

AUTOMATIC HEAD ROTATING SYSTEM PADA DIGITAL PET MEMANFAATKAN FACE DETECTION

Hadriansa¹ dan Yosi Kristian²

¹Teknik Informatika STMIK PPKIA Tarakanita Rahmawati

²Teknik Informatika Sekolah Tinggi Teknik Surabaya
ansar080489@gmail.com dan yosi@stts.edu

ABSTRAK

Deteksi wajah (face detection) dilakukan dengan menggunakan image asli yang di proses baik secara manual maupun secara real time. Mendeteksi wajah dilakukan dengan mengolah image asli dari hasil pengambilan kamera dengan resolusi standar 640x480.

Mendeteksi image dengan output jumlah wajah yang dihasilkan diproses dengan menggunakan algoritma haar-like cascade yang dikembangkan oleh viola dan jones dengan menggunakan data testing yang sudah dibentuk dalam format xml. Bentuk wajah yang dideteksi merupakan bentuk frontal face. Untuk mengolah image dengan haar-like cascade dilakukan dengan mengubah nilai citra image menjadi warna keabu-abuan (grayscale) kemudian dilakukan interpolasi. Hasil dari interpolasi dilakukan ekualiser histogram yang kemudian dilakukan deteksi jumlah wajah yang terdeteksi.

Automatic head rotating system diterapkan dalam bentuk Robot dengan memanfaatkan face detection Robot berfungsi untuk mendeteksi dan mengenal wajah yang sudah diolah dan mengikuti wajah yang dianggap majikan secara real time.

Kata kunci: *Automatic head rotating system, face detection, haar-like cascade*

ABSTRACT

Face detection is done by using an original image processed either manually or in a real time. Detecting the face is done by processing an original image from the camera taking with a standard resolution of 640x480.

Detecting image with a face amount of output produced processed using haar-like cascade algorithm developed by viola and Jones. vit uses testing data has been established in xml format. The shape of the detected face is a frontal face shape. Process the image with haar-like cascade is done by changing the value of image becoming grayish color (grayscale) then interpolation. After that, equalizer detection histograms area applied on the result of the interpolation. The next step is the detection of detected face number.

Automatic head rotating system is applied in the form of robots by using face detection. The robot is used to detect and recognize faces that have been processed and are considered to follow the employer faces in real time.

Keywords: Automatic head rotating system, face detection, haar-like cascade

I. PENDAHULUAN

Wajah adalah bagian dari muka yang memiliki telinga, hidung, mulut, dan mata yang berfungsi sebagai tanda pengenalan orang, wajah juga biasa disebut dengan bagian depan dari kepala. Pada umumnya wajah digunakan untuk ekspresi wajah, penampilan serta identitas, tidak ada wajah pun yang serupa mutlak. Bahkan pada manusia kembar identik sekalipun

Dalam penelitian ini penulis menggunakan algoritma *haar-like cascade* sebagai pendeteksi wajah (*face detection*) sebagai deteksi wajah. hasil deteksi dilakukan scanning pada digital pet. Digital per merupakan bentuk robot cerdas yang dapat mengikuti bentuk wajah manusia.

II. HAAR LIKE CASCADE

Secara umum *Haar-like feature cascade* digunakan dalam mendeteksi objek pada *image digital*. Nama haar merujuk pada suatu fungsi matematika (*haar wavelet*) yang terbentuk kotak. Awalnya pengolahan gambar hanya dengan melihat dari nilai *RGB (Red Green Blue)* setiap piksel. Namun metode ini ternyata tidaklah efektif. *Viola* dan *jone* yang kemudian mengembangkannya sehingga terbentuk *Haar-like feature*. *Haar like feature* memproses gambar dalam kotak-kotak, di mana dalam satu kotak terdapat beberapa piksel. Dalam setiap kotak kemudian diproses dan didapatkan perbedaan nilai (*threshold*) yang menandakan daerah gelap dan terang. Nilai - nilai *threshold* yang nantinya dijadikan dasar dalam *image processing*.

Pada tahap pre-precessing image dilakukan proses scaling scaling, grayscale, dan thresholding untuk menjadi inputan dalam metode *haar like feature cascade*.

1. Scalling

Scaling merupakan proses mengubah ukuran citra digital, hal ini perlu dilakukan agar semua citra digital memiliki ukuran yang sama. Citra digital tersebut diperkecil dengan menggunakan metode interpolasi, metode ini menggunakan rata-rata suatu region untuk mewakili region tersebut.

Tabel 1. Citra asli

121	159	205	88	71	103	231	134
211	177	67	54	81	92	233	146
183	246	199	56	61	191	211	113
98	72	45	153	215	123	222	100
123	111	232	152	63	44	119	211
186	191	124	45	111	100	206	99

Tabel 1 merupakan nilai citra asli untuk dilakukan proses interpolasi dengan membagi beberapa nilai setiap piksel, misalnya pada nilai piksel 121,159,211,177 akan dilakukan interpolasi sehingga bisa dijadikan 1 nilai piksel dengan menggunakan rumus interpolasi. Hasil dari interpolasi pada citra asli seperti pada tabel 3 di bawah ini.

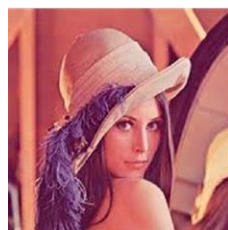
Tabel 3. Citra Interpolasi

167	103,5	86,75	186
149,75	113,25	147,5	161,5
152,75	138,25	79,5	158,75

Nilai piksel pada citra hasil interpolasi diperoleh rata-rata dari 4 nilai piksel citra asli. Dari ukuran 8x6 piksel citra asli dilakukan dengan interpolasi menjadi 4x3 piksel. misalnya pada piksel 121,159,211, dan 117 untuk mendapatkan hasil rata - rata (interpolasi) $(121+159+211+117)/4$, dapat nilai 167, begitu pula dengan nilai piksel - piksel yang lain. Hasil dari perhitungan keseluruhan interpolasi dari citra asli menjadi jadi image interpolasi seperti pada tabel berikut ini.

Tabel 4. Perbandingan Nilai Piksel Citra Asli dan Interpolasi

Nilai piksel citra asli	Nilai piksel cita hasil interpolasi
$(121+159+211+177)/4$	167
$(205+88+67+54)/4$	103,5
$(71+103+81+92)/4$	86,75
$(231+134+233+146)/4$	186
$(183+246+98+72)/4$	149,75
$(199+56+45+153)/4$	113,25
$(61+191+215+123)/4$	147,5
$(211+113+222+100)/4$	161,5
$(123+111+232+191)/4$	152,75
$(232+152+124+45)/4$	138,25
$(63+44+11+100)/4$	79,5
$(119+211+206+99)$	158,75

**(a) 204 x 204****(b) 102x102****Gambar 1. Penskalaan citra dengan metode interpolasi**

2. Grayscale

Citra digital yang telah melalui proses penskalaan kemudian diubah menjadi citra dua warna dengan proses *grayscale*. Proses pengubah citra RGB menjadi citra *grayscale*. Proses pengubahan citra RGB menjadi *grayscale* adalah sebagai berikut. Misalkan suatu citra wajah memiliki nilai :

$$R = 152$$

$$G = 132$$

$$B = 133$$

Maka nilai *grayscale* dari citra tersebut dapat dihitung seperti di bawah ini.

$$W = 0,2989R + 0,5870G + 0,1140B$$

$$W = (0,2989 * 152) + (0,5870 * 132) + (0,1140 * 133) = 138,0788$$

Berikut ini adalah citra hasil *grayscale*:



asli



grayscale

Gambar 2. Perubahan citra RGB menjadi grayscale

3. Cascade classifier

Menurut para sumber buku, wajah merupakan salah satu bagian dari manusia yang memiliki ciri berbeda untuk setiap manusia. Pada penelitian ini digunakan metode *haar cascade classifier* sebagai metode untuk pengenalan wajah. Ada beberapa tahapan untuk menentukan pola wajah adalah sebagai berikut.

a. Proses menentukan *haar-like feature*

Haar-like fitur yaitu selisih dari jumlah piksel dari daerah di dalam persegi panjang. Contoh *haar-like feature* seperti pada gambar di bawah ini.



Gambar 3. Contoh Haar-like feature

Nilai *Haar-like feature* diperoleh dari selisih jumlah nilai piksel daerah gelap dengan jumlah piksel daerah terang

$$F(Haar) = \sum F_{White} - \sum F_{Black} \dots\dots\dots (1)$$

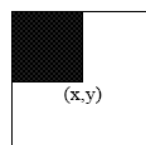
$$F(Haar) = \text{Nilai Fitur total} \dots\dots\dots (2)$$

$$\sum F_{White} = \text{Nilai fitur pada daerah terang} \dots\dots\dots (3)$$

$$\sum F_{Black} = \text{Nilai fitur pada daerah gelap} \dots\dots\dots (4)$$

b. Integral Image

Integral image yaitu suatu teknik menghitung nilai fitur secara cepat dengan mengubah nilai dari setiap piksel menjadi suatu representasi citra baru, sebagaimana disajikan pada gambar berikut.



Gambar 4. Integral image

Berdasarkan gambar 3. citra integral image pada titik pada titik (x, y) ($ii(x, y)$) dapat dicari menggunakan persamaan

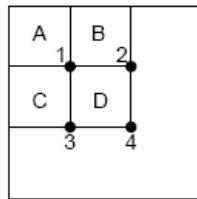
$$ii(x, y) = \sum_{x' \leq x, y' \leq y} i(x', y') \dots\dots\dots (5)$$

Keterangan

$ii(x, y)$ = Citra integral pada lokasi x, y

$i(x', y')$ = nilai piksel pada citra asli

Perhitungan nilai dari suatu fitur dapat dilakukan secara cepat dengan menghitung nilai citra integral pada empat buah titik seperti pada pada gambar berikut.



Gambar 5. Perhitungan Nilai Fitur

Jika nilai integral image titik 1 adalah A, titik 2 adalah A+B, titik A+C dan titik 4 adalah A+B+C+D, maka jumlah piksel di daerah D dapat diketahui dengan cara $4+1 - (2+3)$.

c. Algoritma adaboost learning

Algoritma *adaboost learning* digunakan untuk meningkatkan kinerja klasifikasi dengan pembelajaran sederhana untuk menggabungkan banyak *classifier* lemah menjadi satu *classifier* kuat. *Classifier* adalah suatu jawaban benar dengan tingkat kebenaran yang kurang akurat. Sebuah *classifier* lemah dinyatakan.

$$h_j(x) = \begin{cases} 1, & \text{jika } p_j f_j(x) < p_j \theta_j(x) \\ 0, & \text{jika lainnya} \end{cases} \dots\dots\dots (6)$$

$h_j(x)$ adalah klasifikasi lemah, adalah *parity* ke j , θ_j adalah *threshold* ke j dan x adalah dimensi sub *image*, misalnya 24×24 . Langkah-langkah untuk menentukan *classifier* yang kuat dinyatakan dalam suatu algoritma sebagai berikut:

Diberikan contoh gambar $(x_1, y_1), \dots, (x_n, y_n)$ dimana $y_i = 0$ untuk contoh positif dan $y_i = 1$ untuk contoh negatif, inialisasi bobot $y_{i,1} = \frac{1}{2m}, \frac{1}{21}$ dan l adalah jumlah negatif dan positif. Untuk $t = 1, \dots, T$

- a. Menormalkan bobot sehingga w_t adalah distribusi probabilitas

$$w_{t,i} \leftarrow \frac{w_{t,i}}{\sum_{j=1}^n w_{t,j}} \dots\dots\dots (7)$$

- b. Untuk setiap fitur, j melatih *classifier* h_j , untuk setiap fitur tunggal.

- c. Kesalahan (ϵ_j) dievaluasi dengan bobot w_t

$$\epsilon_j = \sum_i w_i |h_i(x_i) - y_i| \dots\dots\dots (8)$$

- d. Pilih *classifier* h_t dengan error terkecil dimana $e_i = 0$ untuk x_i adalah klasifikasi benar, dan $e_i = 1$ untuk yang lain.
 e. Perbaharui bobot:

$$w_{t+1,i} = w_{t,i} \beta^{1-e_i} \dots \dots \dots (9)$$

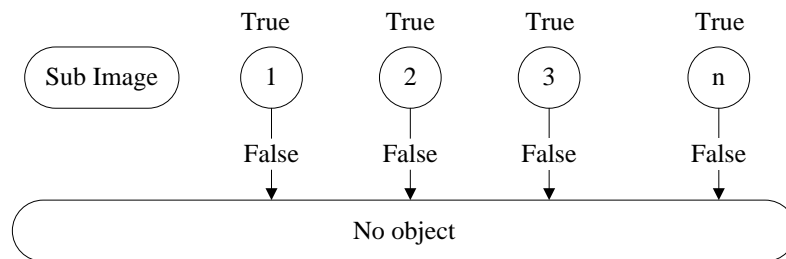
Dimana $\beta_t = \frac{e_t}{1-e_t}$

- f. Dapatkan sebuah *classifier* kuat yaitu

$$h(x) = \begin{cases} 1, & \sum_{t=1}^T \alpha_t h_t(x) \geq \frac{1}{2} \sum_{t=1}^T \alpha_t \\ 0, & \text{lainnya} \end{cases} \dots \dots \dots (4)$$

dimana $\alpha_t = \log \frac{1}{\beta_t}$

cascade classifier adalah sebuah metode untuk mengkombinasi *classifier* yang kompleks dalam sebuah struktur bertingkat yang dapat meningkatkan kecepatan pendeteksian objek dengan menfokuskan pada daerah citra yang berpulang saja, Struktur *cascade classifier* seperti pada gambar berikut.



Gambar 6. Cascade Clasifier

Hasil pendeteksian wajah dengan *haar like cascade clasifier* dapat dilihat seperti pada gambar berikut.



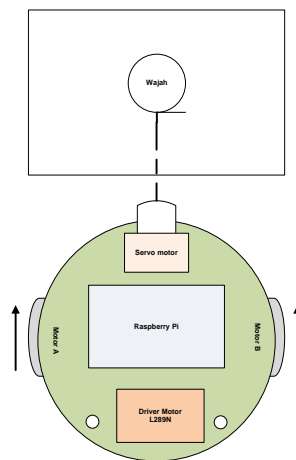
Gambar 7. Hasil deteksi wajah

III. PRINSIP KERJA ROBOT

Robot bekerja dengan menggunakan penglihatan seperti manusia yang dapat melihat objek. Manusia dapat melihat dengan menggunakan mata sedangkan robot dapat melihat dengan menggunakan kamera, manusia dapat melihat jauh secara normal apabila mata manusia dalam kondisi normal. Robot juga dapat melihat dengan jauh terhadap objek yang dikenal sesuai dengan resolusi kamera yang digunakan. Bentuk pergerakan robot terhadap wajah yang dikenal sebagai berikut.

1. Pergerakan dengan posisi wajah di tengah window

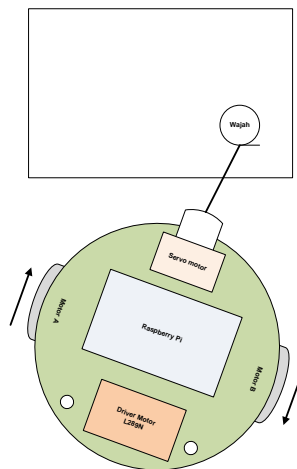
Bentuk pergerakan robot dengan posisi wajah di tengah window, robot bergerak lurus dengan dengan kecepatan Roda motor A dan roda motor B sama, sedangkan posisi motor servo pada leher sama, dan posisi motor servo kepala mengikuti nilai jarak wajah yang dideteksi seperti pada gambar berikut.



Gambar 8. Robot bergerak lurus

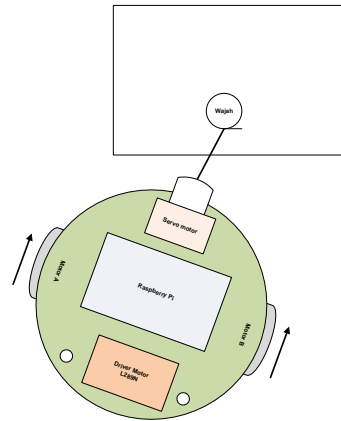
2. Pergerakan dengan posisi pergeseran wajah ke kanan

Pada pergerakan ini di mana wajah majikan bergeser ke kanan dari posisi tengah, robot bergerak dan mengikuti pergerakan wajah majikan seperti pada gambar berikut.



Gambar 9. Robot bergerak ke kanan

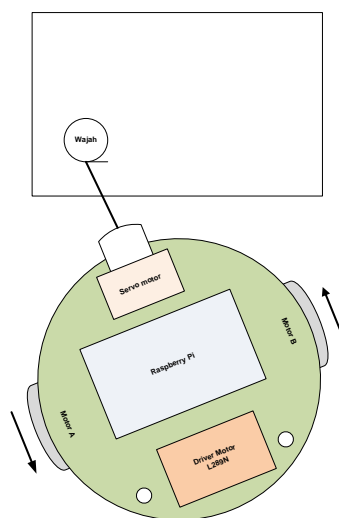
Jika pergerakan wajah yang dideteksi robot bergerak mengikuti wajah. pada posisi pergerakan motor A berputar searah dengan jarum jam (maju) dan motor B berputar berlawanan dengan jarum jam (mundur). Pergerakan motor servo leher dan juga motor servo kepala mengalami pergerakan. Pada pergerakan robot tidak hanya perubahan pergerakan motor DC dan motor servo tetapi juga deteksi terhadap objek berpengaruh terhadap kamera, karena lebar dan tinggi jendela window terhadap kamera terbatas. Perubahan kamera jendela window dapat dilihat seperti pada gambar berikut.



Gambar 10. Perubahan window terhadap kamera

3. Pergerakan dengan posisi pergeseran wajah ke kiri

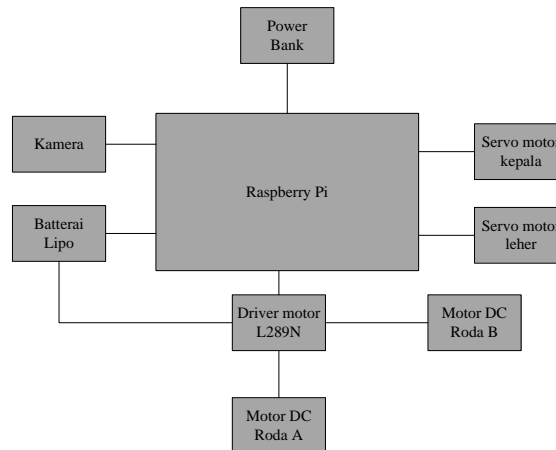
Pergerakan ini di mana wajah bergeser ke kiri dari posisi tengah, pergerakan robot sama seperti pergerakan posisi ke kanan, yang membedakan hanya perputaran motor pada robot begitu pula dengan pergerakan servo motor leher dan servo motor kepala. Robot bergerak dan mengikuti pergerakan wajah seperti pada gambar berikut.



Gambar 11. Robot bergerak ke kiri

IV. IMPELEMENTASI ROBOT BERODA

Pada bagian ini menggambarkan tentang cara melakukan *scanning* wajah dengan cara pergerakan mengikuti wajah yang terdeteksi. Robot bekerja mencari dan mengikuti wajah yang terdeteksi dengan menggunakan motor DC, Motor Servo dan Raspberry PI, serta Webcam, dari keempat perangkat keras tersebut yang paling penting adalah Raspberry Pi. Karena Raspberry Pi merupakan Otak dari robot tersebut di mana dalam melakukan pergerakan dikendalikan menggunakan Raspberry Pi. Berikut bentuk rangkaian Robot



Gambar 8. Rangkaian robot beroda

Untuk melakukan *scanning* wajah ada beberapa tahapan yang dilakukan yaitu dengan menentukan posisi objek wajah yang terdeteksi. Robot bekerja dengan menggunakan penglihatan seperti manusia yang dapat melihat objek. Manusia dapat melihat dengan menggunakan mata sedangkan robot dapat melihat dengan menggunakan kamera, Manusia dapat melihat jauh secara normal apabila mata manusia dalam kondisi normal. Robot juga dapat melihat dengan jauh terhadap objek yang dikenal sesuai dengan resolusi kamera yang digunakan. Bentuk pergerakan robot terhadap wajah yang deteksi sebagai berikut.

1. Pergerakan dengan posisi wajah yang terdeteksi di tengah window

Bentuk pergerakan robot dengan posisi wajah di tengah window, robot bergerak lurus dengan dengan kecepatan Roda motor A dan roda motor B sama, sedangkan posisi motor servo pada leher sama, dan posisi motor servo kepala mengikuti nilai jarak wajah yang dideteksi seperti pada gambar berikut.



Gambar 9. Robot bergerak lurus

2. Pergerakan dengan posisi pergeseran wajah ke kanan
Pada pergerakan ini di mana wajah yang terdeteksi bergeser ke kanan dari posisi tengah, robot bergerak dan mengikuti pergerakan wajah yang terdeteksi seperti pada gambar berikut.



Gambar 10. Robot bergerak ke kanan

3. Pergerakan dengan posisi pergeseran wajah ke kiri
Pergerakan ini di mana wajah yang terdeteksi bergeser ke kanan dari posisi tengah, pergerakan robot sama seperti pergerakan posisi kanan, yang membedakan hanya perputaran motor pada robot begitu pula dengan pergerakan servo motor leher dan servo motor kepala. Robot bergerak dan mengikuti pergerakan wajah yang terdeteksi seperti pada gambar berikut.



Gambar 6. Robot bergerak ke kiri

IV. PENUTUP

Dalam penelitian dapat ditarik kesimpulan bahwa *facetedetection* dengan menggunakan *haar like cascade* dapat diselesaikan dengan baik dengan cepat, kemampuan dalam mendeteksi image wajah tergantung dengan nilai ukuran piksel yang dihasilkan. Robot dapat melakukan scanning wajah secara acak tanpa membatasi nilai wajah yang terdeteksi. Implementasi *haar like cascade* berjalan dengan baik dengan menggunakan komputer mini Raspberry Pi. Dengan kamera standar ukuran 640 x 480 dapat mengolah image dan mendeteksi objek wajah dengan jarak 1 sampai 2 meter. Selain itu nilai *grayscale* terhadap *image* berpengaruh pada proses deteksi wajah. Untuk pengembangan penelitian ini sebaiknya gunakan kamera dengan resolusi lebih besar, gunakan perangkat hardware yang lebih tinggi Seed-nya seperti Eda Board.

V. DAFTAR PUSTAKA

- [1] Paul Viola, Michael Jones. 2001. *Rapid Object Detection using a Boosted Cascade of Simple Features*. IEEE
- [2] R. Lienhart and J. Maydt. *An Extended Set of Haar-like Features for Rapid Object Detection*. In *Int. Conf. on Image Processing (ICIP'01)*, pages 73–76, Thessaloniki, 2001
- [3] Gary Bradski & Adrian Kaehler O'Reilly (2008), “*Learning OpenCV*”, O'REILLY Media.
- [4] Z. Zhang (2008), “*Implementation and Research of Embedded Face Detection using Adaboost*”, Shanghai JiaoTong University, China
- [5] Z. Zeng, Y. Fu, G. I. Roisman, Z. Wen, Y. Hu, and T. S. Huang, “Spontaneous emotional facial expression detection,” *J. Multimedia*, vol. 1, no. 5, pp. 1–8, Aug. 2006.
- [6] M. Richardson and S. Wallace, “*Getting Started with Raspberry Pi*”, 1st ed. O'Reilly, Media, U.S.A: 2012,
- [7] R. P. Website. The making of pi, 2014. <http://www.raspberrypi.org/about/>
- [8] Open Computer Vision Library [Electronic resource], http://sourceforge.net/projects/opencvlibrary/files/opencv_2012
- [9] Raspberry Pi camera module setup www.raspberrypi.org/help/camera-module-setup/