

# Implementasi VB 6.0 pada Face Detection Berbasis Image Processing untuk Sistem Identifikasi

Millatul Maziyah\* dan Andy Noortjahja  
Jurusan Fisika, FMIPA-Universitas Negeri Surabaya  
Jl. Ketintang Kampus UNESA, Surabaya 60231

## Intisari

Pengenalan wajah manusia untuk sistem identifikasi dapat dikembangkan dengan menggunakan *image processing*, yaitu dengan menggunakan sistem warna YCrCb untuk mendeteksi warna kulit, *integral projection* untuk menentukan lokasi wajah dan *edge detection* untuk menentukan tepi-tepi dari obyek, kemudian *feature extraction* untuk mengetahui letak mata dan mulut guna mendapatkan nilai fitur-fitur wajah seperti jarak dekat dan jauh mata, lebar mulut dan jarak mata terhadap mulut. Dari hasil didapatkan nilai rata-rata MSE-noise = 0,7777, MSE-filter = 0,0698 dan SNR = 25,7796 dB untuk filter gaussian blur dengan probabilitas noise 0,05 dan SNR sebelum filter = 14,3406 dB. Prosentase keberhasilan mendeteksi wajah = 98,5%; mendeteksi keseluruhan fitur wajah = 88,75%; dan keberhasilan mengenali wajah = 81,67%.

KATA KUNCI: image processing, YCrCb, integral projection, feature extraction

## I. PENDAHULUAN

Melalui sebuah *webcam* peneliti ingin merealisasikan piranti yang akan dibuat, dengan kemajuan ilmu pengetahuan dan teknologi saat ini, *webcam* dengan tambahan *software* juga dapat dimanfaatkan untuk mendeteksi dan mendata wajah manusia sebagai proses identifikasi. *Software* yang dimaksud adalah *visual basic 6.0* yang berbasis *image processing*.

*Image processing* adalah suatu sistem di mana proses dilakukan dengan masukan berupa *image* dan hasilnya juga berupa *image* [1]. Penggunaan *image processing* pada penelitian ini merupakan suatu alternatif dalam pengenalan wajah (*face recognition*).

Metode yang digunakan untuk mendeteksi wajah manusia yang telah tertangkap oleh *webcam* adalah metode *skin colour detection*, *cropping* dengan integral proyeksi, *edge detection*, dan *feature extraction*. Metode *skin colour detection* adalah proses di mana setiap wajah manusia yang dideteksi akan diambil nilai rata-rata yang berupa warna kulit. Metode *cropping* adalah proses pengambilan obyek yang berupa wajah saja dengan meng-*crop* bagian yang terdeteksi sebagai wajah. *Edge detection* adalah proses untuk menentukan tepi-tepi dari obyek. Metode *feature extraction* adalah proses program mendeteksi fitur-fitur wajah seperti mata dan mulut.

Tujuan penelitian ini untuk membuat implementasi VB 6.0 pada *face detection* berbasis *image processing* untuk sistem identifikasi, dengan batasan masalah diantaranya wajah harus biasa dengan posisi lurus menghadap *webcam* saat pengambilan gambar. Pencahayaan tetap, tidak terlalu gelap dan tidak terlalu terang, serta antara obyek (wajah) dan *background* harus berbeda (*background* tidak boleh menyerupai warna kulit). Hanya satu obyek (wajah) yang di depan *webcam*.

\*E-MAIL: azzie.@plasa.com

## II. DASAR TEORI

### Digital image processing

*Image processing* adalah suatu sistem di mana proses dilakukan dengan masukan berupa *image* dan hasilnya juga berupa *image* [1]. *Digital image processing* memiliki banyak aplikasi seperti pada bidang penginderaan jarak jauh, robotik, pemetaan, biomedis, dan sebagainya. Perlengkapan *Digital image processing* minimal terdiri atas alat pemasukan data *image* berupa *digitizer* atau *scanner*, komputer, dan alat penyimpanan data dengan kapasitas yang besar [2].

### RGB

Untuk dapat menangkap/mengambil *image* dapat dilakukan dengan menggunakan obyek *PictureBox*. Perintah yang digunakan untuk menangkap *image* adalah *point(x,y)* yang menyatakan nilai warna pada posisi (x,y) dalam *PictureBox*. Karena *image* yang ada dianggap sebagai *image* berwarna, maka nilai dari *point(x,y)* ini berupa nilai RGB (*Red-Green-Blue*), di mana nilainya berada dalam range h000000 sampai dengan hffffff, dengan h000000 menyatakan warna hitam dan hffffff menyatakan warna putih. Definisi nilai warna di atas dinyatakan dalam Gambar 1. Variabel h menyatakan angka di belakangnya adalah hexadecimal.

### Noise

Pada saat proses *capture* (pengambilan *image*), ada beberapa gangguan yang mungkin terjadi, seperti kamera tidak fokus



Gambar 1: Definisi nilai warna pada *image* [1]

$\frac{1}{159}$	2	4	5	4	2
	4	9	12	9	4
	5	12	15	12	5
	4	9	12	9	4
	2	4	5	4	2

Gambar 2: Gaussian mask [3]

atau munculnya bintik-bintik yang bisa jadi disebabkan oleh proses *capture* yang tidak sempurna. Setiap gangguan pada *image* dinamakan dengan noise. Noise pada *image* tidak hanya terjadi karena ketidaksempurnaan dalam proses *capture*, tetapi bisa juga disebabkan oleh kotoran-kotoran yang terjadi pada *image*.

Dalam penelitian ini *image* hasil *capture* direduksi menggunakan filter *gaussian blur* dengan melewati *gaussian mask 55* pada *image* seperti ditunjukkan Gambar 2.

**SNR**

Di dalam *image processing*, SNR dari sebuah *image* biasanya disebut sebagai perbandingan dari rata-rata nilai pixel untuk standar deviasi dari nilai-nilai pixel. SNR ini biasanya dipakai untuk mengukur kinerja suatu filter. Jika SNR setelah proses filter nilainya lebih besar dari sebelum proses filter, ini berarti noise pada *image* berkurang. Berikut rumusan SNR [1].

$$SNR(dB) = 10^{10} \log\left(\frac{I}{N}\right) \tag{1}$$

dengan I adalah jumlah nilai *image*, N adalah jumlah nilai noise.

**MSE**

Mean square error (MSE) adalah kesalahan (*error*) minimal dari suatu sinyal *image*. Berikut rumus perhitungan MSE [4]:

$$MSE = \frac{\sum_{k=1}^n (y - y_k)^2}{n} \tag{2}$$

dengan y adalah nilai aktual *image*,  $y_k$  adalah nilai hasil pemrosesan *image*, n adalah jumlah data.

**Skin color detection**

*Skin colour detection* adalah metode yang digunakan untuk mendeteksi warna kulit. Untuk mengubah *image* berwarna yang mempunyai nilai matrik masing-masing R, G dan B menjadi *image* warna YCrCb [5] dapat dilakukan dengan menghitung seperti persamaan-persamaan di bawah ini:

$$\begin{aligned} Y &= 0.59G + 0.31R + 0.11B \\ Cr &= 0.713 * (R - Y) \\ Cb &= 0.564 * (B - Y) \end{aligned} \tag{3}$$

Kemudian nilai warna Cr dan Cb yang terkumpul dinormalisasi untuk mendapatkan nilai rata-rata Cr dan Cb untuk kemudian dilakukan perhitungan jarak kulit pada setiap lokasi yang hendak dikenali ([5]). Jika jarak kulit di bawah nilai *thresholding* maka dianggap sebagai kulit jika tidak dianggap sebagai latar belakang. Berikut rumus perhitungan jarak:

$$jarak = \sqrt{(Cr - \overline{Cr})^2 + (Cb - \overline{Cb})^2} \tag{4}$$

**Thresholding**

*Thresholding* digunakan untuk mengatur jumlah derajat keabuan yang ada pada *image*. Proses *thresholding* ini pada dasarnya adalah proses perubahan kuantisasi pada *image*, sehingga untuk melakukan *thresholding* dengan derajat keabuan b dapat digunakan rumus [5]:

$$x = \frac{w}{b} \tag{5}$$

dengan w = nilai derajat keabuan sebelum *thresholding*, x = nilai derajat keabuan setelah *thresholding*, b = jumlah derajat keabuan yang diinginkan.

**Histogram**

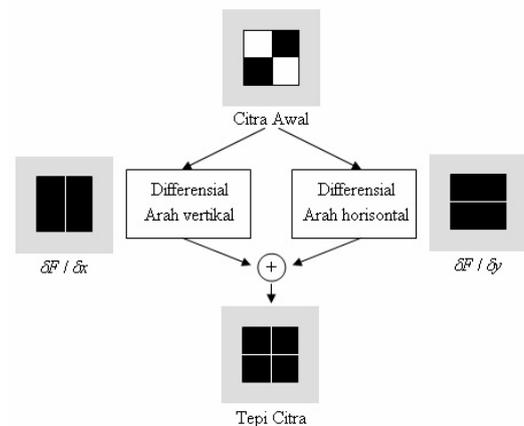
Untuk menyatakan distribusi data dari nilai derajat keabuan ini dapat digunakan nilai histogram. Histogram adalah fungsi yang menyatakan jumlah kemunculan dari setiap nilai. Misalkan diketahui data sebagai berikut:

X = 1 3 2 5 3 0 2 1 2 4 3 2

Maka histogramnya adalah munculnya setiap nilai, yaitu 0 muncul 1 kali, 1 muncul 2 kali, 2 muncul 4 kali, 3 muncul 3 kali, 4 muncul 1 kali, dan 5 muncul 1 kali. Karena *image* mempunyai derajat keabuan 256 yaitu (0-255) maka histogram menyatakan jumlah kemunculan setiap nilai 0-255 [1].

**Integral projection**

Integral projection adalah suatu metode yang digunakan untuk mencari daerah atau lokasi dari obyek. Metode ini dapat digunakan untuk mendeteksi batas dari daerah *image* yang



Gambar 3: Proses *edge detection*

berbeda, sehingga kita bisa mencari daerah lokasi wajah dan fitur-fiturnya. Setelah dilakukan thresholding dari rumus jarak kulit lalu dilakukan perhitungan nilai awal wajah (x1,y1) dan nilai akhir wajah (x2,y2) dengan dua integral proyeksi yaitu integral proyeksi vertikal dan integral proyeksi horisontal untuk mendapatkan letak wajah pada image, di mana image tersebut merupakan image dua dimensi yang mempunyai koordinat x dan koordinat y.

**Edge detection**

Deteksi tepi (*edge detection*) pada suatu image adalah suatu proses yang menghasilkan tepi-tepi dari obyek-obyek image, yang bertujuan untuk menandai bagian yang menjadi detail image dan untuk memperbaiki detail dari image yang kabur, yang terjadi karena error atau adanya efek dari proses akuisisi image. Suatu titik (x,y) dikatakan sebagai tepi (edge) dari suatu image bila titik tersebut mempunyai perbedaan yang tinggi dengan tetangganya [1]. Gambar 3 menunjukkan bagaimana tepi (edge) suatu image diperoleh.

**Microsoft visual basic 6.0**

Visual basic adalah salah satu bahasa pemrograman berbasis Windows yang memiliki fasilitas *Object Oriented Programming* (OOP) yang menyediakan obyek-obyek yang sangat kuat, powerfull, dan mudah digunakan dalam mendesain suatu aplikasi program. Visual basic program terdiri dari banyak sub program (procedur), setiap prosedur mempunyai kode tersendiri dan dapat dieksekusi sendiri dan pada saat bersamaan dapat digabung menjadi satu [6].

**III. METODOLOGI PENELITIAN**

**Rancangan penelitian:**

- Blok diagram system (Gambar 4.)
- Flowchart sistem (Gambar 5.)

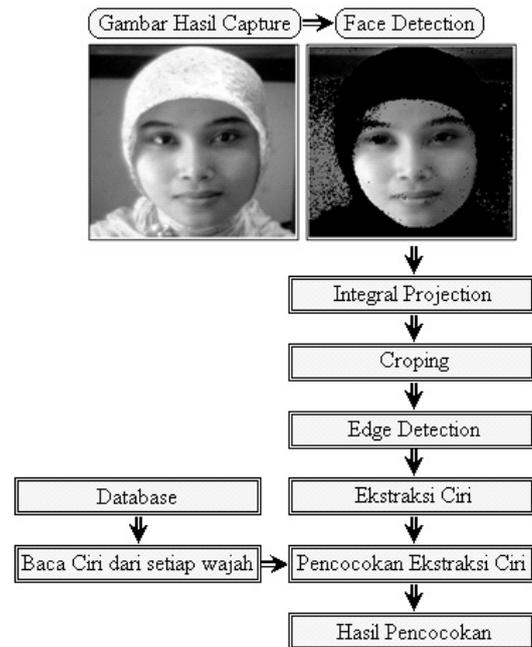
**Variabel penelitian:**

- Variabel manipulasi : wajah manusia
- Variabel terikat : hasil identifikasi
- Variabel kontrol : visual basic 6.0

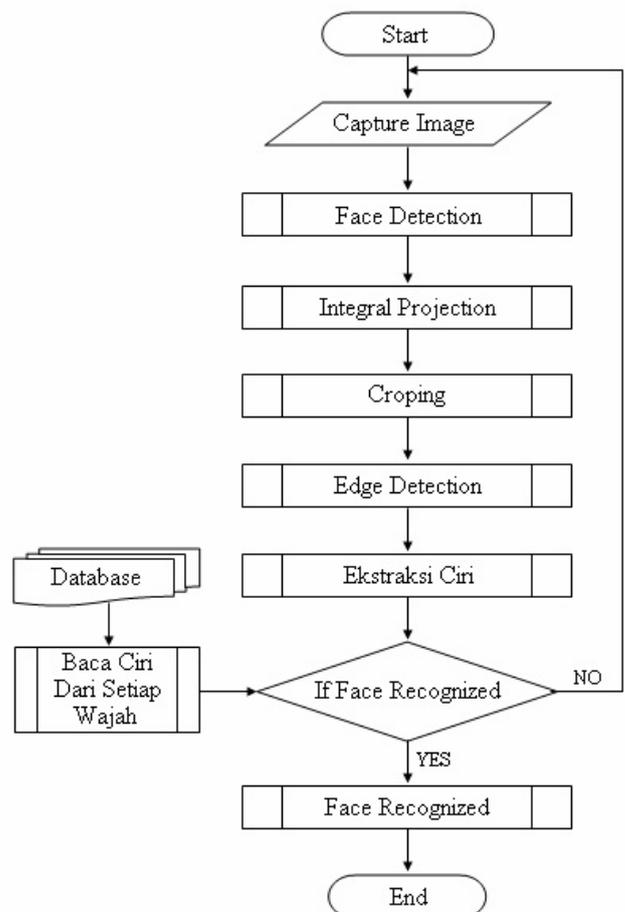
**Prosedur penelitian:**

1. Proses capture
2. Pengukuran nilai SNR dan MSE

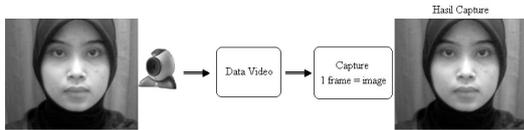
Pengukuran nilai MSE (*Mean Square Error*) untuk mengetahui besar error yang terjadi dari sinyal yang diberi noise dengan membangkitkan noise gaussian (MSE-noise) dan sinyal yang diberi noise tersebut diberikan filter (MSE-filter). Filtering dalam penelitian



Gambar 4: Blok Diagram Sistem



Gambar 5: Flowchart Sistem



Gambar 6: Proses capture

ini memakai gaussian filter dan gaussian blur dengan menggunakan rumusan:

$$MSE_{noise} = \frac{\sum_{n=1}^n (y - y_n)^2}{n}$$

$$MSE_{filter} = \frac{\sum_{f=1}^n (y - y_f)^2}{n} \tag{6}$$

dengan  $y$  = Nilai sinyal masukan,  $y_n$  = Nilai hasil setelah terkena noise,  $y_f$  = Nilai hasil setelah diberi filter,  $n$  = Jumlah data.

Pengukuran nilai SNR (*Signal-to-Noise Ratio*) dilakukan dengan menggunakan rumusan:

$$SNR(dB) = 10 \log_{10} \left( \frac{P_{signal}}{P_{noise}} \right) \tag{7}$$

$$= 20 \log_{10} \left( \frac{A_{signal}}{A_{noise}} \right) \tag{8}$$

3. Proses *skin colour detection*, mengubah sistem warna RGB ke dalam YCrCb dengan persamaan-persamaan:

$$Y = 0.59G + 0.31R + 0.11B$$

$$Cr = 0.713 * (R - Y)$$

$$Cb = 0.564 * (B - Y)$$

dilanjutkan dengan thresholding perhitungan jarak kulit dengan rumusan:

$$jarak = \sqrt{(Cr - \overline{Cr})^2 + (Cb - \overline{Cb})^2}$$

Jika nilai jarak < *thresholding* = warna kulit

Jika nilai jarak > *thresholding* = background

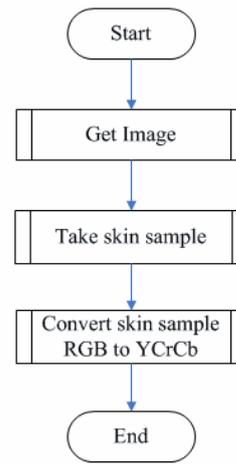
Flowchart *skin colour detection* ditunjukkan Gambar 7.

4. Proses integral projection, dilakukan untuk mencari lokasi obyek yang diinginkan. Flowchart integral projection ditunjukkan Gambar 8.

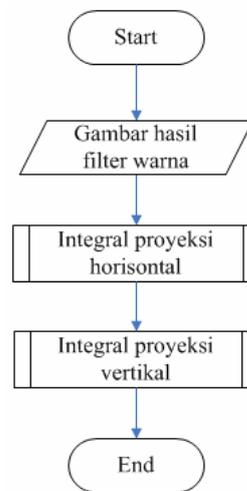
5. Proses *cropping*, menentukan daerah subset dari gambar yang diperkirakan terdapat wajah. Flowchart sistem pada proses *cropping* ditunjukkan Gambar 9.

6. Proses *edge detection*, dilakukan untuk menghasilkan tepi-tepi dari obyek-obyek *image*, dengan flowchart sistemnya ditunjukkan Gambar 10.

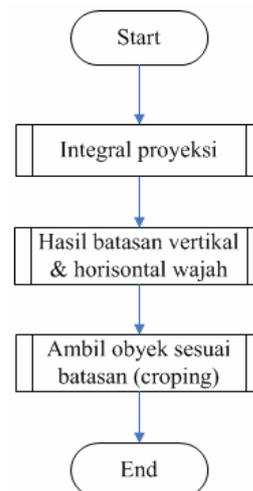
7. Proses *feature extraction*, digunakan untuk mengambil ciri-ciri yang penting dari wajah. Langkah pertama mendeteksi nilai puncak dalam histogramnya [7].



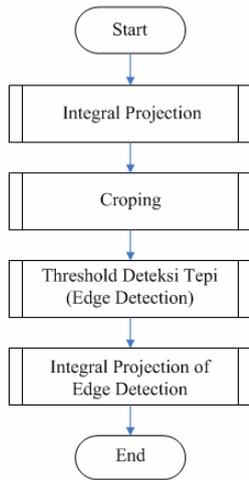
Gambar 7: Flowchart *skin colour detection*



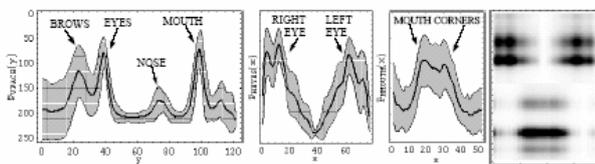
Gambar 8: Flowchart integral projection



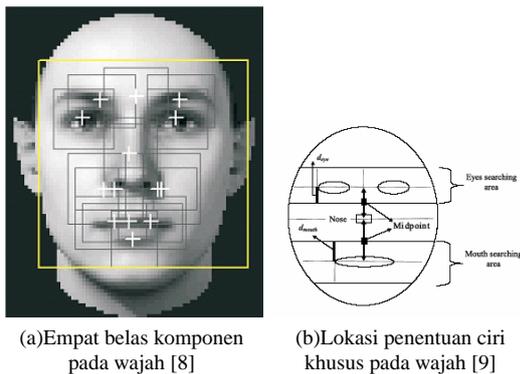
Gambar 9: Flowchart proses *cropping*



Gambar 10: Flowchart *edge detection*

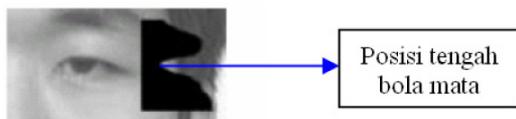


Gambar 11: Grafik Integral proyeksi pada wajah

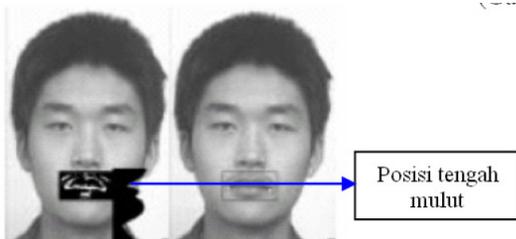


(a)Empat belas komponen pada wajah [8]

(b)Lokasi penentuan ciri khusus pada wajah [9]

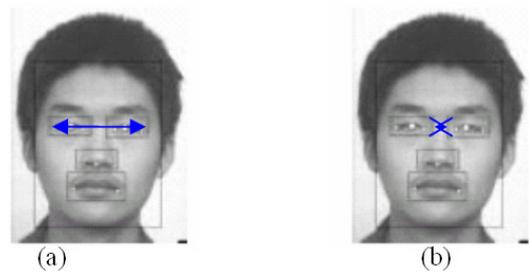


(c)Deteksi pada bola mata [10]

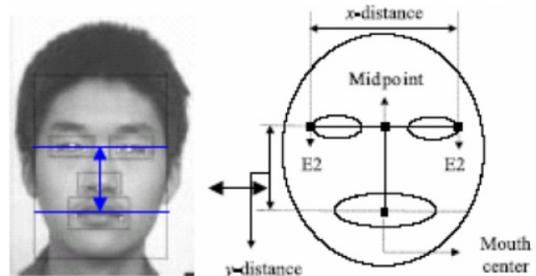


(d)Deteksi pada mulut [10]

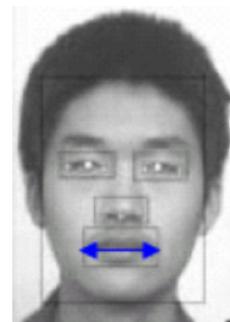
Gambar 12: Metode integral proyeksi



(a)Perhitungan (1) jarak jauh mata dan (2) jarak dekat mata



(b)Perhitungan jarak mata dan mulut



(c)Perhitungan jarak mulut

Gambar 13: Geometri wajah

Proses selanjutnya adalah penentuan komponen wajah yang telah terdeteksi dengan metode integral proyeksi. (ditunjukkan Gambar 12)

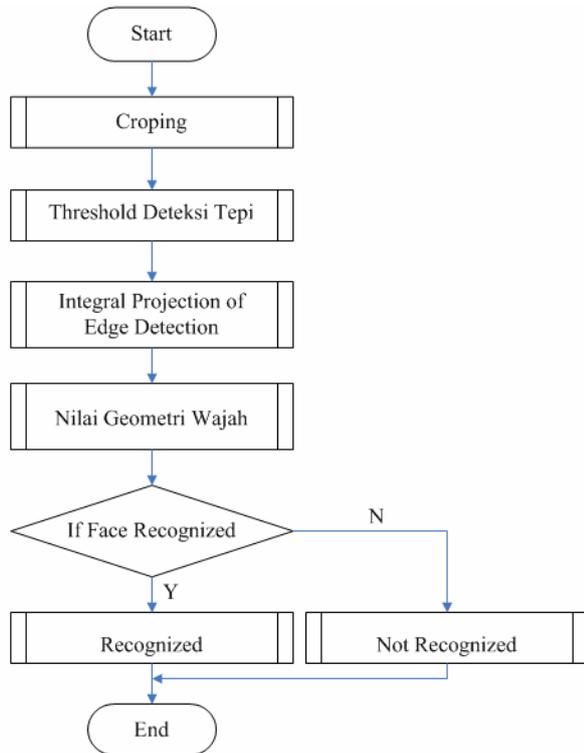
Ada 4 lokasi ciri penting yang bisa didapatkan dari nilai geometri wajah (ditunjukkan Gambar 13)

Flowchart *feature extraction* ditunjukkan Gambar 14. **Instrumen penelitian**

- Hardware : PC, *webcam*.
- software : kamera driver, *visual basic 6.0*.

**Teknik pengumpulan data**

Teknik pengumpulan data dalam penelitian ini adalah data dikumpulkan secara langsung dan secara otomatis data disimpan di dalam database. Data-data yang diambil adalah data fitur-fitur wajah seperti lokasi mata kanan, lokasi mata kiri, lokasi mata dan lebar mulut yang didapatkan dari proses ekstraksi ciri (*feature extraction*). Dari lokasi-lokasi fitur tersebut maka didapatkan fitur 1 jarak dekat antara mata kanan dan mata kiri, fitur 2 jarak jauh antara mata kanan dan mata kiri,



Gambar 14: Flowchart *feature extraction*

fitur 3 lebar mulut dan fitur 4 jarak antara mata terhadap mulut.

**Teknik analisis**

Metode analisis yang digunakan dalam penelitian ini yaitu dengan menghitung nilai rata-rata prosentase keberhasilan sistem yang telah dibuat. Diantaranya yaitu nilai rata-rata keberhasilan *software* mendeteksi wajah (pengambilan obyek / *cropping*), nilai rata-rata keberhasilan *software* mendeteksi fitur-fitur wajah dan mengenali wajah (*face recognition*).

**IV. HASIL DAN DISKUSI**

Data berupa *image* hasil capture. Data diambil dari 6 orang mahasiswa dan masing-masing mahasiswa diambil data sebanyak 10 kali. Sehingga banyak data yang diambil adalah berjumlah 60 data.

Pengujian pada sistem identifikasi yang dikembangkan pada penelitian ini terbagi menjadi 5 (lima) tahap yaitu pengujian tahap *skin color detection*, pengujian tahap deteksi wajah (*cropping*) dengan *integral projection*, pengujian tahap *thresholding edge detection*, pengujian tahap ekstraksi ciri (*feature extraction*) dan pengujian tahap pengenalan wajah (*face recognition*). Namun terlebih dahulu dilakukan tahap pra uji yaitu penentuan filter yang baik digunakan untuk sistem identifikasi melalui nilai MSE dan SNR dari *image* yang diuji.

**A. Tahap pra uji**

Urutan proses tahap pra uji dimulai dari proses *capture image*, kemudian diberi derau yaitu gaussian noise. Kemudian *image* hasil pemberian derau tersebut difilter dengan gaussian filter dan gaussian blur.

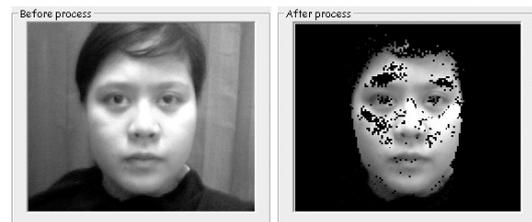
Pada masing-masing data *image* dari 7 (tujuh) orang mahasiswa dilakukan pengukuran nilai MSE dan SNR sebanyak 5 kali kemudian dirata-rata. Data hasil pengukuran nilai MSE dan SNR dari masing-masing *image* dan telah dirata-rata ditunjukkan Tabel 1.

Berdasarkan Tabel 1 dapat disimpulkan bahwa filtering dengan Gaussian blur lebih efektif daripada dengan Gaussian filter, karena pada Gaussian blur mengalami peningkatan nilai SNR dan penurunan nilai MSE sedangkan yang terjadi pada Gaussian filter justru sebaliknya. Oleh karena itu Gaussian blur dipilih sebagai filter pada proses sistem identifikasi.

**B. Tahap pengujian *skin colour detection***

Berikut adalah pengujian *software* untuk tahap *skin color detection* dengan intensitas cahaya yang berbeda pada resolusi 320\*240 :

Percobaan 1: (cahaya terang)



Gambar 15: Proses *skin colour detection*

Percobaan 2: (cahaya normal)



Gambar 16: Proses *skin colour detection*

Percobaan 3: (cahaya gelap)

Berdasarkan hasil di atas maka dapat disimpulkan bahwa intensitas cahaya sangat berpengaruh pada proses *skin colour detection*.

**C. Tahap pengujian *cropping***

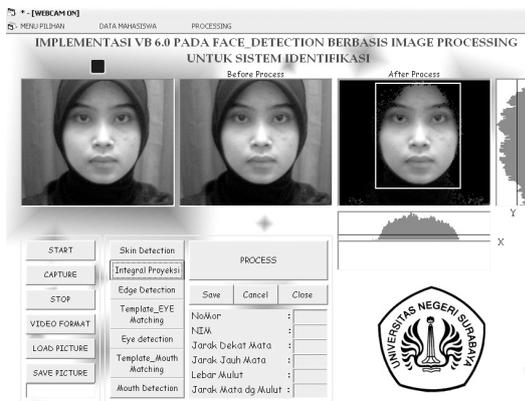
Percobaan 4:

TABEL I: Data nilai rata-rata MSE dan SNR

Probabilitas Noise	Image + noise		Gaussian Filter			Gaussian Blur		
	MSE	SNR (dB)	MSE-Noise	MSE-Filter	SNR (dB)	MSE-Noise	MSE-Filter	SNR (dB)
0,05	0,9631	14,3406	0,4339	0,5661	3,6415	0,7777	0,0698	25,7798
0,5	0,6723	4,8621	0,4358	0,5642	3,6190	0,0648	0,0013	9,5457



Gambar 17: Proses skin colour detection



Gambar 18: Hasil pengujian software cropping

Nilai prosentase menyatakan seberapa besar obyek diambil / wajah dikenali. Jika seluruh wajah terdeteksi maka nilainya 100%, jika hanya setengahnya saja maka nilainya 50% dan jika wajah tidak terdeteksi sama sekali maka nilainya 0. Berikut ini prosentase rata-rata hasil pengujian pengambilan obyek wajah seperti pada Tabel 2.

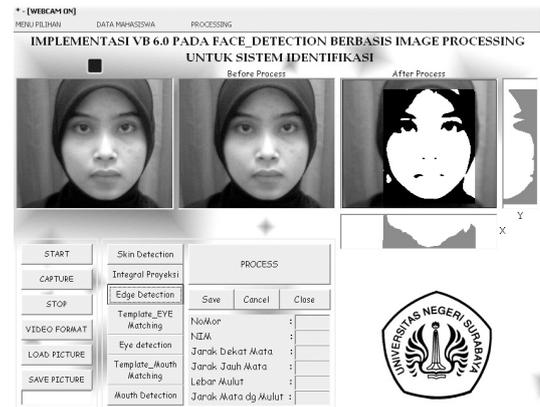
Dari Tabel 2. terlihat bahwa keberhasilan mendeteksi wajah / pengambilan obyek (cropping) wajah adalah 98,5%. Dengan demikian proses ini dapat digunakan untuk mendeteksi wajah.

TABEL II: Prosentase rata-rata cropping

No. Objek	Prosentase cropping
1	100%
2	100%
3	100%
4	100%
5	100%
6	91%
Average	98,50%

D. Tahap pengujian edge detection

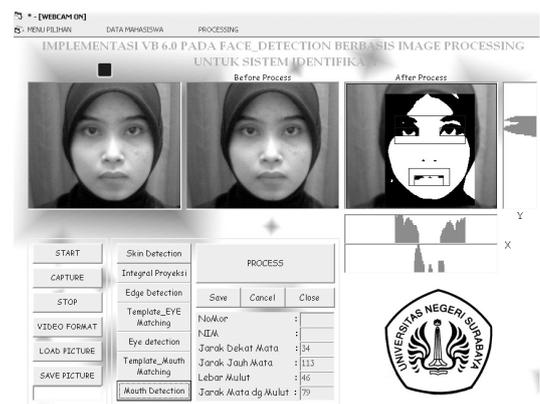
Percobaan 5:



Gambar 19: Hasil pengujian software edge detection dengan filter terpilih

E. Tahap pengujian feature extraction

Percobaan 6:

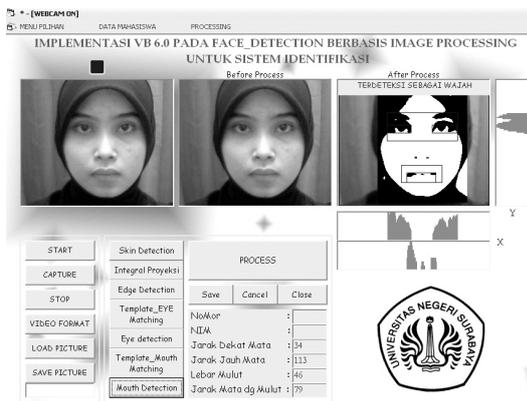


Gambar 20: Hasil pengujian software feature extraction

Dari Tabel 3. terlihat bahwa keberhasilan mendeteksi fitur mata kanan adalah 100%, mendeteksi fitur mata kiri adalah 81,67%, mendeteksi mulut adalah 88,33% dan mendeteksi lebar mulut adalah 85%. Sehingga total keseluruhan keberhasilan mendeteksi fitur / mengekstraksi ciri adalah 88,75%.

**F. Tahap pengujian face recognition**

Percobaan 7:



Gambar 21: Hasil pengujian software face recognition

Berdasarkan tahap pengujian pengenalan wajah terhadap 6 orang mahasiswa sehingga didapatkan data seperti pada Tabel 4.

Dari Tabel 4. ditunjukkan bahwa prosentase keberhasilan pengenalan wajah adalah 81,67%. Hal ini menunjukkan proses sistem identifikasi dapat digunakan dengan baik untuk mengenali wajah.

**V. SIMPULAN**

Dari hasil pengujian pada penelitian ini maka dapat disimpulkan bahwa : " VB 6.0 dapat diimplementasikan pada face detection berbasis image processing untuk sistem identifikasi yaitu dimulai dari proses capture image lalu image discanning guna mendapatkan warna kulit dengan skin color detection. Setelah itu dilakukan proses thresholding edge detection untuk mendapatkan tepi-tepi dari obyek dan memisahkan menjadi dua warna hitam dan putih saja. Dari proses terse-

but maka dapat dilakukan proses deteksi fitur-fitur wajah dengan feature extraction. Dan wajah akan dapat teridentifikasi jika proses feature extraction dapat tepat menentukan lokasi-lokasi fitur-fitur wajah. " Dari data yang diproses diperoleh nilai rata-rata MSE-noise adalah 0,7777 dan MSE-filter adalah 0,0698 dan SNR adalah 25,7796 dB untuk filter terpilih yaitu gaussian blur dengan probabilitas noise 0,05 dan nilai SNR

TABEL III: Prosentase rata-rata Feature extraction

No. Objek	ata kanan	Mata kiri	Mulut	Lebar mulut
1	100%	100%	100%	100%
2	100%	100%	100%	100%
3	100%	40%	60%	40%
4	100%	80%	100%	100%
5	100%	100%	100%	100%
6	91%	70%	70%	70%
Average	100%	81,67%	88,33%	85%

TABEL IV: Prosentase rata-rata Face recognition

No. Objek	Prosentase cropping
1	100%
2	100%
3	40%
4	80%
5	100%
6	70%
Average	81,67%

sebelum filter yaitu 14,3406 dB. Dan hasil yang didapatkan dari sistem identifikasi ini dengan melalui beberapa pengujian dan percobaan yaitu keberhasilan mengambil obyek wajah (cropping) adalah 98,5%, keberhasilan mengekstraksi ciri (feature extraction) adalah 88,75% dan keberhasilan mengenali wajah (face recognition) adalah 81,67%. Dan pada proses pengenalan wajah memiliki dua output yaitu image terdeteksi sebagai wajah dan image terdeteksi bukan wajah.

[1] Basuki, A., Jozua F. P. & Fatchurrochman, *Pengolahan Citra Digital menggunakan Visual Basic* (Edisi pertama, Yogyakarta: Graha Ilmu, 2005).

[2] Mahadityo, B., *Identifikasi Buah Berdasarkan Ciri Warna dan Bentuk*, **Skripsi**, Tidak dipublikasikan, Surabaya: PENS-ITS (2005).

[3] Kuntz, N., 'Canny Tutorial', Available : [http://www.google.co.id] (tanggal akses : 15 Desember 2006) (2006).

[4] Indah, N., *Peramalan Data Time Series Dengan Menggunakan Metode Artificial Neural Network*, **Skripsi**, Tidak dipublikasikan, Surabaya: PENS-ITS (2003).

[5] Sigit, R. , *Sistem Pengenalan Ekspresi Wajah Secara Real Time*, *Tesis*, Tidak dipublikasikan, Surabaya: ITS (2005).

[6] Andayani, D. D., *Otomatisasi Parkir di Politeknik "Menggunakan Sistem Kartu Magnetik dan Monitoring Berbasis PC*, **Skripsi**, Tidak dipublikasikan, Surabaya: PENS-ITS (2002)

[7] Mateos, Gines Garcia, Alberto Ruiz, Pedro E.Lopez-de-Teruel., *Face Detection Using Integral Projection Models*. http://www.google.co.id. Tanggal 29 Juli 2006.

[8] Heisele, Bernd, Thomas Serre, Massimiliano Pontil, Tomaso Poggio. *Component-Based Face Detection*. http://www.google.co.id. Tanggal 14 Juni 2006.

[9] Chuang, M. M., Chang, R. F. & Huang, Y. L. , 'Automatic Facial feature extraction In Model-based Coding', Available : [http://www.google.co.id] (tanggal akses : 29 Juli 2006) (2000).

[10] Gu, H., Su, G. & Du, C., 'Feature Points Extraction from Faces', Available : [http://www.google.co.id] (tanggal akses : 29 Juli 2006)(2006).

[11] Muspa, A. M. , *Sistem Pendeteksi Wajah Secara Real Time*, **Skripsi**, Tidak dipublikasikan, Surabaya: PENS-ITS (2006).

[12] Noortjahja, A. , *Pendekatan Sistem Penalaran Neuro-Fuzzy Adaptive Sebagai Filter Untuk Mereduksi Derau Pada Data*

- Citra Digital*, **Tesis**, Tidak dipublikasikan, Surabaya: ITS (2001).
- [13] Wanti, D. S., *Pengenalan Buah Menggunakan image processing*, **Skripsi**, Tidak dipublikasikan, Surabaya: PENS-ITS(2004).
- [14] The free encyclopedia,'Signal-To-Noise Ratio', Available: [[Http://www.wikipedia.co.id](http://www.wikipedia.co.id)] (tanggal akses : 22 Januari 2007) (2007).