

## PEMBUATAN APLIKASI DETEKSI DINI KANKER SERVIKS

Onny Marleen<sup>1\*</sup>, Ricky Agus Tjiptanata<sup>1</sup>, Widiastuti<sup>2</sup>

Jurusan Sistem Informasi, Fakultas Ilmu Komputer & TI, Universitas Gunadarma

Jl. Margonda Raya 100, Depok 16424

\*Email: onny\_marleen@staff.gunadarma.ac.id

### Abstrak

*Di Indonesia, kaum wanita yang mengidap kanker cukup tinggi terutama pada kanker serviks. Kanker serviks atau kanker leher rahim adalah kanker pada sistem reproduksi wanita. Kanker serviks sampai saat ini merupakan salah satu penyebab kematian kaum wanita yang cukup tinggi, baik di negara-negara maju maupun negara berkembang seperti Indonesia. Bahkan di Indonesia, kanker serviks merupakan kanker nomor satu yang umum diderita wanita Indonesia. Mendeteksi kanker serviks dapat dilakukan dengan menggunakan metode segmentasi warna dan tekstur. Deteksi ini meliputi 6 tahapan yaitu cropping data, konversi warna dari RGB ke HSV, penghilangan pantulan cahaya, segmentasi warna, pencarian area lesi, dan segmentasi tekstur. Aplikasi dibuat dengan menggunakan perangkat lunak Matlab versi 7.0. Aplikasi yang dihasilkan dapat memberikan informasi ada atau tidaknya area lesi sebagai deteksi dini adanya kanker serviks beserta tingkat kekasaran area lesi sebagai informasi tingkat keparahan kanker serviks.*

**Kata kunci:** aplikasi, deteksi dini, kanker serviks, MatLab, segmentasi

## 1. PENDAHULUAN

Kanker leher rahim atau yang dikenal dengan kanker serviks merupakan kanker primer yang menyerang perempuan, termasuk di negara-negara berkembang seperti Indonesia. Penyebab kematian perempuan terbanyak di Indonesia, salah satunya adalah kanker serviks. Kanker serviks atau kanker leher rahim adalah kanker yang terjadi pada serviks uteri yaitu suatu daerah pada organ reproduksi wanita yang merupakan pintu masuk ke arah rahim yang terletak antara rahim (uterus) dengan liang senggama (vagina). Kanker serviks merupakan kanker primer yang berasal dari serviks.

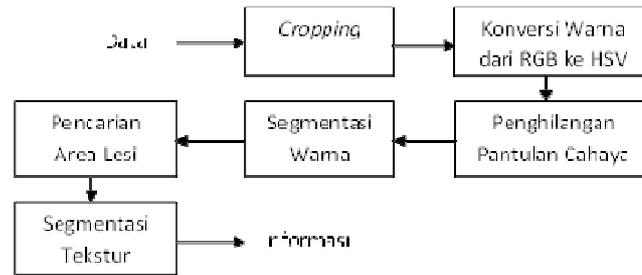
Saat ini perpaduan antara perkembangan ilmu pengetahuan dan teknologi sudah banyak dilakukan orang khususnya dalam bidang pengolahan citra. Pengolahan citra adalah pemrosesan citra, Pengolahan citra saat ini banyak digunakan diberbagai bidang kehidupan manusia, salah satu contohnya adalah dalam bidang kedokteran. Pengolahan citra merupakan proses pengolahan dan analisis citra yang banyak melibatkan persepsi visual salah satunya dalam bentuk tekstur, demikian pula dalam bidang kedokteran sering mengandalkan proses visualisasi yang lebih dikenal dengan pencitraan medis (*medical imaging*). Pencitraan sering juga digunakan untuk mendeteksi kanker serviks.

Pembuatan aplikasi deteksi dini kanker serviks didasarkan pada penelitian sebelumnya dengan menggunakan metode segmentasi warna dan segmentasi tekstur.

## 2. METODOLOGI

Kebutuhan software yang digunakan adalah sistem operasi Windows Xp dan MatLab versi 7.0 sebagai pengolah data dan pembuatan aplikasi. Spesifikasi perangkat keras adalah processor Intel Core i3-2330 M, CPU 2,20 GHz, RAM 4.00 GB dan system type 64 bit.

Langkah-langkah yang dilakukan dalam proses pembuatan aplikasi deteksi dini kanker serviks terdiri atas 6 langkah proses setelah pengumpulan data citra yang disebut juga dengan istilah sevigram yaitu: proses cropping karena tidak semua informasi dalam citra tersebut diambil, kemudian dilakukan proses konversi warna dari RGB ke HSV, proses penghilangan pantulan sinar dilakukan disebabkan sinar yang tampak pada sevigram dianggap sebagai derau yang dapat mengganggu proses pencarian lesi (lesi adalah kelainan jaringan yang mengindikasikan adanya kanker serviks), proses segmentasi warna yang merupakan tahapan proses pencarian area lesi berdasarkan ciri warna pada lesi, dan terakhir segmentasi tekstur untuk melihat tingkat kekasaran permukaan area lesi. Tahapan ini terlihat pada gambar 1.



**Gambar 1. Tahapan proses pembuatan aplikasi**

### 3. HASIL DAN PEMBAHASAN

Algoritma dan pembuatan aplikasi dengan menggunakan MatLab [Wijaya, 2007] dibuat sesuai dengan urutan tahap deteksi dini kanker serviks berdasarkan segmentasi warna dan segmentasi tekstur tersaji pada subbab 3.1 hingga 3.7.

#### 3.1. Algoritma Cropping Data

Dilakukan langsung dengan melakukan click dan drag pada area yang akan digunakan dengan membuang informasi yang tidak diperlukan.

Codingan Proses Cropping Data:

```

function varargout = pushbutton1_Callback(hObject, eventdata, handles, varargin)
set(handles.informasi,'String','Cropping');
cropping=imcrop(handles.gambar);
handles.cropping = cropping;
axes(handles.axes2);
imshow(cropping);
guidata(hObject, handles);
  
```

#### 3.2. Algoritma Konversi Warna dari RGB ke HSV

Dimulai dari membaca citra digital hasil cropping, kemudian melakukan transformasi citra digital dari ruang warna RGB ke ruang warna HSV dan mengambil unsur saturation dari ruang warna HSV.

Codingan Proses Konversi Warna dari RGB ke HSV:

```

function pushbutton2_Callback(hObject, eventdata, handles)
set(handles.informasi,'String','Konversi RGB-HSV');
set(handles.contrastnya,'String','');
set(handles.homogeneitynya,'String','');
axes(handles.axes2);
HSV =rgb2hsv(handles.cropping);
SR = HSV(:,:,2);
imshow(SR);
handles.SR = SR;
handles.HSV = HSV;
guidata(hObject, handles);
  
```

#### 3.3. Algoritma Penghilangan Pantulan Cahaya

Menentukan nilai ambang pantulan sinar, berdasarkan penelitian didapat nilai kisaran antara 0,6181 hingga 0,70, kemudian mengambil area biner yang sesuai untuk menjadi kandidat pantulan sinar dan memberi batas pada area pantulan sinar. Langkah akhir adalah hilangkan area pantulan sinar [Onny, 2013].

Codingan Proses Penghilangan Pantulan Cahaya:

```

function pushbutton3_Callback(hObject, eventdata, handles)
set(handles.informasi,'String','Penghilangan Pantulan Cahaya');
angka_threshold=str2double(get(findobj(gcf,'Tag','angka_batasan1'),'String'));
[m,n]=size(handles.SR);
for brs=1:m
  
```

```

for kol=1:n
    if handles.SR(bris,kol)>=angka_threshold %0.7
        handles.SR(bris,kol) =1;
    else
        handles.SR(bris,kol) =0;
    end;
end;
end;
BW2 = bwareaopen(handles.SR, 5000);
handles.BW2 = BW2;
axes(handles.axes2);
imshow(BW2);
guidata(hObject, handles);

```

### 3.4. Algoritma Segmentasi Warna

Hasil dari penghilangan pantulan cahaya dilihat apakah masih ada area kosong atau tidak, jika masih ada area kosong, maka lakukan pengisian dengan intensitas piksel yang sama dengan tetangga hingga menghasilkan area lesi.

Codingan Proses Segmentasi Warna:

```

function pushbutton4_Callback(hObject, eventdata, handles)
set(handles.informasi,'String','Segmentasi Warna');
em = true(3);
bw3 = imclose(handles.BW2,em);
bw4=imfill(bw3,'holes');
I2=handles.HSV(:,:,2);
I2(~bw4)=0;
[m,n]=size(I2);
for brs=1:m
    for kol=1:n
        if I2(bris,kol)>=0.6181 && I2(bris,kol)<=0.70
            else
                I2(bris,kol) =0;
            end;
        end;
    end;
end;
BT = bwareaopen(I2, 500);
axes(handles.axes2);
imshow(BT);
handles.BT = BT
guidata(hObject, handles);

```

### 3.5. Algoritma Pencarian Area Lesi

Berikan batas tegas pada tepi area lesi yang didapat [Onny, 2012].

Codingan Proses Pencarian Area Lesi:

```

function pushbutton5_Callback(hObject, eventdata, handles)
set(handles.informasi,'String','Pencarian Area Lesi');
en=true(5);
bk = imclose(handles.BT,en);
bw4=imfill(bk,'holes');
axes(handles.axes2);
imshow(bw4);
handles.bw4 = bw4
guidata(hObject, handles);

```

### 3.6. Algoritma Segmentasi Tekstur

Menentukan nilai parameter jarak antar piksel yang bertetangga, kemudian menentukan nilai parameter sudut antar piksel dan tentukan ukuran citra area lesi yang digunakan untuk analisis tekstur untuk mencari nilai matriks kookurensi dan menghitung nilai homogenitas dan kontras hingga didapat citra hasil akhir [Onny, 2015].

**Codingan Proses Segmentasi Tekstur:**

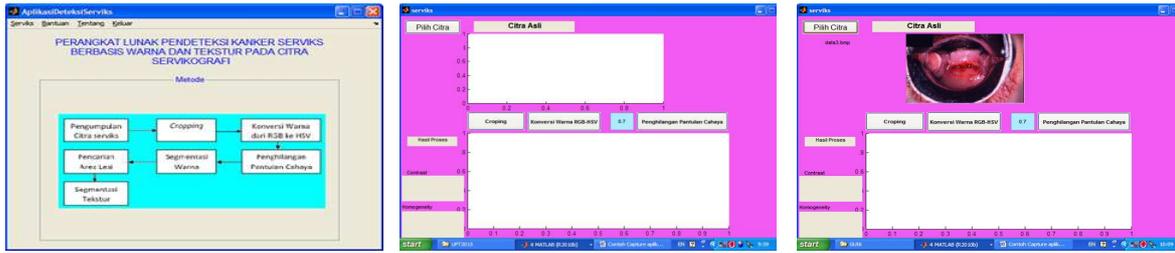
```

function pushbutton8_Callback(hObject, eventdata, handles)
set(handles.informasi,'String','Segmentasi Tekstur dan Citra Hasil Akhir');
[B,L,N,A] = bwboundaries(handles.bw4);
axes(handles.axes2);
imshow(handles.cropping);
hold on;
for k=1:length(B),
    if(~sum(A(k,:)))
        boundary = B{k};
        plot(boundary(:,2),...
            boundary(:,1),'g','LineWidth',2);
        for l=find(A(:,k))'
            boundary = B{l};
            plot(boundary(:,2),...
                boundary(:,1),'g','LineWidth',2);
        end
    end
end
end
BW=regionprops(handles.bw4, 'Area', 'Extrema');
A=[BW.Extrema];
[~,M]=size(A);
Xmax=0;
Ymax=0;
Xmin=1000;
Ymin=1000;
for i=1:M/2
    Ymax=floor(max(Ymax,max(BW(i).Extrema(:,1))))
    Xmax=floor(max(Xmax,max(BW(i).Extrema(:,2))))
    Ymin=floor(min(Ymin,min(BW(i).Extrema(:,1))))
    Xmin=floor(min(Xmin,min(BW(i).Extrema(:,2))))
end
I2=handles.HSV(Xmin:Xmax,Ymin:Ymax,2);
handles.Ymaxnya = Ymax;
handles.Xmaxnya = Xmax;
handles.Yminnya = Ymin;
handles.Xminnya = Xmin;
GLCM2 = graycomatrix(I2,'Offset',[0 4;-4 4;-4 0;-4 -4]);
stats = graycoprops(GLCM2,{'contrast','homogeneity','energy','correlation'})
dataout = zeros(1,length(stats)) % Preallocating data for structure
field = fieldnames(stats);
field
for i = 1:length(field)
    getfield(stats,field{i})
    if i == 1
        set(handles.contrastnya,'String',getfield(stats,field{i}));
    elseif i==4
        set(handles.homogeneitynya,'String',getfield(stats,field{i}));
    else
        end
    end
end
guidata(hObject, handles);

```

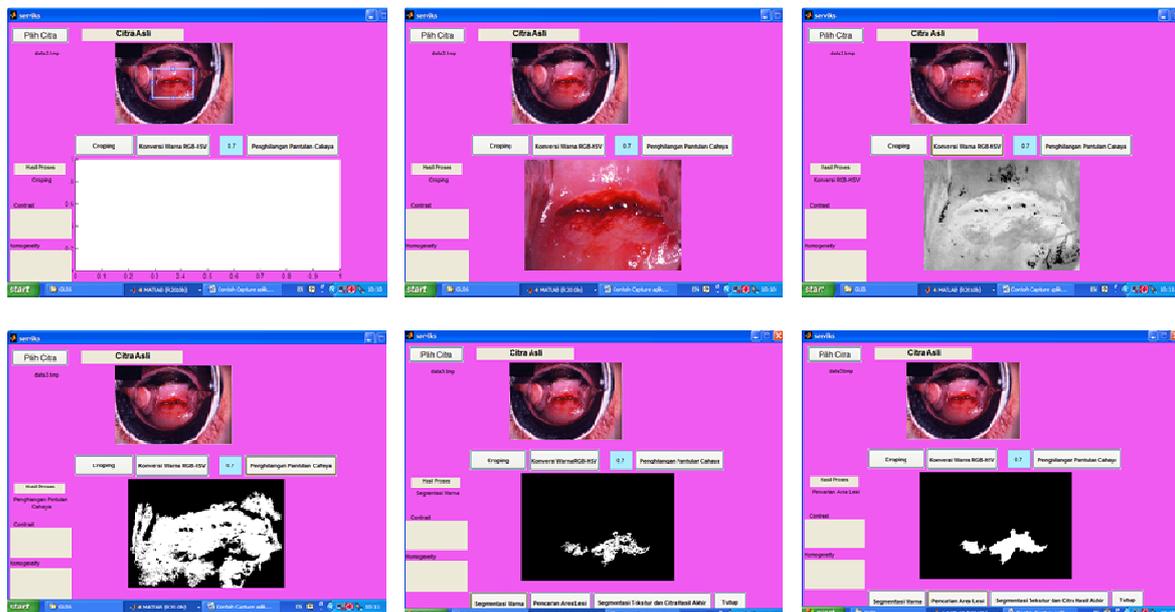
**3.7. Tampilan Aplikasi**

Berikut akan ditampilkan output awal aplikasi sebagai grafik user interface yang dibuat dengan MatLab [Aris, 2006].



Gambar 2. Tampilan awal aplikasi

Gambar 3 adalah tampilan aplikasi mulai dari proses cropping hingga proses akhir deteksi dini kanker serviks.



Gambar 3. Tampilan aplikasi proses deteksi dini kanker serviks

Output akhir dari aplikasi terlihat pada gambar 4.



Gambar 4. Tampilan hasil akhir aplikasi

#### 4. KESIMPULAN

Aplikasi deteksi dini kanker serviks ini dapat digunakan untuk membantu evaluator dalam menentukan ada tidaknya kanker serviks dan tingkat keparahan kanker serviks. Aplikasi ini akan membantu para pasien yang berada jauh dari para evaluator untuk mendapatkan informasi dengan lebih cepat.

Aplikasi ini dapat dikembangkan dengan menambahkan penggunaan database sehingga riwayat pasien tentang kanker serviks dapat disimpan dan dicari dengan mudah.

### **UCAPAN TERIMA KASIH**

Tim peneliti berterima kasih kepada Prof. Dr. E. S Margianti, SE., MM, selaku Rektor Universitas Gunadarma yang telah memfasilitasi segala hal untuk kelancaran penelitian ini. Direktorat Jendral Pendidikan Tinggi (Dikti), yang telah memungkinkan terwujudnya penelitian ini melalui bantuan dana Hibah Unggulan Perguruan Tinggi 2016. Dr. Ir. Hotniar Siringo-ringo, MSc, selaku ketua Lembaga Penelitian Universitas Gunadarma, yang telah memberikan bantuan untuk terlaksananya penelitian ini. Dr.dr. Junita Indarti, SpOG (sebagai evaluator yaitu pakar yang mampu membaca dan menganalisis servigram), dari Rumah Sakit Cipto Mangunkusumo yang telah membantu peneliti untuk mendapatkan data penelitian.

### **DAFTAR PUSTAKA**

- Aris Sugiharto. 2006. Pemrograman GUI dengan MatLab. Yogyakarta. Penerbit Andi.
- Onny Marleen, Karmilasari, Sarifuddin Madenda dan Junita Indarti. 2012. Segmentasi Lesi Kanker Serviks Berbasis Warna pada Citra Servigram. Seminar Santika.
- Onny Marleen, Sigit Wibisono dan Sarifuddin Madenda. 2013. Pendeteksian Pantulan Sinar di Area Serviks pada Citra Servikografi. Seminar Amikom.
- Onny Marleen, Widiastuti, Ricky Agus Tjiptanata. 2015. Menentukan Tingkat Kekasaran Area Lesi pada Citra Servikografi menggunakan Segmentasi Tekstur. Seminar Nasioanal Ilmu Komputer.
- Rumah Sakit Cipto Mangunkusumo. 2012. Data Penelitian. Jakarta.
- Wijaya Marvin Ch. & Prijono Agus. 2007. Pengolahan Citra Digital menggunakan MatLab Image Processing Toolbox. Bandung. Penerbit Informatika.