

```

package cap2;

import javax.imageio.ImageIO;
import javax.swing.*;
import javax.swing.UIManager.LookAndFeelInfo;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.image.BufferedImage;
import java.io.File;
import java.io.IOException;

public class FiltragemEspacialJanelamento implements ActionListener {

    private JButton btnAcao = null;
    private JButton btnAcao2 = null;
    private BufferedImage imagemEntrada = null;
    private JFrame principal = null;
    private JLabel labImagem = null;
    private Container contentPane = null;
    BufferedImage imagemSaida = null;

    public void run() throws IOException {

        checkNimbus();

        principal = new JFrame("Lendo uma imagem PNG");
        principal.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

        String path =
"C:\\\\Users\\\\alunoudi\\\\Downloads\\\\lena_gray_256.png";
        File file = new File(path);
        imagemEntrada = ImageIO.read(file);

        ImageIcon icone = new ImageIcon(imagemEntrada);
        labImagem = new JLabel(icone);

        String infoImagem = "Dimensões: " + imagemEntrada.getWidth() +
"x" + imagemEntrada.getHeight() + "Bandas: " +
imagemEntrada.getRaster().getNumBands();

        contentPane = principal.getContentPane();
        contentPane.setLayout(new BorderLayout());
        contentPane.add(new JScrollPane(labImagem),
BorderLayout.CENTER);
        contentPane.add(new JLabel(infoImagem), BorderLayout.NORTH);

        JPanel painel = new JPanel();
        btnAcao = new JButton("Efeito Especial");
        btnAcao.addActionListener(this);
        btnAcao2 = new JButton("Salvar Imagem");
        btnAcao2.addActionListener(this);
        painel.add(btnAcao);
        painel.add(btnAcao2);
        contentPane.add(painel, BorderLayout.SOUTH);

        principal.setSize(imagemEntrada.getWidth() + 40,
imagemEntrada.getHeight() + 100);
        principal.setVisible(true);
        principal.setLocationRelativeTo(null);
    }
}

```

```

private static void checkNimbus() {
    try {
        for (LookAndFeelInfo info :
            UIManager.getInstalledLookAndFeels()) {
            if ("Nimbus".equals(info.getName())) {
                UIManager.setLookAndFeel(info.getClassName());
                break;
            }
        }
    } catch (Exception e) {
        JOptionPane.showMessageDialog(null, "Nimbus not
available!");
    }
}

@Override
public void actionPerformed(ActionEvent e) {

    if(e.getSource() == btnAcao) {

        double fator = 3.0;

        int larguraEntrada = imagemEntrada.getWidth();
        int alturaEntrada = imagemEntrada.getHeight();

        int larguraSaida = (int) (larguraEntrada * fator);
        int alturaSaida = (int) (alturaEntrada * fator);

        imagemSaida = new BufferedImage(larguraSaida, alturaSaida,
            BufferedImage.TYPE_4BYTE_ABGR);

        for (int x = 0; x < larguraSaida; x++) {
            for (int y = 0; y < alturaSaida; y++) {
                int count = 0;
                int xEntrada = (int) (x / fator);
                int yEntrada = (int) (y / fator);
                int tom1 = 0;
                if(xEntrada > 0 && yEntrada > 0) {
                    count++;
                    tom1 = new Color(imagemEntrada.getRGB(xEntrada-1,
yEntrada-1)).getGreen();
                }
                int tom2 = 0;
                if(yEntrada > 0) {
                    count++;
                    tom2 = new Color(imagemEntrada.getRGB(xEntrada,
yEntrada-1)).getGreen();
                }
                int tom3 = 0;
                if(xEntrada < 255 && yEntrada > 0) {
                    count++;
                    tom3 = new Color(imagemEntrada.getRGB(xEntrada+1,
yEntrada-1)).getGreen();
                }
                int tom4 = 0;
                if(xEntrada > 0){
                    count++;
                    tom4 = new Color(imagemEntrada.getRGB(xEntrada-1,
yEntrada)).getGreen();
                }
            }
        }
    }
}

```

```

        count++;
        int tom5 = new Color(imagemEntrada.getRGB(xEntrada,
yEntrada)).getGreen();
        int tom6 = 0;
        if(xEntrada < 255) {
            count++;
            tom6 = new Color(imagemEntrada.getRGB(xEntrada+1,
yEntrada)).getGreen();
        }
        int tom7 = 0;
        if(xEntrada > 0 && yEntrada < 255) {
            count++;
            tom7 = new Color(imagemEntrada.getRGB(xEntrada-1,
yEntrada+1)).getGreen();
        }
        int tom8 = 0;
        if(yEntrada < 255) {
            count++;
            tom8 = new Color(imagemEntrada.getRGB(xEntrada,
yEntrada+1)).getGreen();
        }
        int tom9 = 0;
        if(xEntrada < 255 && yEntrada < 255) {
            count++;
            tom9 = new Color(imagemEntrada.getRGB(xEntrada+1,
yEntrada+1)).getGreen();
        }
        int total = tom1 + tom2 + tom3 + tom4 + tom5 + tom6 +
tom7 + tom8 +tom9;
        int media = total / count;

        imagemSaida.setRGB(x, y, new Color(media, media,
media).getRGB()); // pixel de saída
    }
}

 ImageIcon icone = new ImageIcon(imagemSaida);
 JLabel labImagem2 = new JLabel(icone);
 contentPane.add(new JScrollPane(labImagem2),
BorderLayout.CENTER);

 principal.setSize(imagemSaida.getWidth() + 40,
imagemSaida.getHeight() + 100);
 principal.setVisible(true);
 principal.setLocationRelativeTo(null);
 principal.repaint();

} else if(e.getSource() == btnAcao2) {

    final JFileChooser fc = new JFileChooser();
    int returnVal = fc.showSaveDialog(principal);
    if (returnVal == JFileChooser.APPROVE_OPTION) {
        File file = fc.getSelectedFile();
        try {
            ImageIO.write(imagemSaida, "PNG", file);
            JOptionPane.showMessageDialog(principal, "OK!");
        } catch (IOException e1) {
            JOptionPane.showMessageDialog(principal, "Erro:
"+e1.getMessage());
        }
    } else {

```

```
        //  
    }  
}  
  
public static void main(String[] args) throws IOException {  
    new FiltragemEspacialJanelamento().run();  
}
```