

BAB II LANDASAN TEORI

2.1 Penelitian Terdahulu

Dalam penelitian ini didasari oleh penelitian-penelitian sebelumnya yang kemudian dikembangkan agar menjadi suatu sistem yang lebih baik berdasarkan dari hasil dan saran yang sebelumnya pernah diteliti. Penelitian tersebut antara lain adalah sebagai berikut:

Tabel 2.1 Penelitian Terdahulu

No.	Judul dan Nama Peneliti	Permasalahan	Metode Penyelesaian	Hasil dan Akurasi Penelitian
1.	Identifikasi Wajah dengan Metode Eigenface untuk Sistem Presensi Pegawai Desa Berbasis Android Nama: I Putrayana Wardana dan Ida Bagus Kade Dwi Suta Negara	Presensi pegawai sulit dan lama pimpinan untuk memperoleh data dan presensi dicatat tidak sesuai dengan data yang sebenarnya karena pegawai sering melakukan presensi tidak tepat pada waktunya (Wardana & Negara, 2021).	Metode penyelesaiannya adalah <i>eigenface</i> untuk mendapatkan hasil yang <i>valid</i> . Untuk menyelesaikan masalah indentifikasi tersebut menggunakan algoritma <i>eigenface</i> yang cukup cepat dalam melakukan pendeteksian.	akurasi hasil akurasi dengan nilai 85% dengan jumlah <i>True Positive</i> sejumlah 8 data, <i>True Negative</i> sejumlah 9 data, <i>False Positive</i> sejumlah 2 data, dan <i>False Negatif</i> sejumlah 1 data. Didalam aplikasi tersebut memiliki fitur menu training wajah, absensi menggunakan pengenalan

Tabel 2.1 Penelitian Terdahulu lanjutan

No.	Judul dan Nama Peneliti	Permasalahan	Metode Penyelesaian	Hasil dan Akurasi Penelitian
				wajah dan menu cetak laporan absensi.
2.	Sistem Deteksi Wajah untuk Identifikasi Kehadiran Mahasiswa dengan Menggunakan Metode Eigenface PCA Nama : Husni Sulaiman, Zahir Zainuddin, dan Supriadi Sahibu	Mahasiswa yang tercatat hadir walau sebenarnya tidak mengikuti perkuliahan tersebut karena ada mahasiswa yang menandatangani-nya (Sulaiman dkk., 2019)	Penelitian ini menggunakan metode eigenface PCA yang kemudian file perhitungan akan disimpan dalam format *xml. Selanjutnya dari perhitungan tersebut system akan mengidentifikasi dan memverifikasi wajah untuk dimasukkan kedalam database.	Hasil dari riset tersebut proses absensi dapat dilakukan <i>real-time</i> melalui metode <i>eigenface</i> dengan total tingkat akurasi adalah 86,6% jika cahaya disekitar cukup dan juga sistem tersebut dapat langsung mengidentifikasi lebih dari satu wajah
3.	Penerapan Face Recognition Berbasis GUI	Sistem presensi masih manual yang	Metode eigenface digunakan	Pada sistem yang dikembangkan

Tabel 2.1 Penelitian Terdahulu lanjutan

No.	Judul dan Nama Peneliti	Permasalahan	Metode Penyelesaian	Hasil dan Akurasi Penelitian
	Visual Studio 2012 Menggunakan Algoritma Eigenface dan Metode Pengembangan Waterfall Pada Sistem Absensi Mahasiswa IT Telkom Purwokerto Nama : Ilham Fauzi , Apri Junaidi , dan Wahyu Andi Saputra	mempengaruhi kelancaran dalam kegiatan perkuliahan mahasiswa sebanding dengan meningkatnya mahasiswa IT Telkom yang meningkat setiap tahunnya (Fauzi dkk., 2022).	sebagai solusi presensi mahasiswa memanfaatkan sistem pengenalan wajah.	menggunakan teknologi <i>face recognition</i> dapat berjalan dengan baik di webcam dan algoritma <i>eigenface</i> dapat berjalan dengan baik sehingga dapat mendeteksi wajah mahasiswa.
4.	Absensi Pengenalan Wajah Menggunakan Algoritma Eigenface Berbasis Web Nama : Satria Putra, Iskandar	Sistem absensi barcode memiliki permasalahan ketika karyawan tidak membawa kartu yang diinput barcode, maka karyawan tidak akan bisa melakukan	Menggunakan metode eigenface sistem presensi berbasis web sehingga karyawan dapat mengakses aplikasi melalui perangkat milik masing-masing	Untuk melakukan proses identifikasi minimal hasil pemindaian memiliki akurasi diatas 60% agar proses

Tabel 2.1 Penelitian Terdahulu lanjutan

No.	Judul dan Nama Peneliti	Permasalahan	Metode Penyelesaian	Hasil dan Akurasi Penelitian
	Fitri dan Sari Nigsih	absensi. Penggunaan sistem absensi sidik jari pada masa pandemic covid-19 tidak bisa dilakukan. Kelemahan alat sidik jari saat alat yang digunakan secara bersamaan oleh seluruh karyawan memungkinkan virus covid-19 menyebar melalui alat tersebut (Putra dkk., 2021)	agar tidak terjadi kontak langsung	absensi dapat dilakukan.
5.	Identifikasi Citra Wajah Menggunakan Algoritma Eigenface	Seberapa akurat identifikasi wajah jika tidak menggunakan identifikasi wajah dilakukan tanpa	Metode yang digunakan adalah eigenface dengan algoritma viola jones dengan melatih 1500	Berdasarkan hasil, akurasi perbaikan wajah meningkat menjadi 98,6% saat menggunakan

Tabel 2.1 Penelitian Terdahulu lanjutan

No.	Judul dan Nama Peneliti	Permasalahan	Metode Penyelesaian	Hasil dan Akurasi Penelitian
	Nama : Andre Mochammad Satrio, Mohammad Mujirudin, Harry Ramza	pemindaian wajah dan seberapa banyak citra latih yang dilakukan agar tingkat akurasi citra wajah maksimal (Satrio dkk., 2019)	wajah serta menguji 500 wajah Setelah pelatihan dilakukan perhitungan menggunakan nilai eigenface yang kemudian disimpan kedalam database.	algoritma viola jones, yang merupakan 89,2% dari seluruh pelatihan gambar hingga 100 orang. Selain itu, semakin banyak gambar wajah yang dibatasi, semakin diperlukan presisi wajah. Namun, itu juga membutuhkan lebih banyak waktu.

2.2 Landasan Teori

2.2.1 Metode Eigenface

Kata *eigenface* merupakan bahasa Jerman "*eigenwert*" dimana "*eigen*" berarti karakteristik dan "*wert*" berarti nilai. Dari situlah nama "*eigenface*" sebenarnya berasal. Menurut (Fatta, 2006) *eigenface* merupakan salah satu metode pengenalan pola wajah berdasarkan *Principle Component Analysis* (PCA). Ide dasar di balik pengenalan wajah adalah sebagai berikut: Informasi wajah yang berbeda selanjutnya *diencode* dan dikontraskan dengan temuan

dari *decoding* sebelumnya dilakukan. Saat menggunakan metode *eigenface*, *decoding* dilakukan dengan melakukan penghitungan *eigenvector* dan kemudian dinyatakan dalam matriks besar. Teknik ini disebut sebagai *eigenface* karena *eigenvector* juga dapat digambarkan sebagai ciri-ciri wajah atau karakteristik wajah. Setiap wajah digambarkan oleh kombinasi linier *eigenfaces*. Pada tahun 1987, Matthew Turk dan Alex Pentland dari Vision and Modeling Group The Media Laboratory di Massachusetts Institute of Technology menciptakan metode *eigenface* yang kemudian disempurnakan kembali pada tahun 1991.

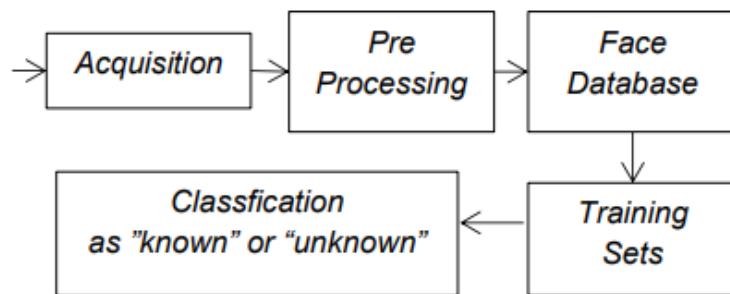
Matriks kolom wajah yang telah dimasukkan ke dalam database dibuat sebelum algoritma pengenalan wajah dijalankan. Vektor rata-rata matriks kolom (*mean*) ditentukan dengan membaginya dengan jumlah total gambar yang disimpan dalam database.

2.2.2 Absensi

Menurut beberapa pendapat di berbagai artikel dapat saya simpulkan bahwa absensi merupakan kegiatan pengambilan data jumlah peserta yang hadir dalam suatu kegiatan. Setiap kegiatan membutuhkan absensi guna mengetahui informasi peserta yang mengikuti kegiatan tersebut. Dalam kegiatan belajar mengajar absensi juga sangat dibutuhkan sebagai informasi bagi dosen mahasiswa yang mengikuti mata kuliahnya. Absensi merupakan daftar ketidakhadiran mahasiswa baik berupa sakit, izin ataupun alpa.

2.2.3 Face Recognition

Face recognition atau pengenalan wajah merupakan teknologi biometrik yang sering kita jumpai dan banyak diaplikasikan dalam sistem keamanan. *Face recognition* atau pengenalan wajah menggunakan kamera untuk menangkap gambar wajah seseorang yang kemudian dibandingkan dengan gambar wajah yang tersimpan di dalam database. Dari berbagai pendapat para ahli dapat peneliti simpulkan jika mekanisme pengenalan wajah adalah sebagai berikut:



Gambar 2.1 Sistem Pengenalan Wajah

- 1) Acquisition
Menampilkan citra wajah ke dalam sistem melalui media input.
- 2) Pre-Processing
Hasil dari proses akuisisi akan dinormalisasi (mengkonversi gambar kedalam bentuk greyscale), dengan melakukan pengolahan citra gambar wajah agar gambar wajah yang akan dikenali menjadi lebih baik.
- 3) Face Database
Tempat penyimpanan citra wajah dari kumpulan data pelatihan (training set).
- 4) Training Set
Pelatihan untuk mengoptimalkan kemampuan pengenalan pada wajah.
- 5) Classification
Membandingkan gambar wajah yang akan dikenali dengan database wajah yang sudah dengan bantuan pattern classifier.

2.2.4 XAMPP

Menurut Bunafit, (Putera & Ibrahim, 2018) XAMPP adalah paket PHP berbasis *open source* yang dibuat oleh komunitas *Open Source*. Pengguna tidak perlu lagi menginstal aplikasi lain saat menggunakan XAMPP karena memenuhi semua kebutuhan yang akan digunakan. Beberapa paket yang tersedia adalah Apache, MySql, PHP, Filezila, dan Phpmyadmin.

2.2.5 MySQL

MySQL merupakan RDBMS (*Relational Database Management System*) yang bersifat *open source*. Menggunakan bahasa SQL (*Structure Query*

Language) yaitu standar bahasa dalam pengolahan data. Menurut (Raharjo 2011) untuk menyimpan, mengatur, dan mengelola data pada aplikasi, pengembang aplikasi lebih memilih menggunakan database MySQL *Relational Database Management System* (RDBMS) untuk aplikasi desktop dan web. Beberapa manfaat MySQL dibandingkan RDBMS lainnya termasuk mudah, bebas biaya, stabil, dan portabel (dapat diterapkan ke beberapa sistem operasi).

2.2.6 PHP

Hypertext Preprocessor (PHP) merupakan bahasa pemrograman yang ditujukan untuk program berbasis web. Bahasa pemrograman PHP banyak digunakan orang diseluruh dunia karena PHP dapat berinteraksi langsung dengan *database*, *file* dan *folder*. Bahasa pemrograman PHP bersifat *open source* dan dapat berjalan pada sistem operasi yang berbeda atau disebut *cross-plaform* (Windows, Linux, MacOS). Bahasa server-side scripting yang disebut PHP bekerja bersama dengan HTML untuk menghasilkan situs web dinamis. server-side scripting berfungsi untuk menyediakan sintaks dan perintah yang akan beroperasi sepenuhnya di server tetapi disertakan dalam dokumen HTML (Sunarfrihantono, 2002).

2.2.7 JSON (Javascript Object Notation)

Douglas Crockford menciptakan standar komunikasi data ringan yang dikenal sebagai JSON (*Java Script Object Notation*). Fokus JSON adalah pada representasi data situs web [1][2][4][5]. JSON adalah perluasan dari fungsi javascript dan dibuat untuk menyederhanakan pertukaran data di situs web (Destian Wijaya dkk., 2015).

2.2.8 Flowchart

Bagan alir sistem (*flowchart system*) adalah diagram yang menampilkan pekerjaan saat ini yang sedang dilakukan oleh sistem secara keseluruhan, menjelaskan urutan langkah-langkah dalam sistem, dan menampilkan apa yang sedang dilakukan di dalam sistem (Riestiana & Sukadi, 2014)

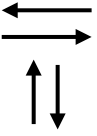



Menurut Anhar (dalam Riestiana & Sukadi, 2014) Flowchart adalah representasi grafis dari langkah-langkah dan urutan dalam metode program atau presentasi sistematis dari logika dan proses penanganan untuk berbagai kegiatan.

Alat teknik industri yang sering digunakan adalah diagram alir proses. Analisis sistem dapat menggunakan diagram alir ini untuk menjelaskan proses dalam suatu metode. Notasi atau simbol yang digunakan dapat dikategorikan ke dalam 3 (tiga) kelompok berikut:

1) *Flow Direction System* (Simbol Penghubung)

Simbol yang digunakan sebagai penghubung simbol lainnya. Nama lain dari simbol ini adalah *connecting line*, jenis simbol tersebut antara lain adalah:



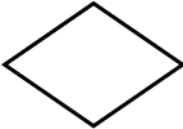


Tabel 2.2 Simbol *Flow Direction*

No.	Simbol	Nama	Keterangan
1.		<i>Arus/ Flow</i>	Menyatakan jalannya arus proses suatu kegiatan
2.		<i>Communication link</i>	Menyatakan adanya transisi data dari lokasi satu ke lokasi lainnya
3.		<i>Connector</i>	Menyatakan sambungan dari satu proses ke proses lainnya dalam lembaran yang sama
4.		<i>Offline connector</i>	Menyatakan sambungan dari satu proses ke proses lainnya halaman beda

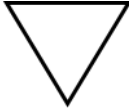

2) *Processing Symbols* (Simbol Proses)

Simbol yang berfungsi untuk menunjukkan berbagai jenis operasi pengolahan dalam suatu proses. Simbol tersebut antara lain adalah sebagai berikut:

Tabel 2.3 Simbol *Processing*

No.	Simbol	Nama	Keterangan
1.		Proses	Merupakan fungsi pemrosesan yang dilakukan oleh komputer untuk menghasilkan perubahan terhadap suatu data atau informasi
2.		Simbol manual	Menyatakan Tindakan/proses yang tidak dilakukan oleh komputer
3.		Logika/ <i>Decision</i>	Menyatakan kondisi tertentu dengan dua kemungkinan Ya/Tidak
4.		<i>Predifined process</i>	Menyatakan penyediaan tempat penyimpanan untuk pemberian harga awal
5.		Terminal	Menyatakan akhir atau permulaan suatu kegiatan




Tabel 2.3 Simbol *Processing* lanjutan

No.	Simbol	Nama	Keterangan
6.		<i>Offline storage</i>	Menyatakan bahwa data yang ada didalam proses ini akan disimpan kedalam media tertentu
7.		<i>Manual input</i>	Memasukkan data secara manual menggunakan <i>online keyword</i>

3) *Input/Output Symbol* (Simbol Input/Output)

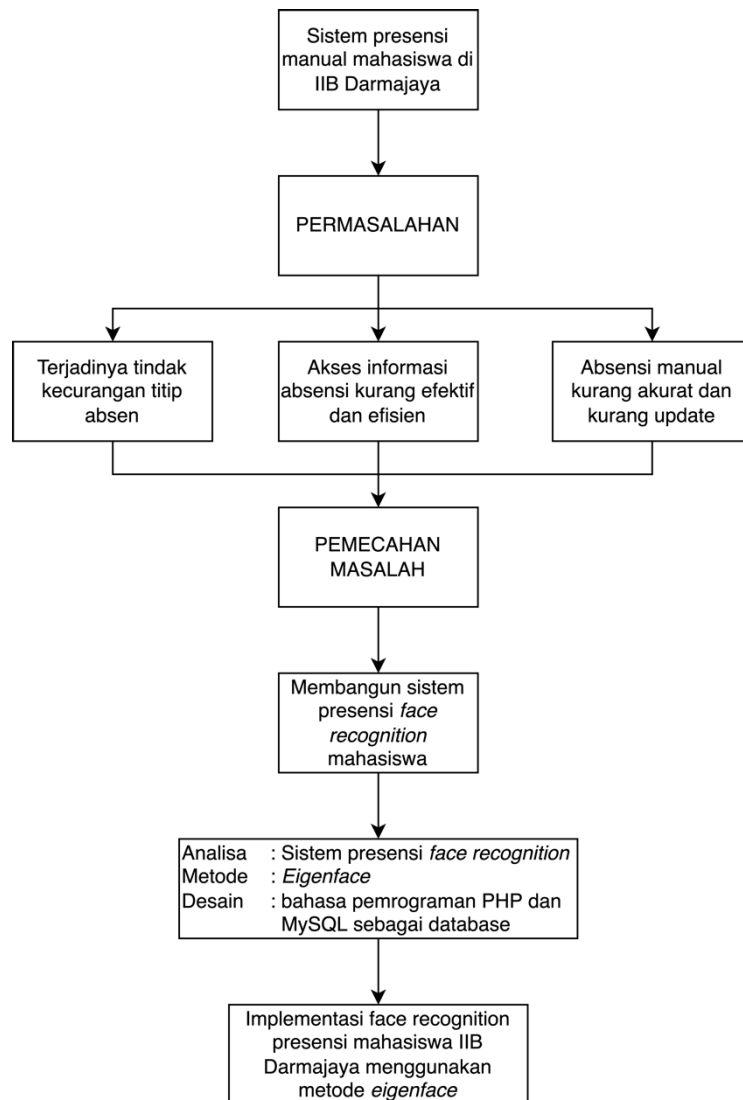
Simbol yang menyatakan jenis peralatan yang digunakan untuk media *input* atau *output*. Simbol tersebut antara lain adalah sebagai berikut:

Tabel 2.4 Simbol *Input/Output*

No.	Simbol	Nama	Keterangan
1.		<i>Input/Output</i>	Menyatakan suatu <i>input/output</i> tidak tergantung oleh peralatannya
2.		<i>Disk storage</i>	Menyatakan <i>input</i> berasal dari <i>disk</i> atau <i>output</i> yang akan disimpan kedalam <i>disk</i>
3.		<i>Document</i>	Mencetak dokumen yang dihasilkan dari suatu kegiatan

2.3 Kerangka Pemikiran

Alur pemikiran peneliti berisi bagan yaitu mengenai sistem absensi *face recognition* mahasiswa di IIB Darmajaya. Permasalahan yang timbul mengenai sistem absensi manual adalah akses informasi kurang efektif dan efisien, absensi manual kurang akurat atau kurang update, dan terjadinya tindak kecurangan mahasiswa dengan melakukan titip absen. Untuk pemecahan masalah yang ada diperlukan pengembangan sistem yaitu dengan membangun sistem informasi berbasis web serta dengan mendaur ulang sistem yang berjalan dengan menggunakan absensi *face recognition* menggunakan metode *eigenface*. Tindakan penyelesaian masalah menganalisa sistem absensi mahasiswa IIB Darmajaya, *face recognition* digunakan sebagai sistem presensi dengan menggunakan metode *eigenface*, serta desainnya menggunakan PHP sebagai bahasa pemrograman dan MySQL sebagai databasenya. Untuk hasil akhir penelitian diimplementasikan aplikasi *face recognition* presensi mahasiswa IIB Darmajaya menggunakan metode *eigenface* seperti gambar 2.2.



Gambar 2.2 Kerangka Pemikiran