

# Edge detection

Jaskaran Singh

May 14, 2024

Edge detection includes a variety of mathematical methods that aim at identifying edges, defined as curves in a digital image at which the image brightness changes sharply or, more formally, has discontinuities. We aim to write a code that will detect the edges. We will be using the Sobel filter for edge detection.

We will use two kernels for horizontal and vertical edge detection, which are:

$$\begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -1 \\ 1 & 0 & -1 \end{bmatrix}, \text{ for horizontal edges}$$

and

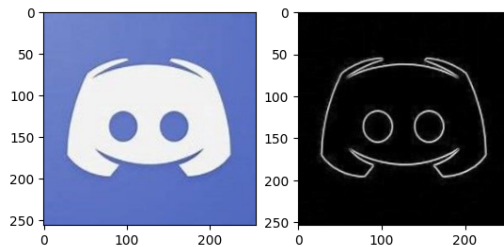
$$\begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}, \text{ for vertical edges}$$

And matrix formed after convolution are  $G_x$ ,  $G_y$

We define final matrix  $G = \sqrt{G_x^2 + G_y^2}$

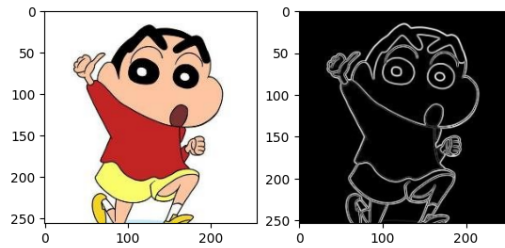
## 0.0.1 Results:

1.



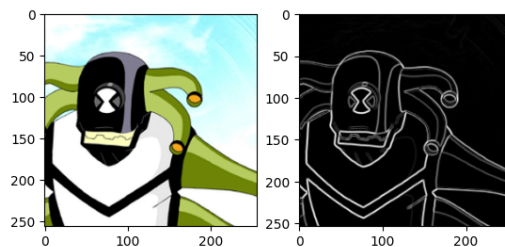
For the first example, I have used an image with solid colours. The operator picks all the edges in the image.

2.



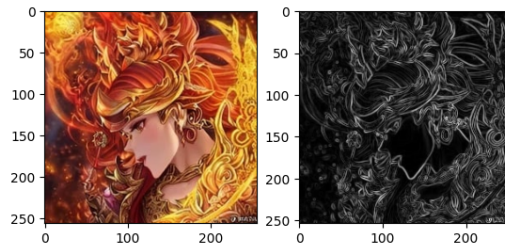
Slightly more shapes than the last image used. The sobel operator picks up edges in this one as well.

3.



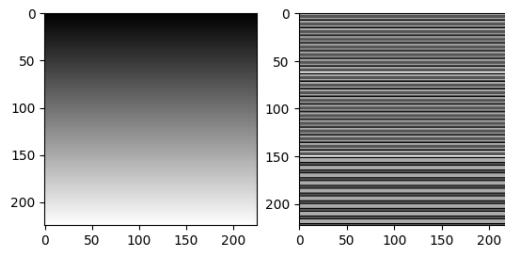
Notice that in this image, the sobel operator traces out lines which are technically not edges, these are lines in the image which represent a reflection from light. However, these are viewed as edges by the operator as the sobel operator picks up edges based on **change in value of intensity/rgb values of pixels**.

4.



This is a highly detailed image, it would be difficult to figure out the different kinds of edges in the image. The edges along the face are easily detected, but those in the bottom right feel a bit messed up. This is not only because of the details, but also due to lots of colour change.

5.



This is now an image without any edge, but with a gradient. We see that the output image seems nothing related to our original image. This is because, as said before the sobel operator picks up change in rgb value of the pixels. The change in colour is detected in the output image.