

PENERAPAN METODE SOBEL UNTUK PENGUKURAN TINGGI BADAN MENGGUNAKAN WEBCAM

Ade Noversi Putra, Agus Basukesti, Dwi Nugraheny

Jurusan Teknik Informatika

Sekolah Tinggi Teknologi Adisutjipto Yogyakarta

informatika@stta.ac.id

Abstract

Measurement of human height is usually done with the help of others. But with the Sobel method, measurement of human height can be done alone. It can simplify the process of measuring human height. The method is a development method robert sobel using filter HPF (high pass filter) that given a zero buffer. This method takes the principle of the laplacian and gaussian function is known as a function to generate HPF. Applications designed using sobel method can be used for high mengukur an object with a small error rate. Best distance of 20 cm from the background to the object seen from the results of the error rate.

Keywords : sobel method, measurements of height, webcam

1. Latar Belakang

Untuk mengukur tinggi badan manusia memang memerlukan waktu sebentar apalagi menggunakan teknologi yang serba maju di jaman sekarang. Sebelum teknologi maju, pengukuran tinggi badan seseorang dilakukan secara manual yaitu dengan alat pengukur tinggi badan model lama menggunakan meteran yang proses pengukuran dilakukan secara langsung, dengan cara seseorang mengukur tinggi badan orang lain. Sehingga dapat diukur berapa tinggi badannya, dimulai dari telapak kaki sampai dengan kepala. Tentunya cara tersebut sangat tidak efektif karena diperlukan orang lain untuk mengukur (membaca meteran) untuk mengetahui berapa tinggi badannya.

Adanya kendala tersebut maka dikembangkanlah sebuah aplikasi pengukur tinggi badan menggunakan sistem digital dengan metode *edge detection*. Sehingga proses pengukuran tinggi badan menjadi lebih cepat dan tidak diperlukan lagi orang lain untuk membantu proses pengukuran. Keadaan orang yang akan di ukur posisinya berdiri.

Metode *sobel* merupakan pengembangan metode *robert* dengan menggunakan filter HPF (*high pass filter*) yang diberi satu angka nol penyangga. Metode ini mengambil prinsip dari fungsi *laplacian* dan *gaussian* yang dikenal sebagai fungsi untuk membangkitkan HPF. Kelebihan dari metode *sobel* ini adalah kemampuan untuk mengurangi *noise* sebelum melakukan perhitungan deteksi tepi. Metode *sobel* mengidentifikasi tepi dengan aproksimasi Sobel dengan turunannya. Metode ini menghasilkan atau memperlihatkan tepi pada area dimana gradiennya bernilai maksimum.

2. Perancangan

2.1 Kebutuhan *Hardware* dan *Software*

Perlengkapan pendukung yang meliputi perangkat keras dan perangkat lunak sangat berpengaruh untuk kinerja aplikasi agar dapat berjalan secara maksimal. Oleh karena itu, sangat diperlukan untuk merancang dan menjalankan sebuah sistem aplikasi.

2.1.1 Kebutuhan *Hardware*

Hardware adalah perangkat keras (alat) yang bisa dilihat dan diraba oleh manusia secara langsung yang mendukung proses komputerisasi. Beberapa kebutuhan perangkat keras yang digunakan dalam pembuatan aplikasi *Edge Detection* ini adalah :

1. *Hardware* Yang Digunakan Dalam Membangun Sistem
 - 1) Laptop : Dell N4030 + Intel Dual Core P6200 2.13GHz.
 - 2) Memori (RAM) 2 GB + Harddisk 250 GB + VGA: 763 MB.
 - 3) Mouse.
 - 4) Kamera / Webcam
2. Kebutuhan Minimal *Hardware* Dalam Menjalankan Sistem
 - 1) PC atau Laptop dengan Processor 1 GHZ atau Lebih.
 - 2) Kapasitas Random Access Memory (RAM) 128 MB.
 - 3) Harddisk dengan ruang kosong 100 MB.
 - 4) VGA Card 32 Mb Onboard atau VGA Card.
 - 5) Camera atau Webcam 1,3 Mb Pixel.

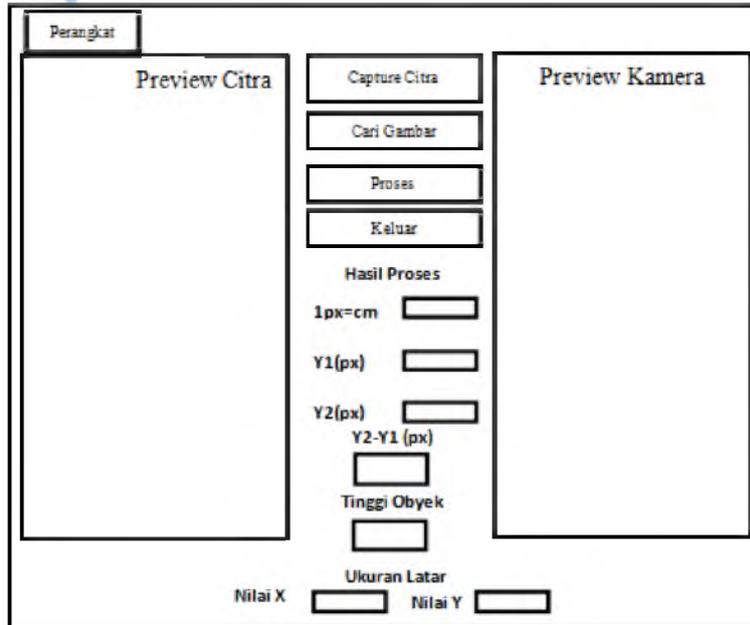
2.1.2 Kebutuhan *Software*

Adapun spesifikasi *software* atau perangkat lunak yang digunakan dalam pembuatan aplikasi *Edge Detection* ini adalah :

1. Sistem operasi *Microsoft Windows 7*, sebagai sistem operasi yang digunakan pada komputer.
2. *Borland Delphi 7*, *software* bahasa pemrograman yang digunakan untuk pembuatan aplikasi.

2.2 Perancangan Antarmuka *Edge Detection*

Perancangan antarmuka *Edge Detection* adalah perancangan tampilan muka yang akan digunakan untuk pengimplementasian dari proses penghitungan citra digital obyek dengan garis tepi. Pada perancangan antarmuka sistem *Edge Detection* terdapat satu *form*. Pada *Form Edge Detection* terdapat lima *button* yang memiliki fungsi untuk memilih kamera (*button* perangkat), untuk mengambil gambar (*button* foto), untuk mencari gambar (*button* cari gambar), untuk mencari garis tepi (*button* proses), dan untuk keluar dari aplikasi (*button* keluar). Selain itu terdapat informasi *input* tampilan obyek di kamera, tampilan citra *digital*, ukuran *pixel*, hasil proses, dan ukuran latar (lihat gambar 3.8).

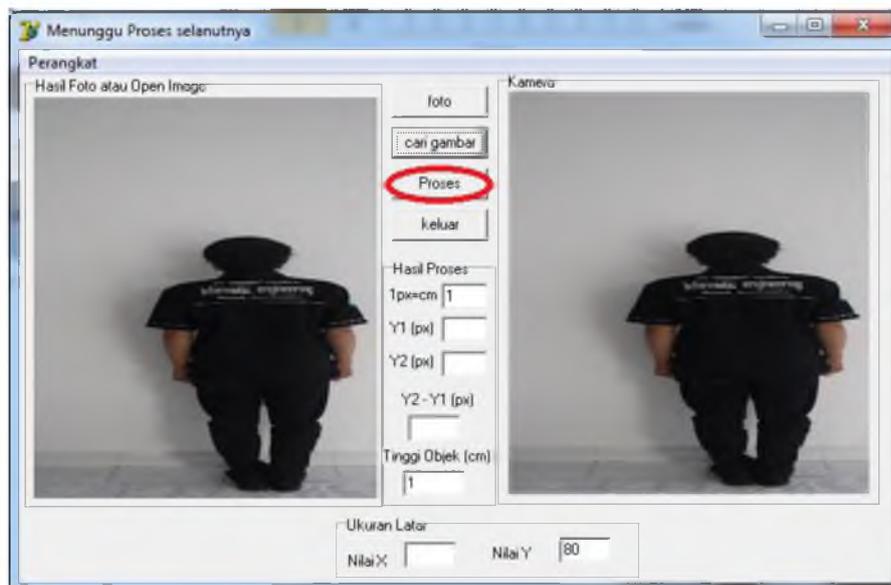


Gambar 3.9 Gambar Form Edge Detection Rancangan

3. Pengujian

3.1 Proses Edge Detection Dengan Sobel

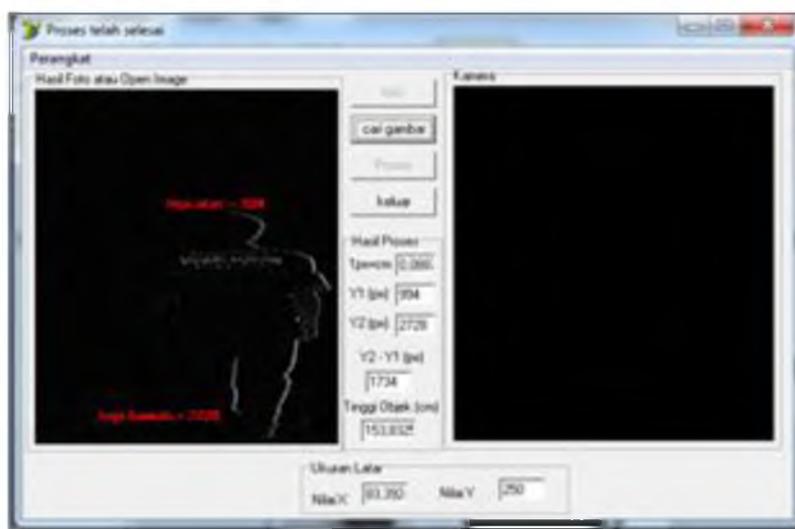
1. Pada saat klik tombol proses maka akan secara langsung citra yang telah di-input-kan akan dikonversi ke citra *grayscale* kemudian dengan *edge detection* metode *sobel* untuk mendapatkan garis tepi. Hasil citra *grayscale* berupa citra warna keabu-abuan pada gambar 3.2, sedangkan *edge detection* hanya berupa garis yang cerah berwarna putih, dan mendapatkan nilai tinggi seperti gambar 3.3, dan gambar 3.4.



Gambar 3.1 Gambar Citra Yang Terpilih



Gambar 3.2 Contoh Citra Digital Yang Dikonversi ke *Grayscale*



Gambar 3.3 Gambar Hasil Deteksi Tepi

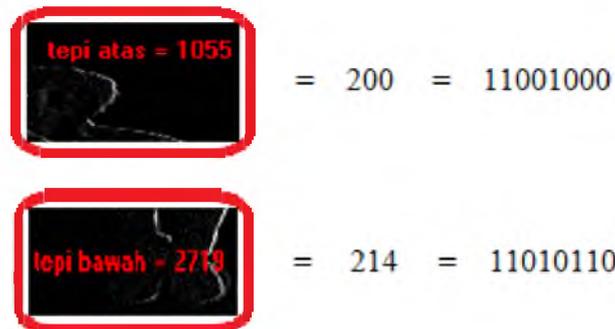


Gambar 3.4 Contoh Untuk Penentuan *pixel* untuk titik tepi atas dan titik tepi bawah

Setelah garis tepi citra digital didapatkan dan tepi atas dan tepi bawah di ketahui dan dikonversi *pixel* ke centimeter maka akan mendapatkan nilai tinggi obyek. Berikut contoh tahapan konversi citra digital ke *grayscale* hingga mendapatkan hasil *biner* kemudian garis tepi dengan metode *sobel* yang ditunjukkan pada gambar 3.5.



Gambar 3.5 Contoh Konversi *Pixel* Dalam Bentuk Matrik *Grayscale*



Gambar 3.6 Contoh Konversi Citra Digital Nilai *Biner* Menjadi Garis Tepi dengan *Sobel*

2. Setelah proses *edge detection* dengan *sobel* selesai, hasil citra dengan format bitmap disimpan dengan otomatis di media penyimpanan di komputer. Citra yang disimpan adalah angka-angka yang didapat dari matrik. Mendapatkan nilai *biner* dari angka-angka matrik pada gambar 3.6 dengan metode *sobel* cara penghitungannya sebagai berikut :

Contoh : $f(2.2) = 214$

95	206	232	50
121	237	210	138
109	28	214	136
121	36	213	149

Gambar 3.7 Gambar Angka-Angka Dengan Format Bitmap

Sobel

Operator Sobel horisontal

-1	0	1
-2	0	2
-1	0	1

Operator Sobel vertikal

-1	-2	-1
0	0	0
1	2	1

Gambar 3.8 Gambar Kernel Operator Sobel Horisontal dan Vertikal

Dari data gambar 3.7 dimasukkan ke dalam matriks Horisontal dan Vertikal

$$M = \sqrt{S_x^2 + S_y^2} \dots\dots\dots(4.1)$$

Horisontal = $K1(x,y) = (-1*237)+(-2*28)+(-1*36)+(1*138)+(2*136)+(1*149) = | - 230 | = 230$

Vertikal = $K2(x,y) = (-1*237)+(-2*210)+(-1*138)+(1*36)+(2*213)+(1*149) = | - 184 | = 184$

Maka f(2,2) bila menggunakan :

$K1(x,y) = (K1(x,y) + K2(x,y)) = 230 + 184 = 414 \approx 255$

$K2(x,y) = \max (K1(x,y), K2(x,y)) = 230$

$K3(x,y) = (K1(x,y) + K2(x,y)) / 2 = (230 + 184) / 2 = 207$

$K4(x,y) = \sqrt{K1(x,y) * K1(x,y) + K2(x,y) * K2(x,y)} = \sqrt{(230*230) + (184*184)} = 230 \approx 255$

Dengan melakukan pengecekan angka jika nilai angka ≥ 128 maka nilai biner yang diberikan adalah 1, jika angka < 128 maka nilai biner adalah 0. Maka nilai biner yang diberikan untuk angka $214 = 255$ adalah 1.

3. Dalam pengujian aplikasi *edge detection* ini dilakukan dengan melakukan tes pengambilan citra dengan latar berwarna putih dengan tinggi latar 250 Cm. Obyek berdiri dari latar dengan jarak 20 Cm. Tinggi kamera 122 Cm, kamera yang digunakan canon 10 mega *pixel* dengan dudukan tripod. Jarak kamera dengan latar 227 Cm.

4. Kesimpulan

1. Aplikasi yang dirancang menggunakan metode sobel dapat digunakan untuk mengukur tinggi suatu benda dengan tingkat kesalahan yang kecil.
2. Jarak terbaik 20 cm dari latar ke obyek dilihat dari proses hasil tingkat kesalahan.
3. Semua kamera dapat digunakan dengan syarat memenuhi kriteria permintaan sistem operasi, driver software harus ada.

5. Daftar Pustaka

- Achmad Basuki, Jozua F. Paladin, dan Fatchurrochman, 2005, *Pengolahan Citra Digital menggunakan Visual Basic*, Andi Offset, Yogyakarta.
- Balza Achmad, Kartika Firdausy, 2005, *Teknik Pengolahan Citra Digital menggunakan Delphi*, Ardi Group, Yogyakarta.
- Bin Ladjamuddin, Albahra, 2005, *Analisis Dan Desain Sistem Informasi*, Graha Ilmu, Yogyakarta.
- Bin Ladjamuddin, 2006, *Rekayasa Perangkat Lunak*, Graha Ilmu, Yogyakarta.
- Darma Putra, 2010, *Pengolahan Citra Digital*, Andi Offset, Yogyakarta.
- Mauridhi Hery Purnomo, Arif Muntasa, 2010, *Konsep Pengolahan Citra Digital dan` Ekstrasi Fitur*, Graha Ilmu, Yogyakarta.
- T.Sutoyo, Edy Mulyanto, Vincent Suhartono, Oky Dwi Nurhayati dan Wijanarto, 2009, *Teori Pengolahan Citra Digital*, Udinus dan Andi Offset, Yogyakarta.