BAB II
TINJAUAN PUSTAKA

2.1 Pengolahan Citra

Citra merupakan suatu fungsi kontinu dari intensitas cahaya dalam bidang dua dimensi, di mana koordinat (x,y) menunjukkan citra dan nilai f menunjukkan tingkat kecerahan atau derajat keabuan. Array digital dua dimensi memiliki nilai $f(x,y)$ diskrit pada koordinat citra dan kecerahan. Pemrosesan citra digital dua dimensi, atau pemrosesan gambar, adalah proses pengolahan dan analisis citra yang banyak bergantung pada pemahaman visual. Ini mengolah data masukan dan menghasilkan informasi dalam bentuk citra[5]. Pengolahan citra digital biasanya dilakukan dengan tujuan meningkatkan kualitas gambar sehingga dapat diinterpretasikan dengan mudah oleh mata manusia dan mengolah informasi dalam gambar untuk pengenalan objek secara otomatis.

Citra harus direpresentasikan secara numerik dengan nilai diskrit agar dapat diolah komputer. Matrik ukuran $M \times N$ adalah cara untuk menyatakan citra digital. Masing-masing komponen dikenal sebagai *pixel (picture element)*, berikut ini adalah elemen piksel citra[5].

$$f(x,y) = \begin{bmatrix} f(0,0) & f(0,1)\cdots & f(0,N) \\ f(1,0) & f(1,1)\cdots & f(1,N) \\ \vdots & \vdots & \vdots \\ f(M-1,0) & f(M-1,1) & f(M-1,N-1) \end{bmatrix} \tag{2.1}$$

2.2 Penelitian terkait Metode dalam Pengenalan Wajah

Secara umum pengenalan wajah salah satunya, yaitu sistem *feature based* dan *image-based.* Pada sistem pertama sistem ini mengekstrak fitur - fitur utama dari wajah seperti mata, hidung, dan mulut. Setelah fitur utama selesai di ekstrak, maka fitur tersebut dihubungkan dengan model geometris untuk pengenalan. Sedangkan sistem kedua menggunakan seluruh gambar wajah untuk pengenalan tanpa fokus pada fitur-fitur individual[6]. Wajah merupakan bagian tubuh yang unik dan mudah dikenali sehingga ideal untuk verifikasi dan autentikasi.

Ada beberapa penelitian sebelumnya perihal pengenalan wajah yaitu, ”Klasifikasi wajah menggunakan *Support Vector Machine* (SVM)”[7], pada penelitian ini

dibangun dengan menggunakan metode SVM dan bahasa pemrograman Matlab dari jumlah sampel 200 subjek yang digunakan sehingga menghasilkan tingkat *true detection* 90% dan *false* 10%. Penelitian Pada Judul *"Face Recognition based Feature Extraction using Principal Component Analysis* (PCA)"[8], dimana penelitiannya melakukan konstruksi pola wajah yang tersimpan pada database citra digital, dengan digunakannya metode PCA yaitu untuk menyederhanakan fitur dan karakteristik wajah sehingga diperoleh proporsi yang mampu mewakili karakteristik wajah asli. Sementara, penelitian dengan judul "Sistem Pengenalan Wajah Menggunakan Metode Eigenface Berbasis Raspberry Pi"[9], dimana hasil pengujian pengenalan wajah sistem menunjukkan bahwa sistem tidak dapat mengidentifikasi semua posisi wajah yang diuji dengan urutan posisi depan, kanan, kiri, atas, dan bawah. Dengan demikian, ada posisi wajah yang belum terdeteksi dengan sempurna yang tidak dapat ditampilkan oleh pengenalan wajah dan pelatihan database. Wajah yang posisi ke depan selalu dikenal dengan baik.

Saat ini telah banyak dikembangkan sistem yang melindungi akses ke rumah atau ruangan dengan menggunakan berbagai metode verifikasi identitas melalui sistem komputer, seperti kunci, kartu, password, dan sebagainya[10]. Pengenalan wajah digunakan untuk mendeteksi dan mengenali orang saat lensa kamera mengambil citra mereka[11].

2.3 *Principal Component Analysis* (PCA)

Principal component analysis (PCA) merupakan teknik yang bermanfaat untuk mengekstrak fitur penting dari citra wajah. Sederhananya, PCA bekerja dengan melakukan transformasi linear pada dataset citra wajah. Transformasi ini menghasilkan sistem koordinat baru yang lebih efektif dalam mewakili data[12]. Teknik PCA mampu mereduksi dimensi data citra wajah tanpa kehilangan informasi penting. Hal ini memungkinkan analisis data yang lebih efisien dan efektif.

Bayangkan sebuah citra wajah $I(x,y)$ dengan dimensi $N \times N$. Citra ini dapat direpresentasikan sebagai array $N \times N$ nilai identitas dua dimensi, atau sebagai vektor dengan dimensi N^2. Sebagai contoh, gambar berukuran 256 x 256 akan menggambarkan sebagai vektor dengan dimensi 65.536, yang setara dengan sebuah titik dalam ruang 65.536 dimensi. Fungsi utama dari *Principal Component Analysis* (PCA) dalam konteks citra wajah adalah menemukan nilai vektor yang paling optimal untuk menggambarkan distribusi data gambar wajah dalam seluruh ruang dimensi gambar. Berikut ini adalah langkah-langkah untuk mengekstraksi fitur PCA:

1. Terdapat satu set gambar M dengan berukuran $N \times N$ dapat digambarkan dengan $I_1, I_2, I_3, ...I_M$ yang membentuk vektor berukuran N^2. Citra wajah direpresentasikan sebagai $\Gamma_i = \{\Gamma_1, \Gamma_2, ..., \Gamma_M\}$, dimana setiap Γ_i adalah vektor dari dimensi N dan jumlah citra wajah dalam dataset pelatihan dilambangkan dengan M.

2. Representasikan setiap citra I_i menjadi sebuah vektor Γ_i sebagai berikut:

$$\begin{bmatrix} a_{1_1} & a_{1_2} & \cdots & a_{1_N} \\ a_{2_1} & a_{2_2} & \cdots & a_{2_N} \\ \vdots & \vdots & \ddots & \vdots \\ a_{N_1} & a_{2_2} & \cdots & a_{2_N} \end{bmatrix} \xrightarrow{\text{concatenate}} \begin{bmatrix} a_{1_1} \\ \vdots \\ a_{1_N} \\ \vdots \\ a_{2_N} \\ \vdots \\ a_{N_N} \end{bmatrix} = \Gamma_i \tag{2.2}$$

3. Rata-rata set pelatihan ditentukan oleh

$$\Psi = \frac{1}{M} \sum_{i=1}^{M} \Gamma_i \tag{2.3}$$

4. Mencari rata-rata dari beberapa wajah yang berbeda dengan vektor $\{\Phi_1, \Phi_2, \Phi_3, \ldots, \Phi_M\}$

$$\Phi_i = \Gamma_i - \Psi \tag{2.4}$$

5. Mencari matriks kovarian dengan persamaan sebagai berikut:

$$C = A \times A^T, \quad \text{dimana} \quad A = [\Phi_1, \Phi_2, \Phi_3, \ldots, \Phi_M] \tag{2.5}$$

   Di dalam kasus ini, kovarian matrik C menggambarkan hubungan antara 2 matrik dan varian yang terdapat di dalam dataset. Sedangkan A^T adalah transpose dari matriks A, dimana matriks A Menyimpan perbedaan antara setiap citra latih dan citra wajah rata-rata *(average face)*. Keistimewaan matriks kovarian ini terletak pada kemampuannya untuk menghasilkan vektor eigen dan nilai eigen yang kita butuhkan.

6. Matriks $w \times h$ dari citra wajah, yang dianggap sebagai vektor dengan dimensi N, menghasilkan matriks C berdimensi N^2. Perhitungan matriks kovarian

yang besar dapat dihindari dengan menggunakan persamaan berikut:

$$C \times u_i = \lambda_i \times u_i \tag{2.6}$$

Vektor eigen u_i dan nilai eigen λ_i merupakan nilai eigen dari citra wajah yang bersangkutan.

7. Kita dapat mencari vektor eigen dan nilai eigen dari matrik kovarian (C). misalkan $L = A^T \text{ X } A$. L adalah matrik dimensi $M \times M$, dimana M lebih kecil dari N. Hal ini membuat perhitungan vektor eigen dan nilai eigen menjadi lebih mudah dari matrik tersebut dengan menyelesaikan persamaan berikut:

$$\begin{gathered} L \times V_i = d_i \times V_i \\ A^T \times A \times V_i = d_i \times V_i \\ A \times A^T \times A \times V_i = d_i \times A \times V_i \\ C \times A \times V_i = d_i \times A \times V_i \end{gathered}$$

Dari persamaan di atas, dapat disimpulkan bahwa $u_i = A \times V_i$ dan $\lambda_i = d_i$ merupakan vektor eigen dan nilai eigen dari matriks kovarian. Dengan menggunakan matrik L, kita dapat memperoleh vektor eigen dan nilai eigen sebanyak M.

8. Langkah selanjutnya adalah memilih K eigen vektor terbalik secara heuristik. setelah itu, sebelum melakukan perhitungan bobot, Φ_i direpresentasikan dengan kombinasi linear dari eigenvektor U_i sebagai berikut:

$$\Phi_i = \sum_{J=1}^{K} W_j u_j, \quad \text{dimana} \quad u_i \quad \text{adalah eigenface.} \tag{2.7}$$

Sehingga nilai bobot dapat dihitung dengan rumus:

$$W_j = U_j^T \Phi_i \tag{2.8}$$

Langkah selanjutnya citra training yang sudah dinormalisasi, diubah menjadi sebuah vektor berikut:

$$\Omega = \begin{bmatrix} W_1 \\ W_2 \\ W_3 \\ \vdots \\ W_K \end{bmatrix} \tag{2.9}$$

Dimana i = 1,2,3 ..., M. Hasil Akhir, vektor yang telah diperhitungkan sesuai dengan setiap gambar dalam *training set* akan disimpan sebagai template. Setelah bobot vektor telah diketahui, langkah berikutnya adalah tahap pengenalan dengan menerapkan proses klasifikasi.

2.4 Klasifikasi

Klasifikasi adalah proses pemisahan hasil ekstraksi ciri untuk membedakan dua kelas yaitu non-wajah dan wajah[13]. Terdapat tiga jenis utama klasifikator, yaitu:

1. *Classifier* berdasarkan keputusan Bayes
   Classifier ini mencoba melakukan klasifikasi kelas M ($w1, w2, ...M$) dan pola yang tidak diketahui yang dipresentasikan dengan vektor x dengan membentuk probabilitas kondisional sebanyak M ($P(w_i, x)$, $i = 1, 2, ...M$). Nilai-nilai ini menunjukkan kemungkinan bahwa pola yang tidak diketahui termasuk dalam kelas w_i tertentu. *Classifier* mencari nilai yang paling mungkin untuk melakukan klasifikasi.

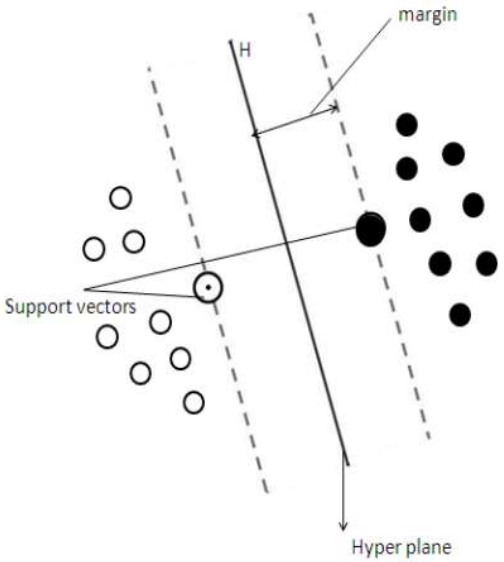
2. *Linear Classifier*
   Classifier ini melakukan klasifikasi dengan menggunakan persamaan linear sederhana. Keunggulan metode klasifikasi ini adalah kemudahan penggunaan dan sederhananya.

3. *Nonlinear Classifier*
   Classifier ini adalah pengembangan dari *linear classifier*. Dalam kasus di mana kelas tidak dapat dipisahkan secara *linear*, *classifier* ini dirancang secara optimal dengan tujuan untuk meminimalkan kuadrat error yang dihasilkan.

2.5 *Support Vector Machine*(SVM)

Support Vector Machine (SVM) adalah metode klasifikasi yang pertama kali diperkenalkan oleh Vapnik pada tahun 1998. Pada dasarnya, klasifikasi ini bekerja dengan menentukan batas antara dua kelas menggunakan data yang paling dekat. Algoritma ini menggunakan ruang fitur dimensi yang tinggi. Hal ini membuat menghitung produk skalar dari ruang fitur yang ada lebih sulit. Untuk menghitung produk skalar ini yaitu fungsi kernel. Dengan fungsi kernel dapat menghitung produk skalar tanpa perlu melakukan perhitungan ruang fitur secara eksplisit[14].

Pada awalnya diciptakan sebagai pengklasifikasi linear, metode SVM juga dapat digunakan untuk pemetaan data non linier, seperti yang ditunjukkan cara pemisahan data dengan metode SVM pada contoh disederhanakan pada Gambar 2.1.



Gambar 2.1. Pemisahan data dengan Metode SVM

SVM memisahkan data p-dimensi menggunakan bidang berdimensi p-1, juga dikenal sebagai *hyperplane*. Ini memaksimalkan margin pengumpulan data, yang diukur sebagai jarak minimal sampel ke *hyperplane*. Dengan menggunakan SVM, proses pembelajaran bertujuan untuk menghasilkan hipotesis sebagai bidang pemisah terbaik yang meminimalkan risiko empiris, yaitu rata-rata kesalahan pada data pelatihan, dan memiliki generalisasi yang baik. Generalisasi adalah kemampuan sebuah hipotesis untuk mengklasifikasikan data yang tidak ada dalam data pelatihan dengan benar. Untuk menjamin generalisasi ini, SVM bekerja berdasarkan prinsip *Structural Risk Minimization* (SRM). Di bawah ini adalah algoritma SVM:

1. Fungsi kernel *(linear, polynomial, radial basis function,* dan *sigmoid)* digunakan untuk memetakan atribut data ke dalam ruang dimensi yang lebih tinggi.

2. Memetakan data yang telah dipisahkan ke dalam ruang dimensi tinggi untuk mencari *hyperplane* terbaik.

3. Memaksimalkan jarak *hyperplane* dengan titik data terdekat *(Support Vector)* adalah cara untuk mendapatkan *hyperplane* terbaik.

2.6 Perhitungan Akurasi

Akurasi adalah ukuran yang digunakan untuk menilai seberapa baik suatu sistem atau model dalam melakukan prediksi atau klasifikasi dengan benar. Dalam pengenalan wajah, akurasi mengacu pada persentase dari total wajah yang diuji

yang dapat dikenali dengan benar oleh sistem. Dengan kata lain, akurasi menunjukkan tingkat ketepatan atau keandalan suatu model dalam mengidentifikasi wajah yang sesuai dengan data referensi yang dimilikinya. Nilai akurasi pengenalan wajah diberikan dalam bentuk persen.

Berikut ini adalah rumus perhitungan hasil dari pengujian pengenalan wajah sebagai berikut:

$$\text{Akurasi} = \left(\frac{\text{Jumlah Wajah yang Dikenali}}{\text{Total Wajah yang Diuji}} \right) \times 100\% \tag{2.10}$$

Pada bagian ini adalah rumus untuk menghitung rata-rata waktu komputasi yang diperlukan untuk memproses setiap gambar sebagai berikut:

$$\text{Rata-rata Waktu Pengenalan} = \frac{\text{Jumlah Total Waktu Pengenalan}}{\text{Jumlah Pengujian}} \tag{2.11}$$

2.7 Raspberry Pi 3 Model B+

Raspberry Pi 3 Model B+ yaitu Raspberry Pi yang menggunakan prosesor Broadcom BCM2837, merupakan 64-bit yang lebih cepat dari BCM2836. Raspberry Pi 3 adalah perangkat pertama yang memiliki konektivitas nirkabel di dalamnya. Bisa terhubung ke jaringan WiFi dan perangkat Bluetooth. Raspberry Pi 3 memiliki beberapa port yang berbeda dari Raspberry Pi 2. Raspberry Pi 3 memiliki 4 port USB, sebuah port jaringan LAN 10/100, dan 40 pin GPIO dan semua port lain seperti model-model sebelumnya[15]. Pada Tabel 2.1 merupakan Spesifikasi Raspberry Pi 3 Model B+ yang digunakan.

Tabel 2.1. Spesifikasi Raspberry Pi 3 Model B+

Spesifikasi Raspberry Pi 3 Model B+		
No	**Item**	**Spesifikasi**
1	SoC	BCM2837 64bit
2	CPU	Quad Cortex A53 @ 1.2GHz
3	Instruction	ARMv8-A
4	GPU	400MHz VideoCore IV
5	RAM	1GB SDRAM
6	Storage	micro-SD
7	Micro SD	32 GB
8	Ethernet	10/100
9	Wireless	802.11n/Bluetooth 4.0
10	Video Output	HDMI/Composite
11	Audio Output	HDMI/Headphone
12	GPIO	40

2.8 Python

Python adalah sebuah bahasa pemrograman yang tersedia sebagai sumber daya *open source*. Guido Van Rossum pertama kali menggunakan Python di Centrum Wiskunde & Informatica (CWI) di Belanda pada awal tahun 1990-an. Python dapat berjalan di banyak sistem operasi. Keunggulan tambahan dari pemrograman Python adalah bentuk programnya yang sederhana, yang membuatnya mudah dipelajari[16].

Sistem ini menggunakan bahasa pemrograman Python dan juga menggunakan *library* OpenCV. OpenCV adalah *library open source* yang dikembangkan oleh Intel yang bertujuan untuk menyederhanakan pemrograman yang berkaitan dengan pengolahan citra digital. Beberapa fitur OpenCV termasuk pengenalan wajah, deteksi wajah, dan berbagai metode kecerdasan buatan. OpenCV bekerja dengan berbagai bahasa pemrograman, seperti C, C++, java, dan Python, serta berbagai platform sistem operasi, seperti Windows, Linux, Mac OS, IOS, dan Android[17].

2.9 Data Wajah Mandiri

Data citra wajah yang diambil secara mandiri, setiap set terdiri dari sepuluh data wajah unik dari empat belas subjek yang berbeda. Setiap foto diambil dengan subjek tegak dan ekspresi yang berbeda, latar belakang, pencahayaan, dan warna yang berbeda. Untuk mendukung penelitian tentang pengenalan wajah, dataset ini dibuat sebagai representasi dari wajah orang Indonesia. Setiap gambar dalam file tersebut berukuran 640×480 piksel dan berformat JPG. Pada Gambar 2.2 merupakan con-

toh dari dataset mandiri yang digunakan pada penelitian ini, citra dataset mandiri dapat dilihat pada Lampiran 1.

Gambar 2.2. Data Mandiri