

SISTEM PENGENALAN WAJAH DAN MATA DENGAN METODE HAAR CASCADE CLASSIFIER

Fariha 'Ulya Rahma
Fakultas Sains dan Teknologi
Program Studi Teknik Informatika
Universitas Darussalam Gontor Kampus Putri
Sambirejo, Mantingan, Ngawi, Jawa Timur 63260
fariharahma@mhs.unida.gontor.ac.id

Abstract---teknologi pengenalan wajah mulai berkembang dengan adanya sistem semi otomatis yang dapat menemukan fitur utama pada wajah yaitu mata, hidung, telinga, mulut didalam foto tersebut dengan menghitung jarak rasio dari satu titik ke referensi global. Deteksi wajah adalah langkah awal dalam proses pengenalan wajah yang memiliki pengaruh besar pada kinerja dari seluruh sistem pengenalan wajah. Penelitian ini menggunakan OpenCV pada pemrograman python untuk mengenali wajah dan mata seseorang menggunakan Haar Cascade. Penelitian ini bertujuan untuk mengimplementasikan computer vision dengan memanfaatkan library yang ada didalamnya.

Keyword---Haar Cascade Classifier, OpenCV, python.

I. PENDAHULUAN

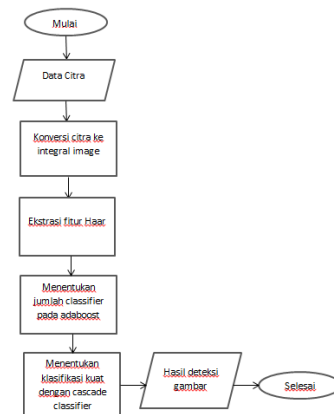
Pengenalan wajah merupakan salah satu teknik dari sistem identifikasi berbasis biometrik yang dikembangkan berdasarkan perbedaan ciri wajah seseorang sebagai parameter utamanya[1]. Sistem biometrik adalah teknologi pengenalan dengan menggunakan bagian tubuh atau perilaku manusia yang memiliki keunikan. Identifikasi biometrik didasarkan pada karakteristik alami manusia, yaitu karakteristik fisiologis seperti wajah, sidik jari, suara, telapak tangan, dan retina mata. Wajah dapat mengidentifikasi seseorang karena wajah adalah objek dinamis yang memiliki tingkat variabilitas yang tinggi (Choong dkk, 1996). Teknologi pengenalan wajah mulai berkembang dengan adanya sistem semi otomatis yang dapat menemukan fitur utama pada wajah yaitu mata, hidung, telinga, dan mulut didalam foto tersebut dengan menghitung jarak dan rasio dari satu titik ke referensi global[2,3]. Deteksi wajah adalah langkah awal dalam proses

pengenalan wajah yang memiliki pengaruh besar pada kinerja dan kegunaan dari seluruh sistem pengenalan wajah. Salah satu metode deteksi wajah adalah Haar-like feature yang dikemukakan oleh Viola & Jones. Metode ini terbagi menjadi 3 komponen utama, diantaranya Integral Image, Adaptive Boosting, dan Cascade of Classifier [4]. Pendeteksian mata dan wajah mengambil data yang telah disediakan oleh Intel yaitu Haar Cascades (koordinat (x,y)) yang mengidentifikasi wajah dan mata dengan pixel pada kamera. Perhitungan numeric menggunakan numpy modul pada python agar dapat menghitung koordinat dengan tepat pada wajah dan mata [5].

II. METODE

A. Proses Dataset

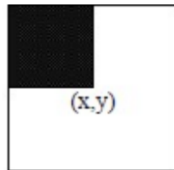
Penelitian ini menggunakan dataset dari pengambilan citra menggunakan gambar dan classifier dalam bentuk xml file untuk deteksi wajah dan proses pengenalan. Selanjutnya melakukan scanning dataset dan dilakukan konversi dengan integral image. Kemudian dihasilkan nilai baru untuk ekstraksi fitur haar. Berikut penjelasan alur proses pendeteksian wajah.



Gambar 1. Flowchart Proses Pendeteksian Wajah

B. Integral Image

Integral image yaitu teknik untuk menghitung nilai fitur secara cepat dengan mengubah nilai dari setiap piksel menjadi suatu representasi citra baru. Nilai untuk masing-masing piksel adalah jumlah semua piksel dari atas sampai bawah. Dimulai dari kiri atas sampai kanan bawah. Seperti contoh gambar dibawah ini.[6]



Gambar 2. Gambar Integral (x,y)

C. Haar Features

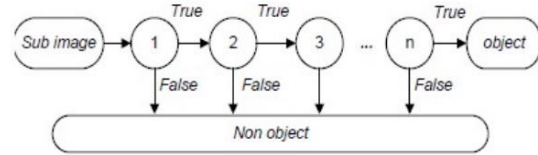
Haar Feature adalah fitur yang mendasar pada *Wavelet Haar*. *Wavelet Haar* adalah gelombang tunggal bujur sangkar (satu interval tinggi dan satu interval rendah). Untuk dua dimensi satu terang dan satu gelap. Seluruh nilai haar feature dengan jenis atau tipe edge features mendapatkan selisih dari jumlah nilai piksel daerah dengan jumlah nilai piksel dari daerah terang dengan rumus :

$$F(\text{Haar}) = \sum F_{\text{white}} - \sum F_{\text{black}}$$

$F(\text{Haar})$ adalah nilai fitur total, $\sum F_{\text{white}}$ adalah nilai fitur daerah terang dan $\sum F_{\text{black}}$ adalah nilai fitur daerah gelap. [7]

D. Cascade Classifier

Cascade classifier adalah sebuah rantai stage classifier, dimana setiap stage classifier digunakan untuk mendeteksi apakah didalam image sub window terdapat obyek yang diinginkan (object of interest). Stage classifier dibangun dengan menggunakan algoritma adaptive-boost (AdaBoost). Algoritma tersebut mengkombinasikan performance banyak weak classifier untuk menghasilkan strong classifier. Weak classifier dalam hal ini adalah nilai dari haar-like feature. Jenis AdaBoost yang digunakan adalah Gentle AdaBoost. Contoh seperti gambar dibawah ini.[6]



Gambar 3. Cascade Classifier

Pengenalan wajah merupakan implementasi dari computer vision, yaitu computer yang dapat menganalisa citra dengan pendeteksian objek wajah dalam sebuah gambar. Pendeteksian objek wajah memiliki banyak metode yang dikembangkan oleh para ahli. Penelitian tersebut menggunakan metode pendeteksi objek yang populer yaitu metode Viola Jones yang berjudul "*Rapid Object Detection Using a Boosted Cascade of Simple Features*" pada tahun 2001[8]. Metode Viola Jones memiliki empat tahap pendeteksian objek, yaitu fitur segi empat sederhana yang disebut fitur Haar, internal image untuk pendeteksian fitur secara cepat, metode machine learning AdaBoost, dan Cascade Classifier untuk menghubungkan banyak fitur algoritma pada sistem.

haarcascade_eyetracker.xml	some attempts to tune the performance	2 years ago
haarcascade_eye.xml	some attempts to tune the performance	2 years ago
haarcascade_eye_tree_eyeglasses.xml	for face landmarks	2 years ago
haarcascade_frontalface_alt.xml	for face landmarks	2 years ago
haarcascade_frontalface_alt2.xml	some attempts to tune the performance	2 years ago
haarcascade_frontalface_alt3.xml	some attempts to tune the performance	2 years ago
haarcascade_frontalface_default.xml	some attempts to tune the performance	2 years ago
haarcascade_frontalface_oldnns.xml	Some real-time face	2 years ago
haarcascade_frontalface_oldnn.xml	some attempts to tune the performance	2 years ago
haarcascade_frontalface_oldnn2.xml	Added face results for support and limited pose detection. 10 stages	2 years ago
haarcascade_frontalface_oldnn3.xml	Some real-time face	2 years ago
haarcascade_profileface.xml	some attempts to tune the performance	2 years ago
haarcascade_righteye_2splits.xml	some attempts to tune the performance	2 years ago
haarcascade_righteye_2splits2.xml	some attempts to tune the performance	2 years ago
haarcascade_righteye_2splits3.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits4.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits5.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits6.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits7.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits8.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits9.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits10.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits11.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits12.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits13.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits14.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits15.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits16.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits17.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits18.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits19.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits20.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits21.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits22.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits23.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits24.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits25.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits26.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits27.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits28.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits29.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits30.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits31.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits32.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits33.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits34.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits35.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits36.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits37.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits38.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits39.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits40.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits41.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits42.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits43.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits44.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits45.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits46.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits47.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits48.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits49.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits50.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits51.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits52.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits53.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits54.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits55.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits56.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits57.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits58.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits59.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits60.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits61.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits62.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits63.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits64.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits65.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits66.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits67.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits68.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits69.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits70.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits71.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits72.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits73.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits74.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits75.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits76.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits77.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits78.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits79.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits80.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits81.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits82.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits83.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits84.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits85.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits86.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits87.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits88.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits89.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits90.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits91.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits92.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits93.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits94.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits95.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits96.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits97.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits98.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits99.xml	Some real-time face	2 years ago
haarcascade_righteye_2splits100.xml	Some real-time face	2 years ago

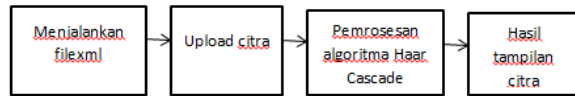
Gambar 4. Daftar File XML Haar Cascade

Pada gambar diatas merupakan file pembaca bagian tubuh manusia. Untuk mendeteksi wajah dan mata maka file yang akan digunakan adalah `haarcascade_frontalface_default.xml` dan `haarcascade_eye.xml`.

III. HASIL

Implementasi dataset dalam file xml berbentuk teks sehingga bila diperlukan untuk membacanya tidak memerlukan bantuan software. File xml dapat menyimpan data lebih baik dalam atribut maupun sebagai isi elemen yang diletakkan diantara tag pembuka dan tag penutup[9]. File `haarcascade_frontalface_default.xml` dan `haarcascade_eye.xml` mempunyai fungsi sebagai sistem mendeteksi wajah untuk diidentifikasi. Selanjutnya upload gambar sebagai sample pendeteksi wajah dan mata, kemudian melakukan

pemrosesan algoritma haar cascade. Lalu akan muncul tampilan gambar hasil akhir. Berikut contoh alur proses pengambilan data wajah.



Gambar 5. Alur pengambilan data wajah

Langkah awal dalam melakukan pendeteksian wajah adalah melakukan training dataset untuk menyimpan beberapa klasifikasi yang membentuk model wajah. Kemudian memilih gambar untuk mendeteksi wajah dan mata. Gambar tersebut berupa format file jpg, dan png.



Gambar 6. Sample data citra 1

Gambar 6 merupakan contoh sample citra 1 yang telah melewati proses algoritma haar cascade, uji coba dalam sample 1 memiliki tingkat keberhasilan 100% dengan menampilkan deteksi pada dua wajah dan mata dalam gambar tersebut. Keberhasilan dalam gambar tersebut mendukung karena adanya cahaya yang baik dan objek wajah yang jelas menghadap kedepan (frontal), dan tidak terhalangi objek lain terhadap apapun. Kemudian contoh sample kedua seperti gambar dibawah ini.



Gambar 7. Sample data citra 2

Gambar 7 merupakan contoh citra yang telah di uji coba dengan tingkat keberhasilan 80% karena tidak semua wajah dan mata dalam gambar tersebut terdeteksi. Kegagalan dalam deteksi ini karena faktor bentuk sebagian wajah yang tertutupi hijab, wajah yang menggunakan kacamata, dan wajah yang kurang jelas serta tidak menghadap kedepan (frontal).

IV. KESIMPULAN

Berdasarkan hasil penelitian yang dilakukan terdapat kesimpulan bahwa proses deteksi wajah dapat berhasil jika beberapa pendukung dapat terpenuhi seperti pencahayaan gambar, objek wajah yang jelas menghadap kedepan, tidak terhalang benda apapun. Sebaliknya jika ada kegagalan dalam pendeteksian wajah tersebut juga karena beberapa faktor seperti minimnya cahaya, objek wajah yang kurang jelas atau tidak menghadap kedepan, sudut pengambilan yang kurang pas. Selain itu juga terdapat gambar yang memiliki noise (titik-titik kecil yang muncul) saat pengambilan gambar pada malam hari.

DAFTAR PUSTAKA

- [1] Li, S. Z. & Jain, A. K., "Handbook of Face Recognition", Springer, US, 2005.
- [2] E. P. Wibowo, B. Pinandito, B. A. Wardijono, dan A. Bagus, "Designing and Implementing System of Real Time Face Detection and Recognition Based on RBF", Asian Transactions on Computers, vol. 1, no. 4, 2011.
- [3] Robinson M, Escarra M, Krueger J, & Kochelek D, "Face Recognition using Eigenfaces", OpenStax CNX, <http://cnx.org/content/col10254/1.2/>, 2004.
- [4] Anonim, "What is Facial Recognition?- Definition from Techopedia", <https://www.techopedia.com/definition/32071/facial-recognition>, 06 Oktober 2021.
- [5] E. Maria, Yulianto, Y. P. Arinda, Jumiatty, and P. Nobel, "Segmentasi Citra Digital Bentuk Daun Pada Tanaman Di Politani Samarinda Menggunakan Metode Thresholding," Jurti, vol. 2, no. 1, pp. 37-46, 2018.

- [6] Prathivi, Rastri, et al. (2020). Sistem Presensi Kelas Menggunakan Pengenalan Wajah Dengan Metode Haar Cascade Classifier
- [7] Puspaningrum, Eva Y., et al. (2018). Deteksi Wajah Dengan Boosted Cascade Classifier
- [8] P Viola, M Jones. (2001). Rapid Object Detection using a Boosted Cascade of Simple Features. Kauai : IEEE.
- [9] Utami, Febiannisa, et al. (2021). Implementasi Algoritma Haar Cascade Pada Aplikasi Pengenalan Wajah.
- [10] Piarsa I N, Kadek S W. (2017). Prototipe Deteksi Dan Pengenalan Wajah Pada Sistem Monitoring Dan Kontrol Visual Keamanan Rumah. Bali : Seminar Nasional Sains dan Teknologi (Senastek) IV.
- [11] Zulkhaidi, Tengku. C.A., et al. (2019). Pengenalan Pola Bentuk Wajah dengan OpenCV.
- [12] I. Urifan, R. Hidayat, & I. Soesanti, "Pengenalan wajah dengan metode Eigenface", Jurnal penelitian teknik elektro, vol. 3, pp. 320-323, Desember 2010.
- [13] Detila, Qadrisa M., et al. (2019). Perbandingan Eigenface, Fisherface, dan LBPH pada Sistem Pengenalan Wajah.