DAA ASSIGNMENT

BISWAJIT DAS

CSM21051

A. THE ALGORITHM:

1: start

2: Read the image and store in 2D array

4: For each rgb value P(i,j) of image 2D array

4.1: Calculate diagonal gradient

gradient_cross = {P(i,j)-P(i+1,j+1)} + {P(i,j+1)-P(i+1,j)}

4.2: Calculate horizontal and vertical gradient

gradient_plus = {P(i,j)-P(i,j+1)} + {P(i,j)-P(i+1,j)}

4.3: if gradient_cross or gradient_plus is none zero than EDGE pixel found and change the color to red.

4.4 else keep the same as image pixel value.

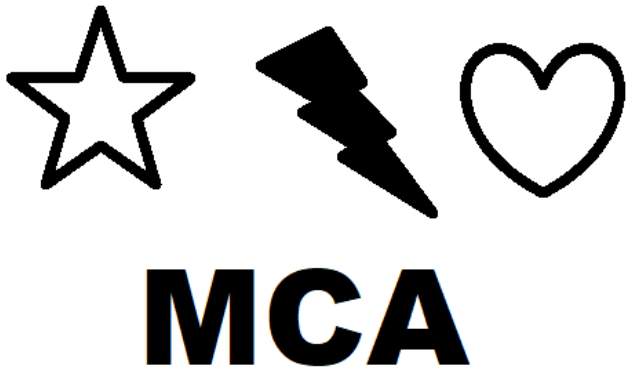
5. Save the new generate 2D array as Image.

6. Stop.

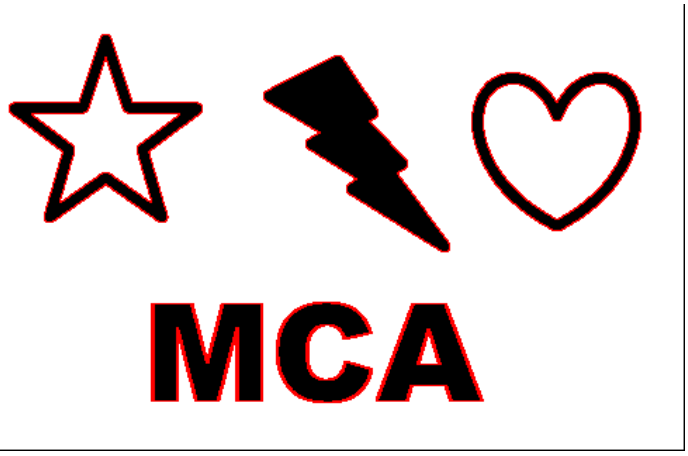
Code Snippet:

```
int[][] GImg = new int[width][height];

for (int i = 0; i < width -1 ; i++) {
    for (int j = 0; j < height-1 ; j++) {

        int p1 = PImg[i][j];
        int p2 = PImg[i][j + 1];
        int p3 = PImg[i+1][j];
        int p4 = PImg[i+1][j+1];
        int gradient_cross = (p1-p4) + (p2-p3);
        int gradient_plus = (p1-p2) + (p1-p3);
        if(gradient_cross!=0 || gradient_plus!=0)
            GImg[i][j]=16711680;
         else
            GImg[i][j]=PImg[i][j];
       }

}
```

C. Input:



Output:



D. **Time Complexity**: (width * height) of the input image.

-> O(n^2).