

IDENTIFIKASI PERSONAL BERDASAR BENTUK TANGAN

Ardi Sanjaya¹⁾, Ema Utami²⁾, Armadyah Amborowati³⁾

^{1) 2) 3)} Magister Teknik Informatika, Program Pasca Sarjana, STMIK AMIKOM Yogyakarta
Jl Ring road Utara, Condongcatur, Sleman, Yogyakarta 55281
Email : dersky@gmail.com¹⁾, emma@nrar.net²⁾, armadyah.a@amikom.ac.id³⁾

Abstrak

Geometri tangan merupakan salah satu *physiological biometrika* yang cukup banyak digunakan disamping sidik jari, wajah dan lain sebagainya. Salah satu struktur geometri tangan yang memiliki ciri unik adalah bentuk tangan. Bentuk tangan dapat digunakan untuk sistem identifikasi personal. Pada penelitian ini mencoba memberikan alternatif baru sistem identifikasi personal yaitu berdasar bentuk tangan menggunakan partisi citra. Data yang diolah adalah data citra punggung tangan kanan dari 30 orang dimana masing-masing orang diambil sebanyak 15 data citra. Dengan proses deteksi tepi menggunakan operator *prewitt* maka dapat memisahkan bentuk tangan terhadap latar. Selanjutnya setelah proses binerisasi citra dipartisi dan diambil nilai binernya untuk masing-masing partisi. Partisi yang digunakan yaitu 6x6, 8x8 dan 10x10. Proses identifikasi menggunakan jarak Euclidean. Dari hasil analisa, diperoleh akurasi sebesar 80,71% pada pengujian dengan data training sebanyak 30 dan data testing 420 partisi 6x6. Untuk data training 270 dan data testing 180 sampai dengan data training 420 dan data testing 30 didapat akurasi 100% untuk partisi 6x6, 8x8 dan 10x10.

Kata kunci: Biometrik, Identifikasi Personal, Euclidean Distance.

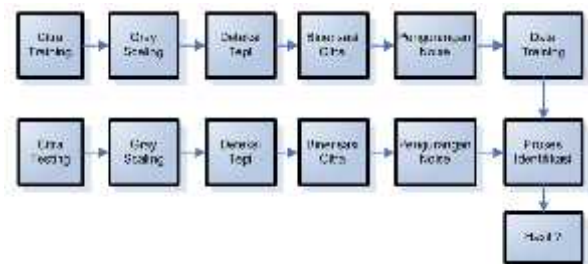
1. Pendahuluan

Di era digital seperti sekarang ini, kebutuhan terhadap sistem pengenalan diri (*personal recognition*) yang handal dan dapat dipercaya semakin meningkat terutama untuk sistem keamanan. Sistem pengenalan diri bertujuan memecahkan identitas seseorang. Terdapat dua tipe sistem pengenalan, yaitu sistem verifikasi dan identifikasi. Sistem verifikasi bertujuan untuk menerima atau menolak identitas yang diklaim oleh seseorang, sedangkan sistem identifikasi bertujuan untuk memecahkan identitas seseorang.

Geometri tangan merupakan salah satu *physiological biometrika* yang cukup banyak digunakan disamping sidik jari, wajah dan lain sebagainya. Struktur geometri tangan menyediakan area yang lebih luas sehingga lebih banyak ciri unik dapat dihasilkan untuk meningkatkan unjuk kerja sistem pengenalan terutama untuk sistem identifikasi seperti lebar dan panjang jari-jari tangan,

lebar telapak tangan, ketebalan tangan dan bentuk tangan merupakan karakteristik unik yang ada pada tangan. Kelebihan lain dari biometrika tangan adalah diantaranya kemudahan dalam penggunaannya [1].

Penelitian ini mencoba memberikan alternatif baru untuk sistem identifikasi biometrika yaitu sistem identifikasi berdasar bentuk tangan dengan menerapkan metode deteksi tepi operator *Prewitt*. Dipilih operator *Prewitt* karena berdasar dari penelitian yang dilakukan Wahyu Pujianto dkk (2009) untuk implementasi sidik jari memiliki prosentase lebih tinggi sebesar 54% dari operator lain yang digunakan yaitu *Sobel* sebesar 49%. Ilustrasi penelitian yang akan dilakukan disajikan pada gambar 1.



Gambar 1. Ilustrasi penelitian

Proses pengolahan citra dibagi menjadi 2 bagian yaitu pada saat *training* atau memasukkan data citra untuk dijadikan acuan proses identifikasi dan proses pengolahan citra yang kedua adalah *testing* atau pengujian sistem. Proses pengolahan citra *training* dan *testing* tersebut memiliki alur yang sama. Data masukan berupa citra punggung tangan kanan diolah dengan proses *grayscale*. Proses *grayscale* adalah proses mengubah citra dari format RGB (*Red Green Blue*) menjadi skala keabuan. Selanjutnya adalah proses deteksi tepi. Proses ini memisahkan obyek terhadap citra latar dan didapat citra bentuk tangan. Kemudian diproses lagi dengan mengubah menjadi citra biner dimana citra hanya memiliki nilai 0 dan 1 atau warna hitam dan putih saja. Pada proses perubahan menjadi citra biner, ada kemungkinan terdapat bintik-bintik yang tidak diinginkan atau disebut dengan *noise*. Sehingga diperlukan suatu proses tersendiri untuk menangani *noise* tersebut.

Dilanjutkan dengan pembagian citra menjadi beberapa partisi atau blok bagian dan pada masing-masing bagian tersebut diambil nilai warna hitamnya. Nantinya jumlah nilai warna hitam pada masing-masing blok pada data citra *training* akan dibandingkan dengan jumlah nilai warna hitam masing-masing blok pada data citra *testing*. Jumlah nilai dari blok yang paling mirip atau mendekati dengan data hasil pembelajaran itulah yang dianggap sebagai hasil identifikasi. Pengukuran kedekatan yang umum dipakai adalah jarak *Euclidian* yang paling minimum. Jumlah nilai dari blok dari citra *testing* yang paling mirip atau mendekati dengan data hasil *training* itulah yang dianggap sebagai hasil identifikasi. Pengukuran kedekatan yang umum dipakai adalah jarak *Euclidian* yang paling minimum.

1.1 Rumusan Permasalahan

Berdasarkan uraian diatas, maka permasalahan yang akan dibahas dalam penelitian ini adalah :

1. Bagaimana merancang sistem identifikasi geometri bentuk tangan yang menggunakan segmentasi citra dan jarak *Euclidean* ?
2. Bagaimana membuat aplikasi atau perangkat lunak untuk pengambilan data gambar punggung tangan kanan dan pengolahan citra bentuk tangan untuk deteksi tepi yang menggunakan Operator *Prewitt* ?
3. Bagaimana merancang dan membuat sistem pengambilan data citra agar citra yang dihasilkan konsisten dari sisi posisi dan pencahayaan ?
4. Bagaimana menganalisa dan mengolah data hasil identifikasi ?
5. Apakah identifikasi berdasar bentuk tangan dapat menjadi salah satu alternatif untuk sistem identifikasi ?

1.2 Batasan Permasalahan

Penelitian ini mencakup beberapa batasan yaitu :

1. Data yang diolah adalah citra bentuk tangan dalam format bitmap dengan tahapan pra-pengolahan *gray scaling* dan *thresholding*.
2. Tepi citra yang akan di identifikasi berwarna hitam dan warna latar berwarna putih.
3. Citra tangan diperoleh melalui pengambilan photo punggung tangan kanan dari atas menggunakan web cam dengan resolusi 240x240 piksel.
4. Dalam proses akuisisi, pengguna diminta menempatkan tangan kanannya pada tempat atau mal yang sudah disediakan dan atribut seperti cincin tidak dipakai.
5. Menggunakan pencahayaan bantu dari lampu LED agar pencahayaan obyek tangan konsisten.
6. Operator *gradien* yang digunakan untuk deteksi tepi adalah Operator *Prewitt*.
7. Penghilang noise yang digunakan hanya menangani noise berupa titik warna hitam tunggal dan garis yang muncul pada sisi tepi data citra.
8. Proses identifikasi bentuk tangan menggunakan segmentasi citra berupa partisi citra dan jarak *Euclidean*.

9. Aplikasi sistem identifikasi bentuk tangan dibangun menggunakan *software* Delphi 7.0

1.3 Tujuan Penelitian

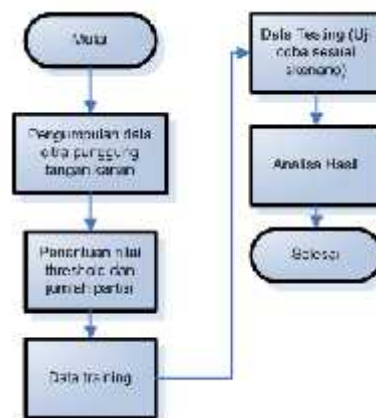
Tujuan yang diharapkan dari penelitian ini adalah :

1. Menghasilkan sistem identifikasi bentuk tangan untuk melengkapi identifikasi biometrika yang sudah banyak dipergunakan untuk meniadakan kesulitan identifikasi.
2. Menghasilkan rancangan sistem identifikasi geometri bentuk tangan yang menggunakan segmentasi citra dan jarak *Euclidean*.
3. Menghasilkan aplikasi pendukung untuk pengolahan citra bentuk tangan untuk deteksi tepi yang menggunakan Operator *Prewitt*.
4. Mengetahui dan menganalisa hasil identifikasi personal berdasar bentuk tangan menggunakan deteksi tepi operator *Prewitt* dan segmentasi citra serta jarak *Euclidean*.

Adapun metodologi penelitian yang diterapkan dalam melakukan penelitian ini dibagi menjadi beberapa bagian pokok yaitu :

1. Penentuan nilai *threshold* pada proses binerisasi citra untuk mendapatkan citra yang stabil.
2. Penentuan jumlah segmen atau blok dari citra yang akan diproses.
3. Penentuan skenario uji coba.

Alur penelitian dimulai dengan mengumpulkan data berupa data citra punggung tangan kanan dari 30 orang dengan masing-masing orang diambil 15 data. Selanjutnya adalah menentukan nilai *threshold* yang nantinya digunakan sebagai nilai ambang untuk proses binerisasi citra. Kemudian dilanjutkan dengan data *training* dan data *testing*. Terakhir adalah analisa hasil. Alur penelitian disajikan pada gambar 2 berikut :



Gambar 2. Alur penelitian

Nilai *threshold* diperlukan sebagai nilai acuan atau nilai ambang untuk proses binerisasi citra dari skala keabuan ke citra biner. Nilai ambang ditentukan 192 yang berdasar nilai ambang dari penelitian tentang verifikasi biometrika geometri tangan dengan menggunakan metode *chain code* [2] dimana hasil pengujian variasi jumlah ciri acuan di database dengan 25 partisipan

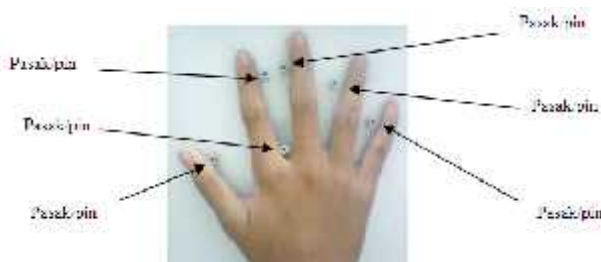
menggunakan nilai ambang 192 menghasilkan akurasi tertinggi sebesar 84,67%. Pengujian sistem atau aplikasi dilakukan dengan menggunakan parameter nilai jumlah partisi 36 (6x6), 64 (8x8) dan 100 (10x10). Skenario uji coba disajikan pada tabel 1 berikut :

Tabel1. Skenario uji coba

Ske-nario	Data Training	Data Testing	Jumlah Data Training	Jumlah Data Testing
1	1	2..15	30	420
2	1, 2	3..15	60	390
3	1,2,3	4..15	90	360
4	1..4	5..15	120	330
5	1. 5	6..15	150	300
6	1. 6	7..15	180	270
7	1. 7	8..15	210	240
8	1. 8	9..15	240	210
9	1. 9	10..15	270	180
10	1..10	11..15	300	150
11	1..11	12..15	330	120
12	1..12	13..15	360	90
13	1..13	14,15	390	60
14	1..14	15	420	30

1.4 Metode Pengumpulan Data

Data penelitian berupa citra *bitmap* bentuk tangan berukuran 240 piksel x 240 piksel didapatkan dari 30 orang. Data diambil menggunakan web cam. Setiap orang akan diambil sampel sebanyak 15 citra. Data di ambil berupa punggung tangan kanan dan menggunakan *template* yang terdapat pasak/pin dengan maksud agar posisi bentuk tangan seragam (gambar 2).



Gambar 2. Penggunaan template untuk keseragaman

1.5 Landasan Teori

1.5.1 Biometrika

Biometrika atau biometric berasal dari kata *bio* dan *metric*. *Bio* berarti sesuatu yang hidup, dan *metric* berarti mengukur [3]. Biometrika berarti mengukur karakteristik pembeda (*distinguishing traits*) pada badan atau perilaku seseorang yang digunakan untuk melakukan

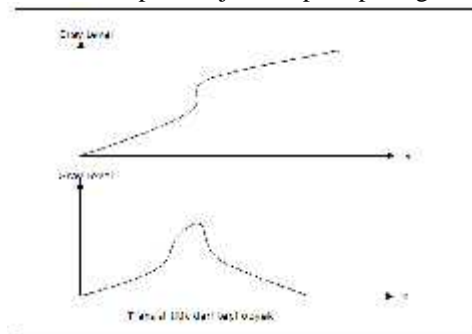
pengenalan secara otomatis terhadap identitas orang tersebut, dengan membandingkannya terhadap karakteristik yang sebelumnya telah disimpan pada suatu database. Sistem pengenalan dengan geometri tangan didasarkan pada sejumlah pengukuran dari tangan manusia yaitu ukuran telapak tangan serta lebar dan panjang jari termasuk bentuk tangan [3].

1.5.2 Sistem Identifikasi Personal

Berdasarkan identitas (ciri-ciri atau tanda-tanda) seseorang maka seseorang itu dapat dikenali. Persoalan yang menyangkut identitas seseorang tersebut dapat dikategorikan menjadi dua persoalan pokok yang berbeda dengan perbedaannya yang sangat kompleks, yaitu *verification* (pembuktian) dan *identification* (pengenalan). *Verification* (*authentication/keasliannya*) menunjukkan bahwa masalah penegasan (*confirming*) atau penyangkalan (*denying*) seseorang tentang identitasnya (“*Am I who I claim I am?*”). *Identification* (“*Who am I?*”) menunjukkan bahwa masalah pembuktian identitas dari subyek (seseorang) [3].

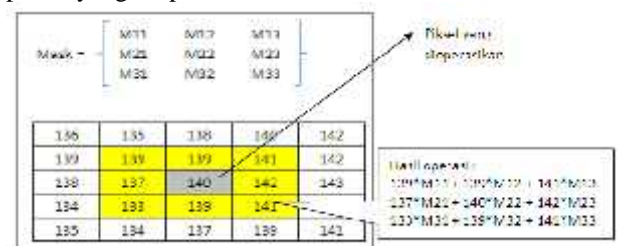
1.5.3 Deteksi Tepi

Deteksi tepi merupakan salah satu operasi dasar dari pemrosesan citra. Tepi merupakan batas dari suatu obyek. Batas suatu citra dapat dideteksi pada perbedaan tingkat keabuannya [4]. Tepi (*edge*) juga bisa diartikan bagian dari citra di mana intensitas kecerahan berubah secara drastis. Deteksi tepi obyek merupakan operasi yang mendeteksi perubahan tingkat keabuan yang signifikan dari suatu citra [4]. Representasi perubahan tingkat keabuan dapat disajikan seperti pada gambar 3 :



Gambar 3 Representasi perubahan skala keabuan

Deteksi tepi citra termasuk operasi konvolusi. Operasi konvolusi adalah operasi yang melibatkan tetangga [4] Dinamakan tetangga dari piksel karena terletak disekitar piksel yang dioperasikan.



Gambar 4 operasi konvolusi

Pencarian tepi dapat dilakukan dengan menghitung selisih atau diferensi antara 2 buah titik yang bertetangga, sehingga diperoleh nilai *gradien* atau turunan order pertama dari citra. Persamaan dasar untuk menghitung gradien 1 dimensi disajikan pada persamaan 1 berikut [5] :

$$\frac{\partial f}{\partial x} = f(x + 1) - f(x) \quad (1)$$

Pada penelitian ini, dipilih operator Prewitt karena berdasar dari penelitian sebelumnya yang dilakukan oleh Wahyu Pujiyanto dkk (2009) operator Prewitt untuk deteksi tepi pada identifikasi sidik jari memiliki prosentase 54% dan rata-rata 49% untuk *Sobel*

1.5.4 Segmentasi Citra dan Thresholding

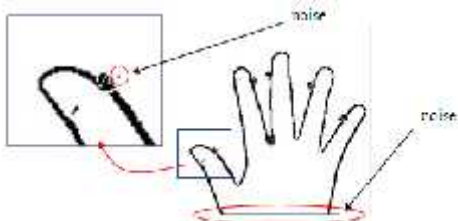
Segmentasi citra merupakan suatu proses pengelompokkan citra menjadi beberapa *region* berdasarkan kriteria tertentu. Berdasarkan pengertiannya, segmentasi memiliki tujuan menemukan karakteristik khusus yang dimiliki suatu citra. Oleh karena itulah, segmentasi sangat diperlukan pada proses pengenalan pola. Semakin baik kualitas segmentasi maka semakin baik pula kualitas pengenalan polanya [6].

1.5.5 Penanganan Noise

Noise merupakan gangguan berupa titik-titik pada citra yang keberadaannya tidak diinginkan dan muncul ketika proses pengolahan citra. Tentu saja hal tersebut menyebabkan hasil pengolahan citra tidak sesuai dengan apa yang diinginkan sebelumnya. Ketidak sesuaian citra yang telah didapatkan disebabkan oleh beberapa persoalan [4] :

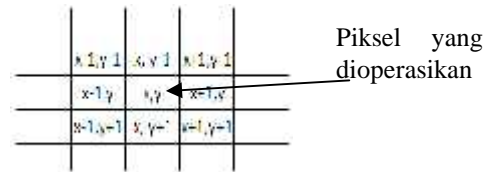
1. Karena lensa kamera yang kurang bersih, sehingga muncul titik-titik (*noise*).
2. Karena adanya gangguan pada saat melakukan pengiriman data.

Pada penelitian ini, *noise* muncul setelah proses binerisasi. *Noise*-nya didefinisikan sebagai titik-titik yang tidak diinginkan berwarna hitam dan tidak memiliki tetangga. Dan juga *noise* yang muncul pada tepi atas, samping kiri, samping kanan dan bawah dari data citra berupa garis lurus yang mungkin muncul akibat dari relatifitas *cropping* pada proses *capturing*.



Gambar 5. Noise

Untuk menghilangkan noise diperlukan suatu operasi tambahan dimana prosesnya adalah apabila terdapat titik-titik berwarna hitam yang tidak memiliki tetangga. operasi konvolusi untuk menghilangkan noise yang berupa titik tunggal menggunakan mask 3x3.

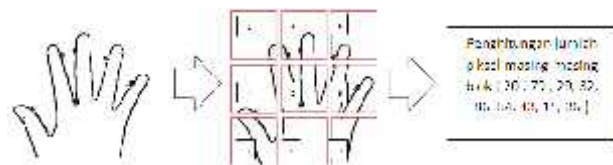


Gambar 6. Mask 3x3 untuk menghilangkan noise titik tunggal

Noise yang berupa garis pada tepi data citra dihilangkan dengan operasi atau proses yang terpisah dari penghilang noise yang berupa titik-titik tunggal. Penghilang noise pada tepi data citra hanya khusus mengubah tepi sisi atas, samping kiri, samping kanan dan bawah dari data citra menjadi warna putih karena sisi tepi atas, samping kiri, samping kanan dan bawah tidak ikut dihitung.

1.5.6 Partisi Citra

Partisi citra adalah membagi citra menjadi beberapa blok dimana masing-masing blok atau bagian memiliki ukuran yang sama besar. Pada penelitiann Implementasi Segmentasi Citra Dengan Menggunakan Metode Robust Fully Automatic Scheme [7] menggunakan partisi citra untuk mendapatkan informasi *threshold* setiap blok dan menggunakan pembagian partisi sebanyak 10, 20, 30 dan 40.



Gambar 7. Contoh data citra dibagi menjadi 9 partisi dan dihitung jumlah pikselnya

Pada penelitian ini juga menggunakan partisi citra pada salah satu pemrosesan data citra. Tujuan dilakukan partisi terhadap citra adalah menghitung jumlah piksel masing-masing blok pada data *training* dan data *testing*. Kemudian masing-masing blok pada data *training* dan *testing* dicari selisih yang terpendek menggunakan *Euclidean*.

1.5.7 Jarak Euclidean

Jarak *Euclidean* (*Euclidean distance*) yaitu metrika yang digunakan untuk menghitung kesamaan 2 vector. Jarak *Euclidean* menghitung akar dari kuadrat perbedaan 2 vector (*root of square difference between 2 vectors*) [3].

$$d_e = \sqrt{\sum_{k=1}^m (fd_{i,k} - kj)^2} \quad (2)$$

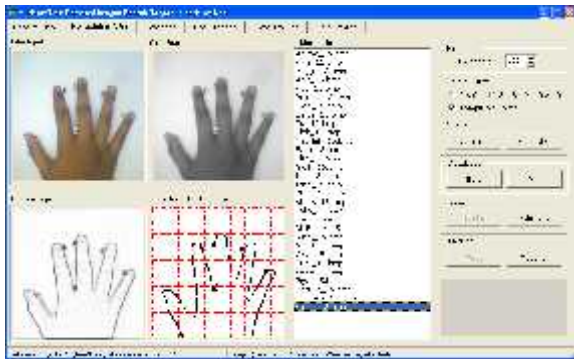
Keterangan :

- de : jarak euclidean
- fdi : bobot citra pelatihan
- kj : data bobot citra test
- m : jumlah data pelatihan

2. Pembahasan

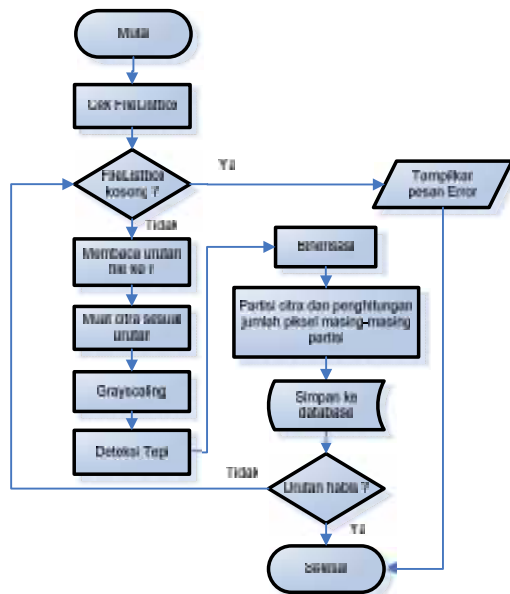
Untuk menunjang penelitian ini, diperlukan suatu aplikasi atau perangkat lunak. Sistem yang dirancang

memiliki 4 fungsi utama yaitu akuisi data citra, pengolahan citra untuk data *training*, pengolahan citra untuk data *testing* dan menyajikan hasil identifikasi. Fungsi akuisi data citra adalah untuk menyalin gambar yang ditangkap *web cam* ke dalam format bitmap.

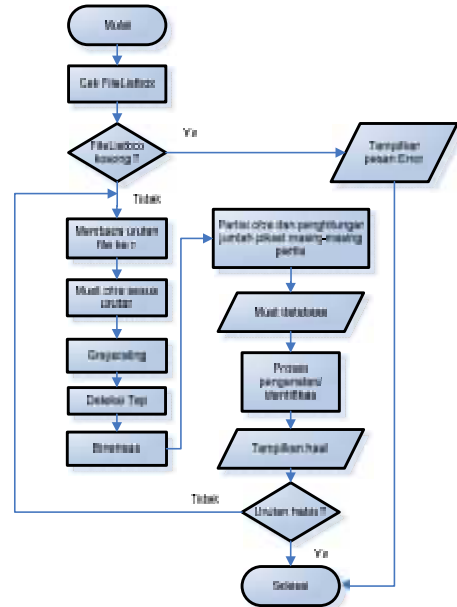


Gambar 8. Tampilan aplikasi

Pada pengolahan citra untuk data *training* dan *testing* memiliki kesamaan proses. Namun pada proses data *training*, hasilnya disimpan dalam database. Database yang digunakan pada aplikasi ini adalah jenis database yang menggunakan format *ini file*. Sedangkan pada proses data *testing*, hasil disajikan dalam bentuk tabel. Alur program untuk proses data *training* disajikan pada gambar 9 dan data *testing* disajikan pada gambar 10.



Gambar 9 Alur data *training*



Gambar 10. Alur data *testing*

2.1 Proses Grayscale

Pengolahan citra yang pertama yaitu grayscale atau mengubah citra menjadi skala keabuan. Penghitungan skala keabuan menggunakan rata-rata dari ketiga elemen warna RGB. Citra masukan dibaca nilai RGB masing-masing pikselnya.

2.2 Proses Deteksi Tepi

Proses deteksi tepi menggunakan operator *Prewitt* seperti yang dibahas pada bab sebelumnya. Operator *Prewitt* pada aplikasi ini menggunakan mask ukuran 3x3 dan didefinisikan pada unit utama dari aplikasi. Variabel *Mask1* dan *Mask2* digunakan untuk operasi konvolusi horizontal dan vertikal. Berikut segmen programnya :

```

const
MaskRahm1: TMask3x3 = ((0,1,0),(0,1,0),(0,0,-1));
MaskRahm2: TMask3x3 = ((0,1,0),(0,1,0),(0,0,-1));
MaskSobelHorizontal: TMask3x3 = ((-1,0,1),(-1,0,1),(-1,0,1));
MaskSobelVertical: TMask3x3 = ((-1,-1,1),(0,0,0),(1,1,1));
MaskSobelHorizontal2: TMask3x3 = ((-1,0,1),(-1,0,1),(-1,0,1));
MaskSobelVertical2: TMask3x3 = ((-1,-1,1),(0,0,0),(1,1,1));
MaskThresh: TMask3x3 = ((0,1,0),(0,1,0),(0,1,0));
    
```

Pada operasi konvolusi, ditambahkan instruksi untuk menghitung konvolusi terhadap kedua arah menggunakan *Mask1* dan *Mask2*, hasilnya disimpan dalam variabel *jumlah1* dan *jumlah2*. Berikut segmen programnya :

```

jumlah1:=0;
for i:=1 to 1 do
  for v:=1 to 1 do
    jumlah1:=jumlah1+ mask1[u,v]*ki[x-u,y-v];
  jumlah2:=0;
  for i:=1 to 1 do
    for v:=1 to 1 do
      jumlah2:=jumlah2+ mask2[u,v]*ki[x-u,y-v];
    
```

Kemudian menentukan nilai keabuan dari masing-masing elemen RGB. Hasilnya disimpan pada masing-

masing variabel elemen warna yaitu variabel Ro, Go dan Bo. Berikut segmen programnya :

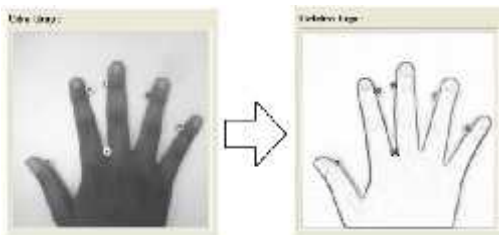
```

R=[x,y]:=Round((aks(jumlah1) + aks(jumlah2))/2);
B=[x,y]:=255-Bo[x,y]//negatif;
jumlah1:=0;
for ur=-1 to 1 do
  for vs=-1 to 1 do
    jumlah1:=jumlah1 + Mask1[u,v] *Bi[x+u, y+v];
  jumlah1:=0;
  for ur=-1 to 1 do
    for vs=-1 to 1 do
      jumlah2:=jumlah2 + Mask2[u,v] *Bi[x+u, y+v];
    jumlah2:=0;
  end;
  G=[x,y]:=Round((aks(jumlah1) + aks(jumlah2))/2);
  B=[x,y]:=255-Bo[x,y] //negatif;
  jumlah1:=0;
  for ur=-1 to 1 do
    for vs=-1 to 1 do
      jumlah2:=jumlah2 + Mask2[u,v] *Bi[x+u, y+v];
    jumlah2:=0;
  end;
  D=[x,y]:=Round((aks(jumlah1) + aks(jumlah2))/2);
  B=[x,y]:=255-Bo[x,y] //negatif;
    
```

Setelah menghitung nilai perbedaan antara titik yang bertetangga, didapatlah hasil deteksi tepi.

```

for y:=0 to h-1 do
  begin
    DR:=FMain.citra_tepi.picture.bitmap.secLine[y];
    for x:=0 to w-1 do
      begin
        R=[3*x]:=R[x,y];
        G=[3*x+1]:=G[x,y];
        B=[3*x+2]:=B[x,y];
      end;
    end;
  end;
    
```



Gambar 11. Citra grayscale diproses dengan deteksi tepi

2.3 Proses Binerisasi Citra

Proses selanjutnya adalah mengubah citra dari skala keabuan ke dalam citra biner. Penentuan nilai biner menggunakan nilai ambang (*threshold*) 192. Ketika nilai rata-rata RGB masing-masing piksel pada citra yang sudah diubah menjadi skala keabuan didapat, jika kurang dari ambang maka nilai piksel menjadi 0 dan jika sama dengan atau lebih dari nilai ambang maka nilainya 255. Berikut segmen program untuk mengubah citra menjadi citra biner :

```

//Proses Binerisasi
for y:=0 to FMain.citra_tepi.Dietura.Height-1 do
  begin
    R:=FMain.citra_tepi.picture.bitmap.secLine[y];
    DR:=FMain.citra_tepi.Dietura.BitMap.secLine[y];
    for x:=0 to FMain.citra_tepi.Dietura.Width-1 do
      begin
        R:=R[3*x+2];
        G:=G[3*x+1];
        B:=B[3*x];
        K:=Round((R+G+B)/3);
      end;
    end;
  end;
    
```

2.4 Penanganan Noise

Noise yang muncul pada penelitian ini berupa titik berwarna hitam tunggal serta garis yang ada pada bagian tepi atas, samping kiri, samping kanan dan bawah dari data citra. Berikut segmen program untuk menghilangkan noise berupa titik hitam tunggal :

```

if FMain.citra_tepi.Dietura.BitMap[x,y]=0 then
  begin
    if ((FMain.citra_tepi.Dietura.BitMap[x-1,y]=0) AND
        (FMain.citra_tepi.Dietura.BitMap[x+1,y]=0) AND
        (FMain.citra_tepi.Dietura.BitMap[x,y-1]=0) AND
        (FMain.citra_tepi.Dietura.BitMap[x,y+1]=0) AND
        (FMain.citra_tepi.Dietura.BitMap[x,y]=0) AND
        (FMain.citra_tepi.Dietura.BitMap[x-1,y-1]=0) AND
        (FMain.citra_tepi.Dietura.BitMap[x-1,y+1]=0) AND
        (FMain.citra_tepi.Dietura.BitMap[x+1,y-1]=0) AND
        (FMain.citra_tepi.Dietura.BitMap[x+1,y+1]=0)) then
      begin
        K:=255;
        DR[3*x+2]:=K;
        DR[3*x+1]:=K;
        DR[3*x]:=K;
      end;
    end;
  end;
    
```

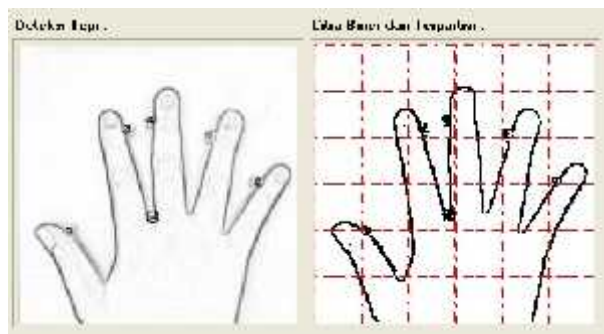
Sedangkan untuk *noise* yang berupa garis pada bagian tepi atas, samping kiri, samping kanan dan bawah dari data citra menggunakan algoritma sebagai berikut :

```

if (y=0) OR (x=0) OR (y=FMain.citra_tepi.Dietura.Height-1) OR
(x=FMain.citra_tepi.Dietura.Width-1) then K:=255;
    
```

2.5 Partisi Citra

Ketika memproses pembagian partisi, aplikasi juga menghitung jumlah titik-titik atau piksel pada masing-masing partisi. Pada proses data *training*, jumlah titik masing-masing partisi disimpan pada database. Sedangkan pada proses data testing, jumlah titik masing-masing partisi disimpan sementara pada sebuah stringgrid. Berikut contoh proses partisi citra :



Gambar 12. Proses dari deteksi tepi menjadi citra biner dan terpartisi

2.6 Proses Identifikasi

Pada saat proses data *testing*, setelah nilai jumlah piksel masing-masing partisi diperoleh kemudian dihitung selisihnya atau dicari jarak *euclidean*-nya terhadap semua data citra yang ada pada *database*. Untuk mengetahui benar atau tidak dalam mengidentifikasi dengan memeriksa kesamaan nama file. Contoh hasil proses data *testing* disajikan pada gambar 13 :

No	Nama File	Hasil Identifikasi	Jarak	Kategori
1	Aditya_11.bmp	Aditya	35,59	Benar
2	Aditya_12.bmp	Aditya	24,49	Benar
3	Aditya_13.bmp	Aditya	51,21	Benar
4	Aditya_14.bmp	Aditya	50,21	Benar
5	Aditya_15.bmp	Aditya	51,28	Benar

Gambar 13. Contoh hasil proses data *testing*

Pada proses data *testing* pertama, file *Aditya_11.bmp* dikenali sebagai *Aditya*. Disini aplikasi memeriksa kesamaan nama file yang diperiksa dengan mengabaikan nomor urut atau angka pada nama file yang di uji terhadap nama data yang muncul atau hasil pengenalan. Jika nama hasil pengenalan tidak sama terhadap nama file yang diuji maka hasil identifikasi dikategorikan benar. Sebaliknya jika nama hasil pengenalan tidak sama terhadap nama file yang diuji maka hasil identifikasi dikategorikan salah.

2.7 Hasil

Hasil pengujian dengan data akurasi yang sesuai dengan skenario uji coba dirangkum dan disajikan pada tabel 2.

Tabel 2. Data akurasi hasil pengujian

Skenario	Partisi		
	6x6 (%)	8x8 (%)	10x10 (%)
1	80,71	83,81	82,62
2	91,28	95,90	97,44
3	92,50	99,17	99,44
4	93,03	97,88	99,09
5	96,33	97,67	99,33
6	95,93	97,41	98,89
7	98,33	100,00	100,00
8	99,05	100,00	100,00
9	100,00	100,00	100,00
10	100,00	100,00	100,00
11	100,00	100,00	100,00
12	100,00	100,00	100,00
13	100,00	100,00	100,00
14	100,00	100,00	100,00

Dari tabel 2 diatas dapat dilihat pada skenario 1 dengan data *training* 30 dan data *testing* 420 untuk partisi citra 6x6 memiliki nilai akurasi 80,71 %. Hal tersebut

dikarenakan acuan yang digunakan pada proses identifikasi jauh lebih sedikit dibandingkan dengan data *testing*. Pada skenario 7 dan 8 partisi 8x8 dan 10x10 hasil akurasi 100% karena perbandingan jumlah data *training* dan data *testing* selisihnya tidak terlalu jauh. Pada skenario 9 sampai 14 hasil akurasi 100% pada ketiga macam partisi karena data *training* lebih banyak dibandingkan data *testing*. Berdasarkan rata-rata data akurasi, semakin besar jumlah partisi citra maka nilai akurasi juga akan semakin besar.

Tabel 3. Data durasi proses identifikasi (detik)

Skenario	Partisi		
	6x6 (detik)	8x8 (detik)	10x10 (detik)
1	4,01	2,12	1,45
2	3,66	1,96	1,23
3	3,34	1,87	1,29
4	3,06	1,67	1,08
5	2,84	1,64	0,94
6	2,47	1,35	0,85
7	2,92	1,41	0,76
8	2,18	1,14	0,66
9	1,67	1,06	0,57
10	1,75	0,76	0,52
11	1,15	0,72	0,46
12	1,06	0,54	0,35
13	0,7	0,35	0,35
14	0,39	0,15	0,14

Dari data tabel 3, dapat diketahui bahwa rata-rata durasi proses identifikasi paling minimum adalah pada skenario 1 partisi 6x6 yaitu sebesar 4,01 detik. Sedangkan skenario 14 partisi 10x10 memiliki durasi 0,14 detik. Hal tersebut disebabkan semakin besar data *training* atau data acuan maka durasi untuk proses identifikasi menjadi lebih lama karena proses membaca data *training* memerlukan durasi yang lebih lama.

3. Kesimpulan

Berdasarkan dari yang diuraikan diatas, maka dapat diambil kesimpulan :

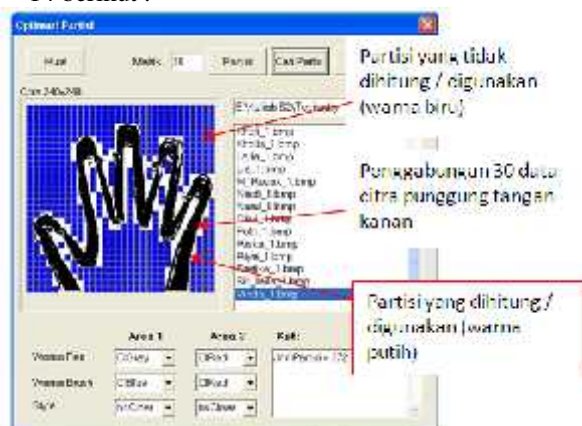
1. Penempatan pasak atau pin sangat membantu saat pengambilan data citra punggung telapak tangan karena data seragam dan penggunaan lampu led mampu membantu memberikan pencahayaan sehingga pencahayaan terhadap punggung tangan kanan yang dihasilkan konsisten.
2. Perbandingan jumlah data *training* dengan data *testing* berpengaruh terhadap akurasi proses identifikasi. Semakin banyak jumlah data *training* dibandingkan data *testing* maka hasil akurasi

mendekati atau sama dengan 100% seperti pada skenario 9 sampai dengan 14 untuk ketiga macam partisi.

3. Data *training* yang jumlahnya lebih sedikit dibandingkan data testing ada kemungkinan terjadi kesalahan pencocokan lebih besar seperti pada skenario 1 partisi citra 6x6 yang memiliki nilai akurasi 80,71%.
4. Konsekuensi dari jumlah data *training* yang lebih banyak dibandingkan data *testing* adalah durasi proses identifikasi menjadi lebih lama. Rata-rata durasi proses identifikasi paling minimum adalah pada skenario 1 partisi 6x6 yaitu sebesar 4,01 detik. Sedangkan skenario 14 partisi 10x10 memiliki durasi 0,14 detik.
5. Hasil penelitian akan valid jika data yang diolah tidak memiliki perubahan fisik yang dapat menyebabkan struktur bentuk tangan berubah.
6. Sistem identifikasi personal berdasar bentuk tangan menggunakan segmentasi citra yaitu *thresholding* dan partisi citra dapat menjadi alternatif untuk sistem identifikasi.

Saran untuk tindak lanjut dari penelitian ini :

1. Kelemahan dari hasil penelitian ini adalah apabila diterapkan pada aplikasi pencatat kehadiran/presensi maka harus memperhatikan perubahan fisik bentuk tangan, ada luka yang bersifat permanen, memakai asesoris seperti cincin atau sejenisnya sehingga perlu dilakukan penelitian lebih lanjut serta mempertimbangkan faktor cahaya luar yang mungkin bisa berpengaruh pada proses akuisi data citra.
2. Untuk lebih meningkatkan kinerja dari sistem ini, maka pada penelitian selanjutnya akan menggunakan rata-rata partisi yang digunakan berdasarkan 30 data yang digunakan, seperti diilustrasikan pada gambar 14 berikut :



Gambar 14. Rata-rata penggunaan partisi

Daftar Pustaka

- [1] Duta, N. 2009, *A Survey Of Biometric Technology Based On Hand Shape*, Pattern Recognition, 2797-2806
- [2] Darma Putra, I. Ari Made Santosa, 2011, *Verifikasi Biometrika Geometrika Tangan Dengan Metode Chain Code*, SNATIKA 2011, Edisi 10

- [3] Darma Putra, I. 2009, *Sistem Biometrika*, Andi Publishing, Yogyakarta
- [4] Purnomo, M. Muntasa, A. 2010, *Konsep Pengolahan Citra Digital dan Ekstraksi Fitur*, Graha Ilmu
- [5] Achmad, Balza. 2005, *Teknik Pengolahan Citra Digital menggunakan DELPHI*. Yogyakarta: Ardi Publishing.
- [6] Sa'adah, S. 2009, *Segmentasi Citra*, http://digilib.itelkom.ac.id/index.php?option=com_content&view=article&id=575:segmentasi-citra&catid=20:informatika&Itemid=14, diakses 15 Mei 2013
- [7] Desti A, A. Subakti, I. Soelaiman, R. 2011, *Implementasi Segmentasi Citra Dengan Menggunakan Metode Robust Fully Automatic Scheme*, <http://digilib.its.ac.id/public/ITS-Undergraduate-15287-Paper-pdf.pdf>, diakses 15 Mei 2013

Biodata Penulis

Ardi Sanjaya, S.Kom., pernah bekerja di PT.Advanced Interconnect Technologies Batam tahun 2000-2006 sebagai Sealing Metal QUAD and Laser Mark Process Engineer. Memperoleh gelar Sarjana Komputer (S.Kom) pada Program Studi Teknik Informatika STT Cahaya Surya Kediri tahun 2010. Saat ini melanjutkan studi di Program Magister Teknik Informatika STMIK AMIKOM Yogyakarta dan menempuh tesis. Saat ini mengajar di Program Studi Teknik Informatika Universitas Nusantara PGRI Kediri

Dr. Ema Utami, S.Si, M.Kom, memperoleh gelar Sarjana Sains (S.Si) dari Program Studi Ilmu Komputer UGM pada tahun 1997. Tahun 2002 memperoleh gelar Magister Komputer (M.Kom) dengan predikat cumlaude dari Program Pascasarjana Ilmu Komputer UGM. Tahun 2010 memperoleh gelar Doktor dari Program Doktor Ilmu Komputer UGM. Sejak 1998 menjadi Staff Pengajar di STMIK AMIKOM Yogyakarta dan sejak 2010 menjadi Wakil Direktur I Bidang Akademik Program Pascasarjana STMIK AMIKOM Yogyakarta.

Armadyah Amborowati, S.Kom, M.Eng, memperoleh gelas S.Kom di STMIK AMIKOM Yogyakarta pada tahun 2003. Tahun 2009 memperoleh M.Eng dari MTI Fakultas Teknik Elektro UGM. Mulai 2011 sedang menempuh program Doktor di Ilmu Komputer UGM. Saat ini sebagai staf pengajar di STMIK AMIKOM Yogyakarta.