Tugas Pengolahan Citra

Tugas 6

Nama : Calvin Fadhil Mahendra

NIM : 09021381722115

Kelas : Teknik Informatika Bilingual A 5

Source Code:

```
<
import java.awt.*;
import java.io.File;
import java.util.ArrayList;
import java.util.HashMap;
import java.awt.BorderLayout;
import java.awt.Color;
import java.awt.Paint;
import java.awt.event.ActionEvent;
import java.awt.image.BufferedImage;
import java.awt.image.Raster;
import java.io.IOException;
import javax.imageio.ImageIO;
import javax.swing.AbstractAction;
import javax.swing.ImageIcon;
import javax.swing.JCheckBox;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JPanel;

public class EdgeDetect extends JFrame {
   BufferedImage gambar,gambar2,gambar3,gambar4 ,gambar5;
   int width,height;
   ImageIcon GambarAsli = new ImageIcon("D:\\Program Calvin\\Hasilgbr\\Cat.jpg");
   ImageIcon GambarGray = new ImageIcon("D:\\Program Calvin\\Hasilgbr\\camera.png");
   ImageIcon GambarHisto = new ImageIcon("D:\\Program Calvin\\Hasilgbr\\CatHisto.jpg");

   EdgeDetect(String img){
      try{
         File file = new File(img);
         gambar = ImageIO.read(file);
         width = gambar.getWidth();
         height = gambar.getHeight();

      }catch(Exception e){
```

```
            System.out.println("Error : "+e);
        }
        GridLayout layout = new GridLayout(1,5,5,5);
        setLayout(layout);

        JPanel p1 = new JPanel(new GridLayout(1,5));

        p1.add(new JLabel(GambarAsli));
        p1.add(new JLabel(GambarGray));
        p1.add(new JLabel(GambarHisto));

        JPanel p2 = new JPanel(new GridLayout(1,5));
        p2.add(p1,BorderLayout.NORTH);

        add(p1,BorderLayout.NORTH);
        add(p2,BorderLayout.SOUTH);


    }

    public void scalingGrey(){
        for (int y = 0; y < height; y++) {
            for (int x = 0; x < width; x++) {
                int p = gambar.getRGB(x,y);
                int a = (p>>24)&0xff;
                int r = (p>>16)&0xff;
                int g = (p>>8)&0xff;
                int b = p&0xff;
                int avg = (r+g+b)/3;
                p = (a<<24) | (avg<<16) | (avg<<8) | avg;
                gambar.setRGB(x, y, p);
            }}

    }

    public void scalingScharr(){
        int[][] filter = {{3,0,-3},
                {10,0,-10},
                {3,0,-3}};

        gambar2 = new BufferedImage(width+(filter.length-1), height+(filter[0].length-
1),BufferedImage.TYPE_INT_RGB);
        int filterW = filter.length;
        int filterH = filter[0].length;

        for (int y=0; y<height;y++){
            for(int x=0; x<width; x++){
                int rTemp = 0;
```

```
int gTemp = 0;
int bTemp = 0;

for (int i=0; i<filter.length;i++){
   for (int j=0; j<filter[i].length;j++){
      int imgX = (x - filterW/2 +j +width) % width;
      int imgY = (y - filterH/2 +i +height) % height;

      int p = gambar.getRGB(imgX,imgY);
      int r = p >> 16 & 0xff;
      int g = p >> 8 & 0xff;
      int b = p & 0xff;

      rTemp += (r*filter[j][i]);
      gTemp += (g*filter[j][i]);
      bTemp += (b*filter[j][i]);

   }
}

if(rTemp>255){
   rTemp = 255;
}else{
   if(rTemp<0){
      rTemp=0;
   }else{
      rTemp = rTemp;
   }
}

if(gTemp>255){
   gTemp = 255;
}else{
   if(gTemp<0){
      gTemp=0;
   }else{
      gTemp = gTemp;
   }
}

if(bTemp>255){
   bTemp = 255;
}else{
   if(bTemp<0){
      bTemp=0;
   }else{
      bTemp = bTemp;
   }
```

```
            }
            gambar2.setRGB(x,y,new Color((int)rTemp,(int)gTemp,(int)bTemp).getRGB());
          }
       }
    }

    public void scalingSobel(){
       int[][] filter = {{1,0,-1},
            {2,0,-2},
            {1,0,-1}};

       gambar3 = new BufferedImage(width+(filter.length-1), height+(filter[0].length-
1),BufferedImage.TYPE_INT_RGB);
       int filterW = filter.length;
       int filterH = filter[0].length;

       for (int y=0; y<height;y++){
          for(int x=0; x<width; x++){
             int rTemp = 0;
             int gTemp = 0;
             int bTemp = 0;

             for (int i=0; i<filter.length;i++){
                for (int j=0; j<filter[i].length;j++){
                   int imgX = (x - filterW/2 +j +width) % width;
                   int imgY = (y - filterH/2 +i +height) % height;

                   int p = gambar.getRGB(imgX,imgY);
                   int r = p >> 16 & 0xff;
                   int g = p >> 8 &
>
```

Screenshot:

<

>