

PRAKTIKUM 12

Segmentasi Citra dengan metode Region Growing

12.1. Tujuan :

Mahasiswa mengetahui cara membuat program segmentasi citra dengan menggunakan metode region growing pada sebuah citra.

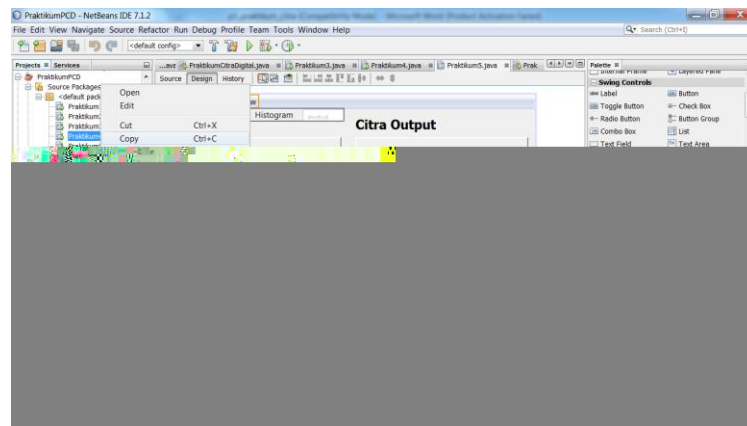
12.2. Dasar Teori :

Segmentasi merupakan proses mempartisi citra menjadi beberapa daerah atau objek, Segmentasi citra pada umumnya berdasar pada sifat discontinuity atau similarity dari intensitas piksel. Pendekatan discontinuity mempartisi citra bila terdapat perubahan intensitas secara tiba-tiba (edge based), sedangkan Pendekatan similarity mempartisi citra menjadi daerah-daerah yang memiliki kesamaan sifat tertentu (region based), salah satunya adalah metode segmentasi region growing, kekurangan dari metode region growing adalah belum tentu menghasilkan wilayah-wilayah yang bersambungan. Prosedur region growing memerlukan criteria of uniformity (kriteria), Memerlukan penyebaran seeds atau dapat juga dengan pendekatan scan line. Berikut adalah algoritma Region Growing:

- Tentukan beberapa piksel *seed*.
 - Seed bisa ditentukan manual atau secara random
- Untuk setiap piksel seed, lihat 4 atau 8 tetangganya, jika kriterianya sama (kriteria bisa berupa perbedaan keabuan dengan seed, dll) maka tetangga tersebut bisa dianggap berada dalam 1 region/daerah dengan piksel seed.
- Teruskan proses dengan mengecek tetangga dari tetangga yang sudah kita cek, dst.
- Tidak bisa hanya digunakan kriteria saja, tanpa melihat konektivitas ketetanggaan, karena bisa tidak membentuk daerah
- Stopping rule kadang tidak mencakup semua kemungkinan sehingga pada akhir region growing ada piksel yang belum dicek sama sekali.

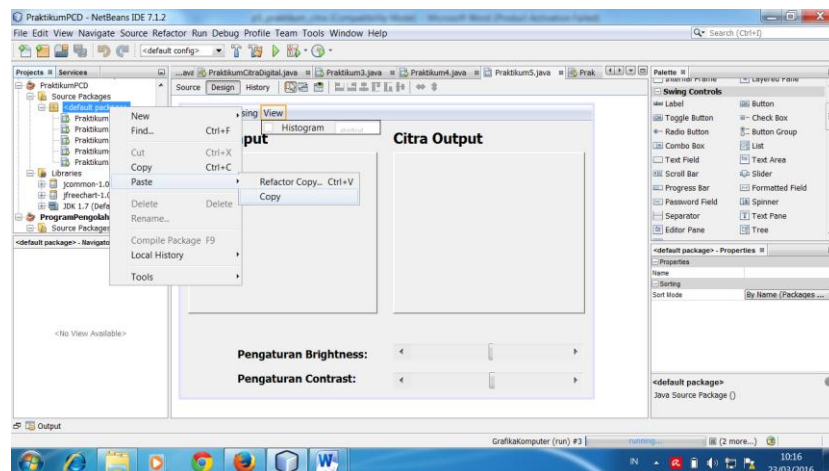
11.4. Langkah Praktikum :

- 1). Copy class JFrame Form pada praktikum sebelumnya dengan cara klik kanan pada folder Source Package Praktikum11 lalu pilih Copy seperti tampak pada gambar 12.1 berikut:



Gambar 12.1. Copy class JFrame Form

- 2). Kemudian klik kanan pada folder source packages pada <default Packages>, kemudian pilih Paste kemudian pilih Copy, seperti tampak pada gambar 12.2.



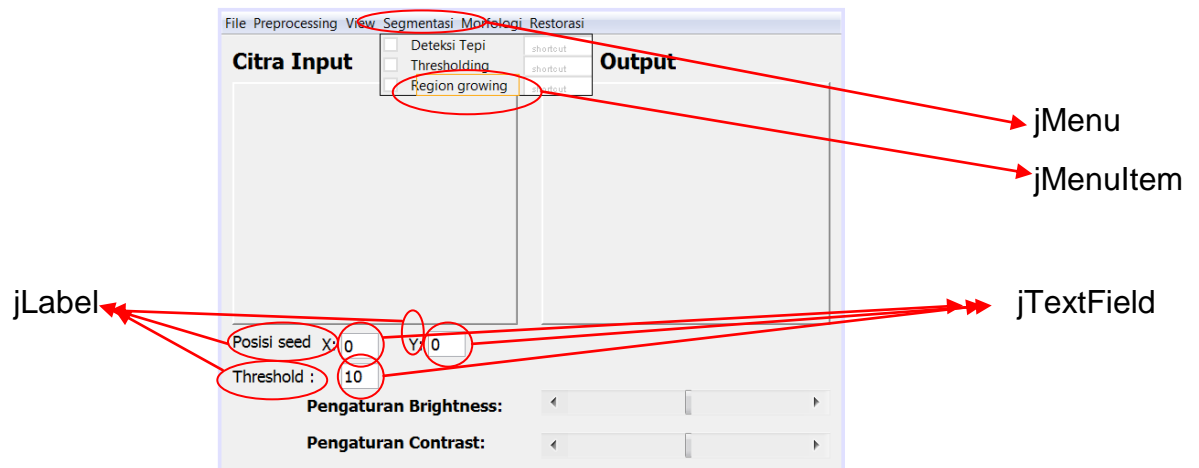
Gambar 12.2. Paste ke class JFrame Form baru

- 3). Rubah nama class menjadi Praktikum12 kemudian centang pilihan Copy Without Refactoring. Kemudian klik OK. Seperti tampak pada gambar 12.3. kemudian rubah nama class dari Praktikum11_1 menjadi Praktikum12.



Gambar 12.3. Jendela Dialog Copy Class

- 4). Klik Design, kemudian tambahkan sebuah JMenuItem baru pada Menu Segmentasi tersebut kemudian rubah text nya menjadi region growing. Serta tambahkan 3 buah jTextField serta 3 buah JLabel dan ubah textnya masing-masing menjadi posisi seed x , posisi seed y dan threshold. Seperti tampak dalam gambar 12.4 berikut:



Gambar 12.4. Design Form

- 5). Tambahkan Kode metode untuk proses segmentasi region growing pada bagian Source Praktikum12 pada event method JMenuItem region growing sebagai berikut.

```
private void jMenuItem9ActionPerformed(java.awt.event.ActionEvent evt) {
    BufferedImage grayscale = regiongrowing(sumber);
    int x = jLabel2.getWidth();
    int y = jLabel2.getHeight();
    ImageIcon imageIcon = new ImageIcon(resize(grayscale, x, y));
    jLabel2.setIcon(imageIcon);
}
```

- 7). Tambahkan method baru untuk proses region growing dengan kode sebagai berikut:

```
public BufferedImage regiongrowing(String sumber) {
    int pixel=0;
    int a1=0, a2=0;
    int thr=0, ada2=0;
    int h= 0 ;
}
```

```
BufferedImage prosesGambar;
BufferedImage loadIng = loadImage(sumber);
ukuranX = loadIng.getWidth();
ukuranY = loadIng.getHeight();
int Bufferku2 [][]= new int [ukuranX][ukuranY];
int Bufferku3 [][]= new int [ukuranX][ukuranY];
prosesGambar = new BufferedImage(ukuranX, ukuranY,
    BufferedImage.TYPE_BYTE_GRAY);
Graphics g = prosesGambar.getGraphics();
g.drawImage(loadIng, 0, 0, null);
WritableRaster raster = prosesGambar.getRaster();

//inisialisasi gambar2 dan gambar3
for (int i=0; i<ukuranX ; i++) {
    for (int j=0; j<ukuranY ; j++) {
        Bufferku2[i][j]=255;
        Bufferku3[i][j]=0;
    }
}

//mencatat posisi awal region dan dimapping di gambar 3
seedx = Integer.parseInt(jTextField1.getText()); // posisi awal x region
seedy = Integer.parseInt(jTextField2.getText()); // posisi awal y region
Bufferku3[seedx][seedy]=50; // mapping posisi awal di gambar 3
ada2 = 1;
thr = Integer.parseInt(jTextField3.getText());
int rgbs = loadIng.getRGB(seedx, seedy);
int hijaugs = (rgbs >> 8) & 0xff;
a1 = hijaugs - thr;
a2 = hijaugs + thr;

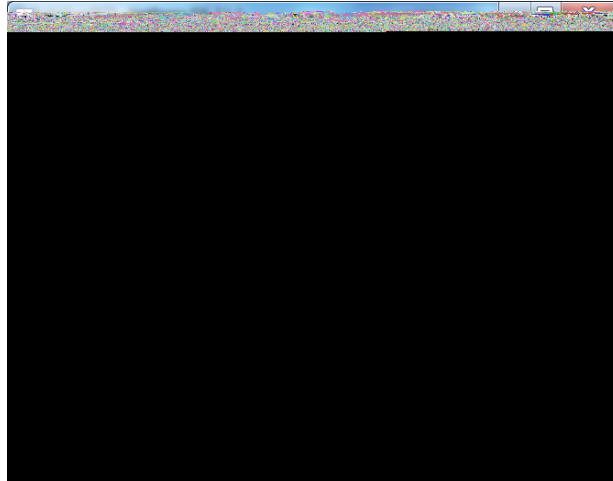
// letakkan disini code proses region growing
while (ada2 == 1){
    ada2=0;
    // cari pada gb3 apa ada yang bernilai 50
    for (int i=1; i<ukuranX ; i++) {
        for (int j=1; j<ukuranY ; j++) {
            // ambil pixel pada posisi i,j
            if (i==0) {Bufferku3[i][j]=1;}
            if (i==ukuranY) {Bufferku3[i][j]=1;}
            pixel = Bufferku3[i][j]; // nilai pixel pada posisi i,j
            // uji apa pixel bernilai 50
            if (pixel == 50) {
                ada2 = 1 ;
                int rgb = loadIng.getRGB(i, j-1);
                int hijaug = (rgb >> 8) & 0xff;
                if ((hijaug >= a1) && (hijaug <= a2) &&
                    (Bufferku3[i][j-1] != 200)) {
                    Bufferku3[i][j-1]=50;
                }
                rgb = loadIng.getRGB(i, j+1);
                hijaug = (rgb >> 8) & 0xff;
                if ((hijaug >= a1) && (hijaug <= a2) &&
                    (Bufferku3[i][j+1] != 200)) {
                    Bufferku3[i][j+1]=50;
                }
                rgb = loadIng.getRGB(i-1, j);
```

```

        hijau = (rgb >> 8) & 0xff;
        if ((hijau >= a1) && (hijau <= a2) &&
            (Bufferku3[i-1][j] != 200)) {
            Bufferku3[i-1][j]=50;
        }
        rgb = loadIng.getRGB(i-1, j+1);
        hijau = (rgb >> 8) & 0xff;
        if ((hijau >= a1) && (hijau <= a2) &&
            (Bufferku3[i-1][j+1] != 200)) {
            Bufferku3[i-1][j+1]=50;
        }
        rgb = loadIng.getRGB(i-1, j-1);
        hijau = (rgb >> 8) & 0xff;
        if ((hijau >= a1) && (hijau <= a2) &&
            (Bufferku3[i-1][j-1] != 200)) {
            Bufferku3[i-1][j-1]=50;
        }
        rgb = loadIng.getRGB(i+1, j);
        hijau = (rgb >> 8) & 0xff;
        if ((hijau >= a1) && (hijau <= a2) &&
            (Bufferku3[i+1][j] != 200)) {
            Bufferku3[i+1][j]=50;
        }
        rgb = loadIng.getRGB(i+1, j+1);
        hijau = (rgb >> 8) & 0xff;
        if ((hijau >= a1) && (hijau <= a2) &&
            (Bufferku3[i+1][j+1] != 200)) {
            Bufferku3[i+1][j+1]=50;
        }
        rgb = loadIng.getRGB(i+1, j-1);
        hijau = (rgb >> 8) & 0xff;
        if ((hijau >= a1) && (hijau <= a2) &&
            (Bufferku3[i+1][j-1] != 200)) {
            Bufferku3[i+1][j-1]=50;
        }
        // copy kan data posisi i,j di gb1 ke posis i,j di gb2 dan
        // set 200 pada gb3
        rgb = loadIng.getRGB(i, j);
        hijau = (rgb >> 8) & 0xff;
        Bufferku2[i][j]=hijau;
        Bufferku3[i][j]=200;
    }
}
}
}
for (int i=0; i<ukuranX ; i++) {
    for (int j=0; j<ukuranY ; j++) {
        raster.setSample(i, j, 0, Bufferku2[i][j]);
    }
}
return prosesGambar;
}

```

8). Jalankan program sehingga tampilannya seperti gambar 12.5. berikut:



Gambar 12.5. Tampilan aplikasi hasil proses region growing