

Identifikasi Citra Wajah Menggunakan Algoritma *Eigenface*

Andre Mochammad Satrio¹⁾, Mohammad Mujirudin²⁾, Harry Ramza³⁾

^{1,2,3)}Program Studi Teknik Elektro, Fakultas Teknik

Universitas Muhammadiyah Prof. Dr. HAMKA

Jalan Tanah Merdeka No 6, Kp Rambutan, Jakarta, Indonesia

Telp : +62-21-8400341, Faks : +62-21-8411531, ¹⁾Email : andremochammadsatrio81@gmail.com,

²⁾mujirudin@uhamka.ac.id, ³⁾hramza@uhamka.ac.id

Abstrak – Pada makalah ini, dipaparkan mengenai pengolahan citra digital menggunakan pendekatan Analisa Komponen Utama dan Algoritma Viola Jones sebagai pendeteksi wajah. Banyak citra yang dilatih pada penelitian ini yaitu 1500 citra wajah. Citra tersebut terdiri atas 100 individu (17 wanita dan 83 pria) dimana masing-masing individu memiliki 15 citra wajah. Selain itu, terdapat total citra yang diujikan sebanyak 500 citra wajah dimana jumlah citra tersebut dibagi oleh 100 individu sehingga masing-masing individu memiliki citra yang dilatih sebanyak 5 citra. Metode yang digunakan yaitu citra wajah dideteksi dengan algoritma Viola-Jones kemudian nilai *Eigenface* citra latih dibandingkan dengan citra uji menggunakan *Eulidean Distance*. Hasil yang diperoleh yaitu tingkat akurasi identifikasi citra wajah mencapai 89.2%, ketika citra latih berjumlah 100 citra dan latar belakang citra wajah tidak dieliminasi. Namun ketika latar belakang citra wajah dieliminasi, tingkat akurasi identifikasi meningkat menjadi 98.6%. Lain halnya jika citra latih berjumlah 1500 tingkat akurasi identifikasi citra wajah meningkat menjadi 100%.

Kata kunci: Pengolahan Citra Digital, Pendekatan Analisa, Analisa Komponen, Algoritma *Eigenface*, Algoritma Viola-Jones.

Abstract – This paper presents the processing digital image by using the principal component analysis approach and viola jones algorithm as detection of the face imagery. The amount of the image-training is 1500 images of 100 individual's face (17 women and 83 men). In addition, there are total images tested the amount of 500 images the face and it will be divided by 100 individuals so that each individual having image-training as many as 5 images. The methodology uses the Viola-Jones Algorithm to detect the face of images then the *eigenface* value of image-training is compared to the image of trials uses *Euclidean distance*. The research result is the accuracy level of the face imagery amount of 89.2% when the image-training around 100 images and the background of the face's imagery are not eliminated. Nevertheless, when the background of the face imagery is eliminated then the accuracy level increase into 98.6%. Another thing, if the image-training amounts 1500 the accuracy level of the face imagery increase being 100%.

Keywords: Digital Image Processing, Principle Component Analysis, *Eigenface* Algorithm, Viola-Jones Algorithm.

1 Pendahuluan

Dalam keseharian, wajah merupakan salah satu pengenalan seorang individu yang paling cepat teridentifikasi dibanding pengenalan yang lain seperti sidik jari, tanda tangan, telinga maupun mata. Seiring dengan perkembangan teknologi, citra wajah dapat diolah dan diidentifikasi dengan mudah sehingga dapat digunakan pada sistem kehadiran. Salah satu algoritma yang digunakan untuk identifikasi citra wajah yaitu algoritma *eigenfaces* dengan pendekatan Analisis Komponen Utama atau biasa disebut dengan *PCA* (*Principle Component Analysis*). Algoritma *eigenfaces* adalah

himpunan eigenvektor yang digunakan untuk pengenalan wajah pada *computer vision* [1]. Algoritma ini merupakan basis dari pendekatan Analisis Komponen Utama [2][3]. Untuk meningkatkan nilai ketelitian identifikasi citra wajah, dibutuhkan algoritma deteksi wajah seperti algoritma *viola jones*.

Permasalahan yang ditimbulkan berdasarkan uraian tersebut adalah seberapa besar tingkat ketelitian identifikasi wajah jika proses identifikasi wajah dilakukan tanpa proses deteksi wajah. Selain itu, timbul juga permasalahan seperti seberapa banyak citra latih yang dibutuhkan agar ketelitian identifikasi citra wajah maksimal.

2 Dasar Teori

2.1. Matriks

Dalam bidang matematika, matriks merupakan kumpulan angka numerik yang disusun berdasarkan baris dan kolom. Matrik yang memiliki M baris dan N kolom biasa disebut matrik dengan ukuran MxN.

$$C_{M \times N} = \begin{bmatrix} c_{11} & \dots & c_{1N} \\ \vdots & \ddots & \vdots \\ c_{M1} & \dots & c_{MN} \end{bmatrix} \quad (1)$$

Matrik memiliki identitas yang disebut dengan *eigenvalue* dan *eigenvector*. *Eigenvalue* merupakan bilangan skalar dan *eigenvector* merupakan suatu matriks yang menjelaskan matriks tersebut. Namun, tidak semua matrik memiliki nilai *eigenvalue* dan *eigenvector*. Identitas tersebut yang akan digunakan untuk pengolahan gambar.

2.2. Citra

Citra ditinjau dari sudut pandang matematis merupakan suatu fungsi yang berkesinambungan dari intensitas cahaya pada suatu bidang. Citra dapat diolah atau dimodifikasi jika citra tersebut telah menjadi citra digital yang dapat disimpan pada perekam magnetik maupun elektrik. Karena pada dasarnya, citra digital merupakan sebuah matriks atau kumpulan dari beberapa matriks.

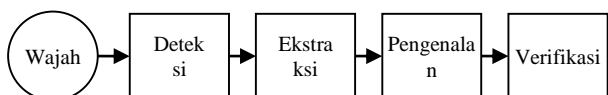
Berdasarkan format penyimpanan nilai warna, citra digital dibagi menjadi beberapa jenis seperti citra skala keabuan dan citra warna. Untuk skala 8 bit, citra skala keabuan dan warna memiliki nilai tiap piksel citra antara 0 sampai 255. Namun, citra warna terdiri atas tiga layer warna seperti citra RGB (*Red Green Blue*). Ketiga warna dengan nilai tertentu tersebut dicampur sehingga membentuk warna baru atau warna kombinasi [4][5]. Citra RGB dapat diubah kedalam bentuk citra skala keabuan dengan rumus [6]:

$$\text{Skala Keabuan} = 0.2989R + 0.587G + 0.114B \quad (2)$$

Dimana R = warna merah, G = warna hijau dan B = warna biru.

2.3. Pengenalan Wajah

Pada umumnya, pengenalan wajah (*Face Recognition*) dapat dicapai dengan beberapa tahap yaitu deteksi wajah, ekstraksi wajah dan pengenalan wajah. Deteksi wajah merupakan suatu teknik yang mampu menentukan letak dan ukuran wajah pada citra digital yang kemudian wilayah wajah yang telah ditentukan diproses lebih lanjut. Tahapan pengenalan wajah dapat dilihat pada Gambar 2.



Gambar 1. Tahapan Umum Pengenalan Wajah

Algoritma yang digunakan untuk mendeteksi wajah dibagi menjadi dua jenis yaitu berdasarkan fitur dan pembelajaran. Algoritma berdasarkan fitur dapat mendeteksi wajah berdasarkan beberapa fitur sederhana yang terdapat di daerah wajah. Fitur yang paling sering digunakan pada algoritma ini yaitu warna kulit pada wajah. Sedangkan algoritma pembelajaran didasarkan pada model statistik dan

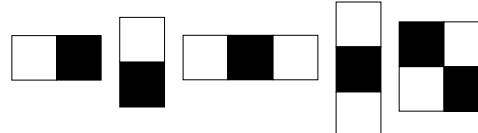
algoritma pembelajaran mesin. Untuk rotasi wajah yang berbeda dan di lingkungan yang memiliki cahaya yang redup, algoritma pembelajaran dapat mendeteksi wajah lebih baik dibanding algoritma berdasarkan fitur. [7]

Tahap selanjutnya merupakan ekstraksi wajah yaitu mengambil wilayah yang dianggap wajah pada bagian depan wajah pada citra digital untuk diproses pada pengenalan wajah. Tahap terakhir adalah tahap pengenalan wajah. Pada tahap ini, wajah seseorang dibandingkan dengan wajah yang telah ada sebelumnya. Berdasarkan [1] dan [8] terdapat berbagai pendekatan atau metode yang dapat dilakukan untuk pengenalan wajah. Salah satu pendekatan tersebut diantaranya adalah pendekatan *Principle Component Analysis (PCA)*. Tujuan utama pendekatan PCA adalah mencari nilai *eigenvector* dari tiap citra digital. Dikarenakan *eigenvector* merupakan suatu proyeksi dari citra tersebut.

2.4. Algoritma Viola Jones

Algoritma Viola – Jones merupakan algoritma berdasarkan pembelajaran yang digunakan untuk deteksi wajah pada citra digital. Selain digunakan untuk deteksi wajah, algoritma tersebut juga bisa digunakan untuk deteksi objek lain. Terdapat beberapa tahap Fitur Haar digunakan pada algoritma ini untuk identifikasi objek dan Citra Integral digunakan dalam proses perhitungan Fitur Haar agar perhitungan Fitur Haar lebih cepat dibanding tidak menggunakan Gambar Integral. Selanjutnya, dengan menggunakan algoritma *Adaboost (Adaptive Boost)* dapat diperoleh fitur terbaik dari citra digital.

Umumnya terdapat tiga jenis fitur Haar sederhana yang digunakan untuk deteksi wajah pada citra digital yaitu fitur dua segiempat, fitur tiga segiempat dan fitur empat segiempat. Fitur wajah akan ditemukan oleh salah satu fitur tersebut dengan cara menghitung fitur wajah secara horizontal dan vertikal piksel per piksel citra wajah. Fitur Haar-like dapat dilihat pada Gambar 3[7][9].



Gambar 2. Fitur Dasar Haar-Like

Fitur Haar-like dihitung dengan cara mengurangi nilai piksel pada wilayah segiempat yang berwarna hitam dengan nilai piksel pada wilayah segiempat yang berwarna putih. Proses perhitungan fitur Haar-like dapat dilihat pada rumus:

$$FH = \sum \text{Hitam(Piksel)} - \sum \text{Putih(Piksel)} \quad (3)$$

Dalam proses menghitung fitur wajah, akan menghabiskan banyak penyimpanan dan proses perhitungannya akan semakin lama jika ukuran piksel citra besar yaitu MxN. Oleh sebab itu, dibutuhkan suatu metode agar proses perhitungan lebih cepat dan efisien yaitu dengan mengubah citra wajah menjadi Citra Integral seperti contoh pada Gambar 4.

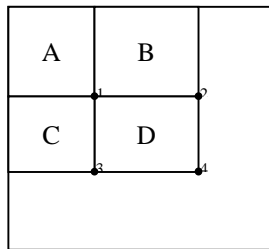
7	5	6
6	1	0
13	7	3

7	12	18
13	19	25
26	39	48

Gambar 3. Perubahan Citra Wajah Menjadi Citra Integral

Pada Gambar 4, sisi kiri merupakan citra wajah sedangkan sisi kanan merupakan citra wajah yang telah diolah dalam bentuk citra integral. Sebagai contoh, angka 19 pada citra integral merupakan hasil penjumlahan dari 7+5+6+1 citra wajah yang berada di sisi kiri. Begitu juga angka lainnya pada citra integral yang merupakan hasil penjumlahan citra wajah.

Setelah citra wajah diubah kedalam bentuk citra integral, maka proses perhitungan citra integral dapat dilihat pada Gambar 5.

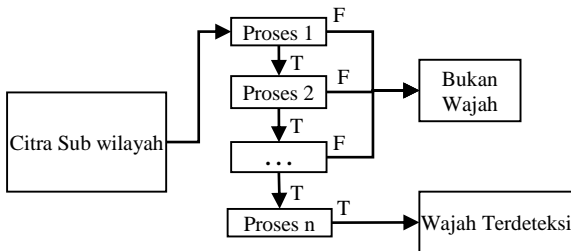


Gambar 4. Perhitungan Citra Integral

Berdasarkan Gambar tersebut, luas wilayah suatu citra dapat dihitung menggunakan dua operasi matematika dasar yaitu jumlah dan kurang. Sebagai contoh jika citra wajah yang telah ditemukan di wilayah D, maka luas wilayah D dapat dihitung menggunakan rumus:

$$\text{Luas D} = \text{titik 4} - \text{titik 3} - \text{titik 2} + \text{titik 1} \quad (4)$$

Proses deteksi wajah dilakukan berulang kali menggunakan basis ukuran sub wilayah citra dan fitur yang berbeda. Peralnya, setiap sub wilayah citra yang dianggap terdapat wajah akan diteruskan dari proses 1 ke proses 2. Namun, jika sub wilayah citra tersebut dianggap tidak terdapat wajah maka citra tersebut akan dieliminasi. Proses ini terus berlangsung sampai proses ke n. Semakin banyak proses yang dilalui sub wilayah citra, semakin akurat hasil deteksi wajah yang dilakukan. Akan tetapi waktu yang dibutuhkan juga akan semakin lama. Proses Cascade Classifier dapat dilihat pada Gambar 6.



Gambar 5. Proses Cascade Classifier

2.5. Pendekatan Analisis Komponen Utama

Pendekatan Analisa Komponen Utama merupakan pendekatan yang merepresentasikan atau merubah citra wajah ke dalam komponen *eigenface* yang merupakan himpunan dari *eigenvector* yang digunakan untuk pengenalan wajah pada *computer vision* [1] secara efektif dan efisien. Pada pendekatan ini, terdapat beberapa tahap yang perlu dilakukan yaitu [4][10][11][12][13][14]:

1. Beberapa citra yang akan dilatih dan yang akan diuji dipersiapkan. Misalnya 10 citra yang akan dilatih dan 5 citra yang akan diuji. Citra latih pada penelitian ini merupakan citra basis data. Sedangkan citra uji merupakan citra yang baru diambil sensor visual.
2. Beberapa citra wajah dengan ukuran matrik $M \times N$ piksel dirubah kedalam vektor citra (Γ) dengan ukuran matrik $K \times 1$, dimana K merupakan matriks MN . Sehingga didapatkan citra latih $\Gamma' = [\Gamma'_1 \Gamma'_2 \Gamma'_3 \dots \Gamma'_P]$ dan citra uji $\Gamma_b = [\Gamma_{b1} \Gamma_{b2} \Gamma_{b3} \dots \Gamma_{bQ}]$.
3. Kemudian, nilai tengah vektor citra wajah (Ψ) dapat dihitung dengan rumus

$$\Psi = \frac{1}{P} \sum_{i=1}^P \Gamma'_i \quad (5)$$

Dimana P merupakan banyaknya citra latih.

4. Selisih vektor citra (Φ) dapat dihitung dengan nilai citra latih (Γ_i) diselisihkan dengan nilai tengah vektor citra wajah (Ψ) menggunakan rumus sebagai berikut

$$\Phi_i = \Gamma_i - \Psi \quad (6)$$

Dimana Γ_i merupakan citra wajah yang dilatih. Sehingga didapatkan matriks yang berbeda $A = [\Phi_1 \Phi_2 \Phi_3 \dots \Phi_P]$

5. Hitung Matrik Kovarian (C)

$$C = AA^T \quad (7)$$

Dimana ukuran matriks C adalah $K \times K$ yang sangat besar sehingga matriks C diganti dengan Matrik L yang memiliki ukuran lebih kecil dengan pengurangan dimensi yaitu:

$$L = A^T A \quad (8)$$

Dengan Ukuran matriks L adalah $P \times P$

6. *Eigenvector* dan *eigenvalue* dapat dihitung dengan rumus sebagai berikut

$$LX_i = \lambda_i X_i \quad (9)$$

Dimana X_i adalah *eigenvector* dan λ_i adalah *eigenvalue*. Setelah didapatkan nilai *eigenvector*, *eigenface* dapat dihitung menggunakan rumus

$$U_i = \sum_{t=1}^P X_{it} \Phi_t \quad (10)$$

Dimana U_i merupakan *eigenface* yang akan dicari pada penelitian ini. Maka diperoleh $\Omega_i = U_1, U_2, U_3, \dots, U_Q$

- Tahapan selanjutnya adalah dilakukan perhitungan eigenfaces citra uji menggunakan rumus

$$U_b = X_i(\Gamma_b - \Psi) \tag{11}$$

Sehingga diperoleh $\Omega_\psi = U_{b1}, U_{b2}, U_{b3}, \dots, U_{bQ}$

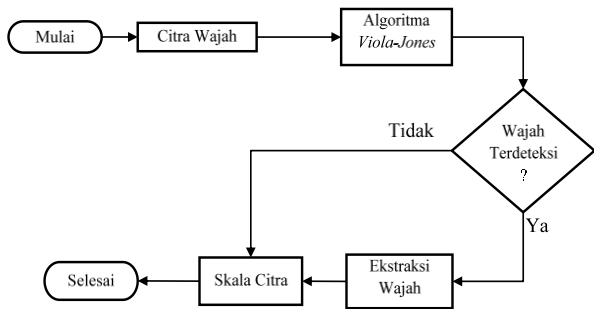
- Metode *Euclidean Distance* (δ_i) digunakan untuk menemukan jarak *eigenface* terpendek antara citra yang dilatih dengan citra yang diuji menggunakan rumus sebagai berikut

$$\delta_i = ||\Omega - \Omega_\psi|| = \sum_{i=1}^P (\Omega_i - \Omega_{\psi_i}) \tag{12}$$

Dari rumus tersebut diperoleh nilai *Euclidean Distance* antara citra latih dan citra uji. Sehingga program akan menentukan kecocokan citra wajah dengan cara memilih nilai *Euclidean Distance* yang terkecil dari semua nilai *Euclidean Distance* yang diperoleh berdasarkan banyaknya citra latih.

3 Perancangan Program Pengenal Wajah Menggunakan Algoritma *Eigenfaces*

Perangkat Lunak yang dirancang penulis yaitu perancangan program pengolahan citra wajah menggunakan algoritma *eigenfaces* pada pendekatan Analisis Komponen Utama dan algoritma *viola jones* sebagai deteksi wajah. Perancangan program tersebut dirancang di perangkat lunak Matlab. Diagram alir perancangan program deteksi wajah dapat dilihat pada Gambar 6.

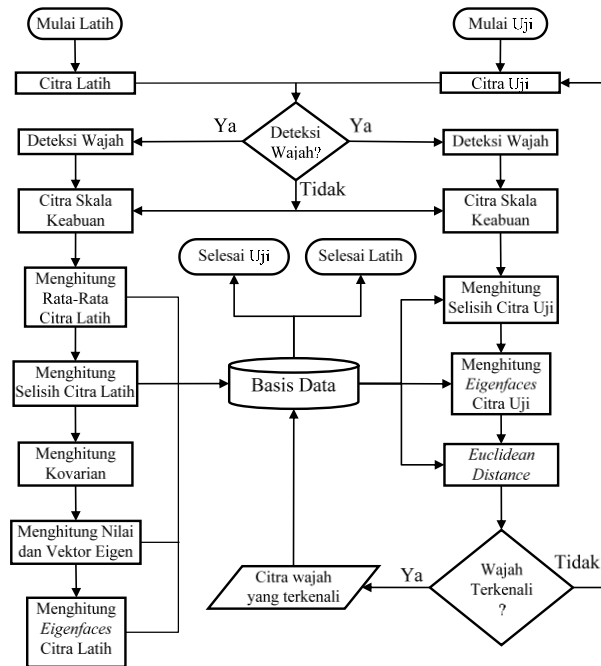


Gambar 6. Program Deteksi Wajah

Berdasarkan diagram tersebut, program dimulai dengan mendapatkan citra wajah. Citra tersebut diperoleh dari basis data face94 [8][15]. Total citra yang dilatih berjumlah 1500 dan total citra yang diuji berjumlah 500.

Citra wajah yang telah diperoleh tersebut, dideteksi dengan menggunakan algoritma *Viola-Jones*. Fungsi "*vision.CascadeObjectDetector();*" merupakan fungsi algoritma *Viola-Jones* pada perangkat lunak MATLAB yang dapat mendeteksi objek pada suatu citra. Namun pada penelitian ini, algoritma tersebut digunakan untuk mendeteksi wajah. Setelah wajah terdeteksi, dilakukan ekstraksi wajah. Ekstraksi wajah dilakukan agar wilayah yang bukan wajah dibuang sehingga dapat meningkatkan nilai ketelitian dalam proses identifikasi wajah pada tahap selanjutnya. Setelah dilakukan ekstraksi wajah, citra wajah

yang telah diekstraksi tersebut dilakukan penskalaan citra. Namun jika wajah tidak terdeteksi, maka citra wajah langsung ke proses penskalaan citra. Penskalaan citra dilakukan agar skala citra tersebut sesuai dengan skala citra pada citra yang terdapat pada basis data atau citra latih. Pada tahap ini program deteksi wajah berakhir dan dilanjutkan dengan program pengenalan wajah. Program Pengenal wajah dapat dilihat pada Gambar 7.



Gambar 7. Program Pengenal Wajah

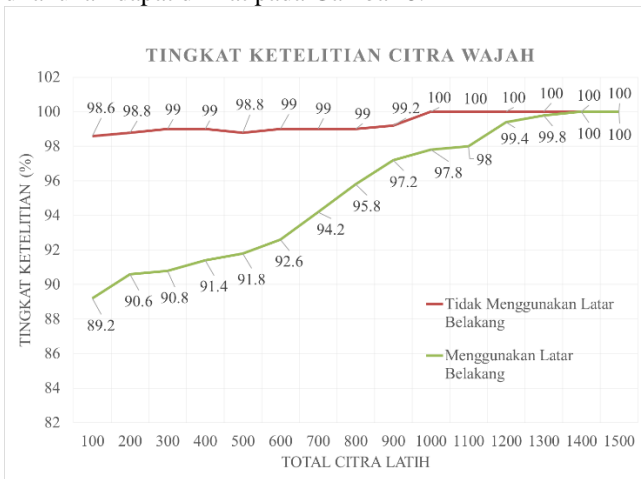
Berdasarkan diagram alir yang ditunjukkan oleh Gambar 7, proses identifikasi terbagi atas dua proses utama yaitu proses untuk citra latih dan proses untuk citra uji. Pada proses pelatihan untuk citra latih, program dimulai dengan mendapatkan citra latih. Citra tersebut dideteksi wilayah wajah menggunakan program deteksi wajah. Hasil deteksi wajah kemudian diubah ke dalam citra skala keabuan. Kemudian dihitung rata-rata citra latih tersebut dan kemudian hasil dari perhitungan tersebut disimpan di basis data serta diolah di tahap selanjutnya. Selisih citra latih dihitung setelah mendapatkan rata-rata citra latih dan hasil perhitungan disimpan di basis data. Matriks kovarian dihitung setelah diperoleh selisih citra latih. Nilai dan vektor *eigen* dihitung dan hasil perhitungan disimpan di basis data. Tahap terakhir merupakan perhitungan nilai *eigenfaces* dari citra uji dan hasil dari perhitungan tersebut disimpan ke dalam basis data. Proses tersebut dilakukan satu kali dan ketika terdapat perubahan pada citra latih seperti penambahan atau pengurangan citra latih.

Proses kedua dari program yang ditunjuk diagram alir pada gambar tersebut adalah proses citra uji. Proses dimulai setelah citra latih telah diproses dan citra uji telah diperoleh dari sensor visual. Citra uji tersebut kemudian dideteksi dengan program deteksi wajah. Setelah itu citra tersebut diubah kedalam citra skala keabuan. Selisih citra uji dihitung dengan cara citra uji dikurangi dengan rata-rata citra latih yang telah dibebankan dari basis data. *Eigenfaces* citra uji

dihitung setelah vektor eigen dibebankan dari basis data. Jarak *Euclidean* dihitung untuk mendapatkan kemiripan citra wajah. Kondisi citra uji akan teridentifikasi jika nilai dari jarak *euclidean* mendekati nilai nol. Jika wajah teridentifikasi maka ditampilkan citra wajah yang telah teridentifikasi pada citra uji beserta citra latih dan hasil identifikasi disimpan ke dalam basis data. Jika wajah tidak teridentifikasi maka pengguna ditampilkan sebuah pesan bahwa citra wajah tersebut tidak dapat diidentifikasi dan proses akan diulang dari proses citra uji.

4 Temuan Penelitian

Berdasarkan penelitian yang telah dilakukan, terdapat pengaruh penggunaan latar belakang pada citra wajah terhadap tingkat ketelitian identifikasi citra wajah. Ketelitian identifikasi wajah secara keseluruhan pengujian yang telah dilakukan dapat dilihat pada Gambar 8.

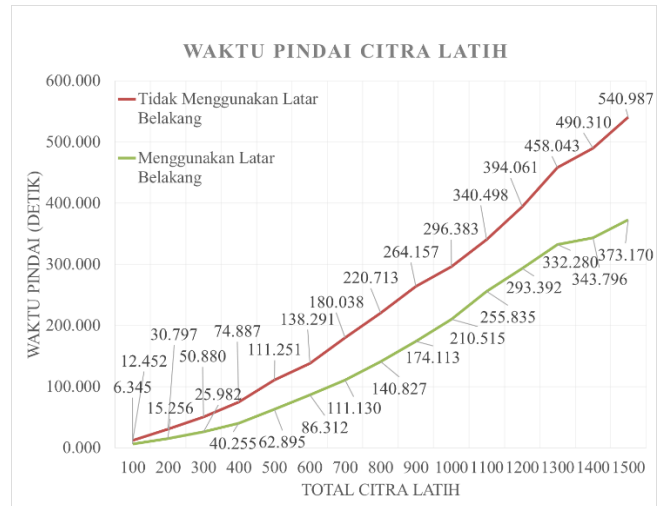


Gambar 8. Tingkat Ketelitian identifikasi Citra Wajah

Grafik tersebut menunjukkan bahwa tingkat ketelitian identifikasi citra wajah meningkat hingga mencapai 98.6% ketika latar belakang tidak digunakan pada citra wajah. Namun ketika latar belakang digunakan pada citra wajah, tingkat ketelitian wajah bernilai 89.2%. Hal tersebut terjadi pada saat total citra latih berjumlah 100 buah.

Selain itu, semakin banyak citra wajah yang dilatih pengaruh penggunaan latar belakang pada citra wajah tidak berdampak besar terhadap tingkat ketelitian identifikasi citra wajah. Akan tetapi, semakin banyak citra wajah yang dilatih maka semakin besar tingkat ketelitian identifikasi wajah yaitu mencapai 100%.

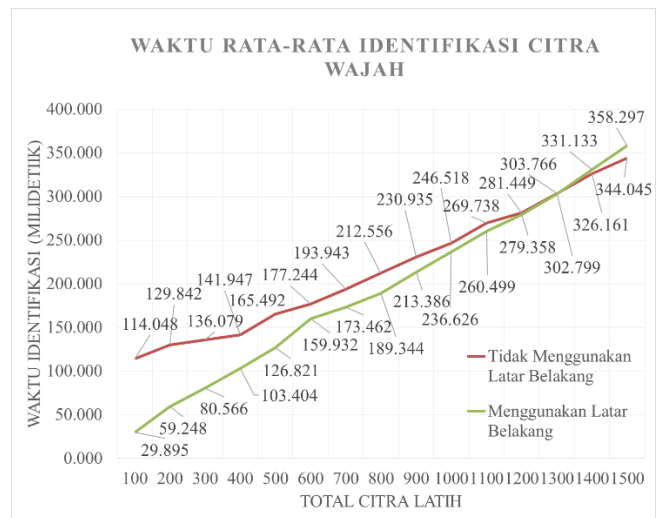
Waktu yang dibutuhkan untuk memindai citra latih dapat dilihat pada Gambar 9.



Gambar 9. Waktu Pindai Citra Latih

Berdasarkan grafik tersebut, semakin banyak citra latih yang digunakan, maka semakin lama waktu yang digunakan untuk memindai citra latih tersebut. Namun, memindai citra latih hanya dilakukan satu kali dan jika terdapat perubahan atau penambahan citra latih pada basis data. Sedangkan waktu yang dibutuhkan untuk melakukan identifikasi citra wajah menggunakan algoritma *eigenfaces* dapat dilakukan berulang kali.

Waktu identifikasi citra wajah tersebut dapat dilihat pada Gambar 10.



Gambar 10. Waktu Rata-Rata Identifikasi Citra Latih

Grafik pada Gambar 10 menunjukkan bahwa waktu rata-rata yang dibutuhkan untuk melakukan identifikasi citra wajah ketika latar belakang citra tidak digunakan tidak jauh berbeda dengan kondisi citra menggunakan latar belakang. Namun, semakin banyak citra yang dilatih maka semakin besar waktu yang dibutuhkan untuk melakukan identifikasi citra wajah.

5 Simpulan

Berdasarkan temuan yang telah diperoleh, maka ketelitian identifikasi wajah meningkat menjadi 98.6% dibanding tidak menggunakan algoritma viola jones yaitu 89.2% ketika total citra latih sebanyak 100 individu. Selain itu, semakin banyak citra wajah yang dilatih maka semakin besar tingkat ketelitian identifikasi wajah yang diperoleh. Namun, waktu yang diperlukan juga semakin lama.

Kepustakaan

- [1] K. S. Kumar, S. Prasad, B. S. Vijay, dan R. C. Tripathi, "Real Time Face Recognition Using ada Boost Improved Fast PCA algorithm," *Int. J. Artif. Intell. Appl.*, vol. 2, no. 3, hal. 45–58, 2011.
- [2] A. A. Shah, Z. A. Zaidi, B. S. Chowdhry, dan J. Daudpoto, "Real time face detection/monitor using raspberry pi and MATLAB," *Appl. Inf. Commun. Technol. AICT 2016 - Conf. Proc.*, hal. 1–4, 2016.
- [3] A. Zein, "Pendeteksian Multi Wajah Dan Recognition Secara Real Time Menggunakan Metoda Principal Component Analysis (Pca) Dan," *J. Teknol. Inf. ESIT*, vol. 12, no. 01, hal. 1–7, 2018.
- [4] D. Siswo Wardoyo, "Sistem Presensi Berbasis Algoritma Eigenface Dengan Metode Principal Component Analysis," *Setrum (Sistem Kendali Tenaga Elektron. Telekomun. Komputer)*, vol. 3, no. 1, hal. 61–68, 2014.
- [5] I. W. Angga, W. Kusuma, dan R. L. Ellyana, "Penerapan Citra Terkompresi Pada Segmentasi Citra Menggunakan Algoritme K-MEANS," *JUTEI (Jurnal Terap. Teknol. Informasi)*, vol. 2, hal. 65–74, 2018.
- [6] H. Santoso dan A. Harjoko, "Haar Cascade Classifier dan Algoritma Adaboost Untuk Deteksi Banyak Wajah Dalam Ruang Kelas," *J. Teknol.*, vol. 6, no. 2, hal. 108–115, 2013.
- [7] M. N. Chaudhari, "Face Detection using Viola Jones Algorithm and Neural Networks," *2018 Fourth Int. Conf. Comput. Commun. Control Autom.*, hal. 1–6, 2018.
- [8] T. F. Karim, M. S. H. Lipu, M. L. Rahman, dan F. Sultana, "Face recognition using PCA-based method," *ICAMS 2010 - Proc. 2010 IEEE Int. Conf. Adv. Manag. Sci.*, vol. 3, hal. 158–162, 2010.
- [9] K. Vikram, "FACIAL PARTS DETECTION USING VIOLA," *Int. Conf. Adv. Comput. Commun. Syst. (ICACCS -2015)*, hal. 1–4, 2017.
- [10] M. R. Muliawan, B. Irawan, dan Y. Brianorman, "Implementasi Pengenalan Wajah Dengan Metode Eigenface Pada Sistem Absensi," *J. Coding, Sist. Komput. Untan*, vol. 03, no. 1, hal. 41–50, 2015.
- [11] M. Abdullah, M. Wazzan, dan S. Bo-saeed, "Optimizing Face Recognition Using PCA," *Int. J. Artif. Intell. Appl.*, vol. 3, no. 2, hal. 23–31, 2012.
- [12] L. C. Paul dan A. Al Sumam, "Face Recognition using Principal Component Analysis, OpenCV and EmguCV," *Int. J. Adv. Res. Comput. Eng. Technol.*, vol. 1, no. 9, hal. 135–139, 2012.
- [13] H. H. Lwin, A. S. Khaing, dan H. M. Tun, "Automatic Door Access System Using Face Recognition," *Int. J. Sci. Technol. Res.*, vol. 4, no. 06, hal. 294–299, 2015.
- [14] H. Kurniawan dan T. Hidayat, "Perancangan Program Pengenalan Wajah Menggunakan Fungsi Jarak Metode Euclidean Pada Matlab," *Semin. Nas. Apl. Teknol. Inf. 2008 (SNATI 2008) Yogyakarta, 21 Juni 2008*, vol. Vol 1, no. Pengenalan Wajah Menggunakan Fungsi Jarak Metode Euclidean Pada Matlab, hal. 15–18, 2008.
- [15] "Face Recognition Data." [Daring]. Tersedia pada: <https://cswww.essex.ac.uk/mv/allfaces/faces94.html>. [Diakses: 20-Feb-2019].