair of 3x3 kernels, it can efficiently compute the gradient values in both the x and y directions. This project can be used to extract important features from the image, such as edges, corners, and other regions of interest. The Sobel operator is particularly useful in applications such as computer vision, where it is often used as a preprocessing step for tasks such as object detection, image segmentation, and feature extraction. Overall, the Sobel operator is a powerful tool for analyzing images and extracting useful information from them.

Annex: Code

```
clear all;
% Detecta lineas horizontales
Gy=[1,1,1;0,0,0;-1,-1,-1];
%Detecta lineas verticales
Gx=[-1,0,1;-1,0,1;-1,0,1];
% Read image
I=double(imread('Fig3.45(a).jpg'));
% Determinate rows and columns
[rows,cols]=size(I);
% Create matix about Gx
g=zeros(rows,cols);
% Create matix about Gy
g2=zeros(rows,cols);
vf=zeros(rows,cols);
for i=2:rows-1
    for j=2:cols-1
        i2=I(i-1:i+1,j-1:j+1);
        g(i,j)=sum(sum(Gx.*i2));
        g2(i,j)=sum(sum(Gy.*i2));
    end
end
% The magnitude of this vector is given by
vf=(g.^2+g2.^2).^(1/2);
% Gradient direction
o=atan(g2./g);
% magnitude and direction
% Show image
figure()
subplot(2,3,1), imagesc(I),title(['Original']);
subplot(2,3,2), imagesc(g),title(['Gx']);
subplot(2,3,3), imagesc(g2),title(['Gy']);
subplot(2,3,4), imagesc(vf),title(['Vf']);
subplot(2,3,5), imagesc(o),title(['Angulo']);
subplot(2,3,6), quiver(g,g2),axis equal,title(['Vectors']);
colormap('gray');
```

References

- [1] Vincent, O. R., & Folorunso, O. (2009, June). A descriptive algorithm for sobel image edge detection. In *Proceedings of informing science & IT education conference (InSITE)* (Vol. 40, pp. 97-107).
- [2]Dharampal, V. M. (2015). Methods of image edge detection: A review. *J. Electr. Electron. Syst*, *4*(2), 2332-0796.