

BAB II

LANDASAN TEORI

2.1 Tinjauan Pustaka

2.1.1 Internet of Things

Pada buku yang ditulis oleh (Mambang, 2021) dengan judul “BUKU AJAR TEKNOLOGI KOMUNIKASI INTERNET (*Internet of Things*)”, Teknologi *Internet of Things* (IoT) adalah gagasan tentang perangkat yang memiliki kemampuan untuk mengirimkan data tanpa terhubung dengan manusia, menggunakan internet sebagai medianya. Singkatnya, manusia dapat mengontrol benda dan perangkat IoT tersebut dari jarak jauh daripada harus mengontrolnya secara langsung, misalnya mengontrol perangkat menggunakan perantara aplikasi mobile ataupun website yang terkoneksi ke internet. Pengembangan big data dan penggunaan data center di Indonesia saat ini dan masa depan akan sangat didorong oleh konsep *Internet of Things* ini.

Internet of Things (IoT) juga didefinisikan sebagai konsep komputasi tentang objek sehari-hari yang terhubung ke internet dan memiliki kemampuan untuk mentransfer data melalui jaringan tanpa memerlukan interaksi manusia ke manusia atau manusia ke komputer. *Internet of Things* terdiri dari dua komponen utama yaitu: "Internet", yang mengatur konektivitas, dan "Things", yang berarti barang atau perangkat. Secara sederhana, "Things" yang memiliki kemampuan untuk mengumpulkan dan mengirimkan data ke Internet seperti sensor, dan "Things" lainnya seperti lampu atau alat output lainnya dapat mengakses data ini.

2.1.2 Pengenalan Wajah

Face Recognition atau pengenalan wajah adalah proses otomatis untuk mengidentifikasi atau memverifikasi identitas seseorang dari gambar wajah mereka. Ini melibatkan penggunaan teknologi komputer untuk mengekstrak fitur-fitur unik dari wajah seseorang seperti bentuk wajah, tekstur, warna kulit dan membandingkannya dengan wajah yang telah didaftarkan ke database untuk dicocokkan atau mengenali individu tersebut.

Pengenalan wajah adalah sistem yang digunakan untuk identifikasi yang menggunakan karakteristik wajah sebagai objek pengenalnya. Wajah memiliki fitur – fitur atau bagian seperti dahi, dagu, alis, mata, hidung, pipi, mulut, bibir dan memiliki ekspresi wajah, penampilan serta identitas yang dapat dilakukan ekstraksi oleh sistem sebagai parameter pengenalan wajah individu. Oleh karena itu, identitas akan sulit dikenali ketika wajah tertutup sebagian seperti ketika menggunakan masker. (Putra, 2021)

2.1.3 Python

Python ialah salah satu bahasa pemrograman tingkat tinggi yang sedang amat diminati pada masa kini. Diciptakan oleh Guido van Rosum dan dirilis perdana pada tahun 1991. Saat ini, tersedia untuk beragam platform dan dapat diunduh secara gratis melalui laman <https://www.python.org/>. Bahasa ini tak cuma digunakan oleh pemula, namun juga oleh para peneliti tingkat global. Ketersediaan pustaka yang amat luas membuat banyak masalah dapat diselesaikan dengan mudah menggunakan bahasa ini. (Kadir, 2019)

Python adalah bahasa pemrograman yang sangat serbaguna dan dapat digunakan dalam berbagai konteks, mulai dari pengembangan web hingga analisis data dan kecerdasan buatan. Ketersediaan pustaka dan kerangka kerja yang kaya, seperti NumPy, pandas, TensorFlow, dan scikit-learn, menjadikannya pilihan utama bagi para pengembang dan peneliti dalam mengatasi berbagai tantangan perangkat lunak dan ilmiah. Python dapat di install dengan cara mengunduhnya langsung melalui laman yang telah dilampirkan sebelumnya, atau dengan menggunakan perintah “*pip install python*” pada terminal sistem operasi.

2.1.4 Open CV

OpenCV merupakan pustaka berbasis “*Open Source*” yang mengandung lebih dari 500 fungsi yang ditujukan untuk menangani visi komputer. Pada Oktober 2018, pustaka *OpenCV* telah diunduh lebih dari 14 juta kali dan komunitas pemakaiannya mencapai lebih dari 47 ribu yang beberapa diantaranya adalah pekerja perusahaan besar seperti IBM, Microsoft, Google dan lain lain serta pusat riset seperti Stanford, MIT, Cambridge dan lain lain. (Kadir, 2019)

OpenCV sudah memiliki banyak fitur, seperti pengenalan wajah, pelacakan wajah, deteksi wajah, dan berbagai metode AI (*Artificial Intelligence*). Selain itu, *OpenCV* adalah kumpulan algoritma terkait penglihatan komputer yang sederhana untuk penggunaan tingkat rendah, dan tersedia untuk bahasa pemrograman C/C++, Python, Java, dan Matlab. *OpenCV* merupakan library dengan beragam fitur, termasuk Image and video I/O untuk membaca dan menciptakan file gambar dan video, serta Computer vision dan pengolahan citra digital yang memungkinkan eksperimen dengan berbagai algoritma standar seperti deteksi garis, tepi, dan pengenalan wajah, serta Modul computer vision tingkat tinggi dan metode untuk

aplikasi AI dan machine learning seperti pengenalan wajah atau *face recognition*. (Susim & Darujati, 2021)

2.1.5 Haar Cascade Classifier

Haar cascade classifier, atau dikenal juga sebagai fitur haar-like, adalah fitur-fitur berbentuk persegi panjang yang secara spesifik menunjukkan keberadaan suatu objek dalam gambar atau citra, khususnya untuk mendeteksi wajah. Algoritma ini memiliki kemampuan untuk mendeteksi objek, termasuk wajah manusia, dengan cepat dan secara *real-time*. Keunggulan dari algoritma Haar Cascade Classifier terletak pada kecepatan komputasinya, yang hanya bergantung pada jumlah piksel di dalam sebuah area persegi dari gambar. (Jamal Rosid, 2022).

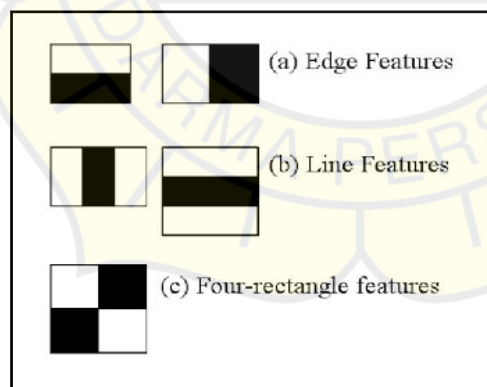
Haar Cascade adalah konsep yang dikembangkan oleh Paul Viola dan Michael Jones sebagai sebuah algoritma pembelajaran mesin untuk deteksi objek dalam gambar. Algoritma ini dapat dilatih untuk mengenali berbagai jenis objek dan terkenal karena kemampuannya dalam mendeteksi wajah dan bagian tubuh manusia. Terdapat empat tahap dalam algoritma *Haar Cascade*, yang meliputi: seleksi fitur, pembuatan gambar integral, pelatihan dengan metode *Adaboost*, dan proses klasifikasi dengan metode cascading (Budiman & Surbakti, 2019).

Haar Cascade menjadi sebuah konsep inovatif yang telah membuka jalan bagi algoritma pembelajaran mesin yang efektif dalam mendeteksi berbagai objek dalam gambar. Sebagai dasar yang penting untuk mendeteksi objek dengan akurat, *Haar Cascade Classifier* memiliki empat tahap, yaitu:

1. Seleksi Fitur

Seleksi Fitur pada Haar Cascade Classifier adalah proses awal di mana fitur-fitur yang paling relevan dan diskriminatif dipilih untuk membedakan antara objek yang ingin dideteksi dan latar belakang. Ada beberapa jenis fitur yang digunakan dalam seleksi fitur, di antaranya adalah:

- a. Edge Features: Fitur-fitur ini berfokus pada perbedaan intensitas piksel di sepanjang tepi objek. Mereka berupaya untuk menangkap perubahan tajam dalam intensitas warna di sepanjang tepi objek.
- b. Line Features: Fitur-fitur ini mencoba untuk menangkap pola garis atau batas yang ada dalam objek. Mereka mampu mendeteksi garis-garis lurus atau kurva yang mungkin hadir dalam objek.
- c. Four-rectangle features: Fitur-fitur ini terdiri dari dua area putih dan dua area hitam yang diatur secara berbeda dalam sebuah jendela. Fitur ini dapat membantu dalam menangkap variasi intensitas piksel di dalam objek.



Gambar 2. 1 Tipe dari Haar Features

2. Integral Image

Menurut Zhang & Zhang, 2010 pada penelitian berjudul “A Survey of Recent Advances in Face Detection”, Integral image, juga dikenal sebagai tabel jumlah total, merupakan algoritma yang digunakan untuk dengan cepat dan efisien menghitung jumlah nilai di dalam suatu subset persegi panjang dalam kisi-kisi. Kemudian, Viola dan Jones menggunakan integral image untuk menghitung dengan cepat fitur-fitur mirip Haar, seperti yang dijelaskan di bawah ini.

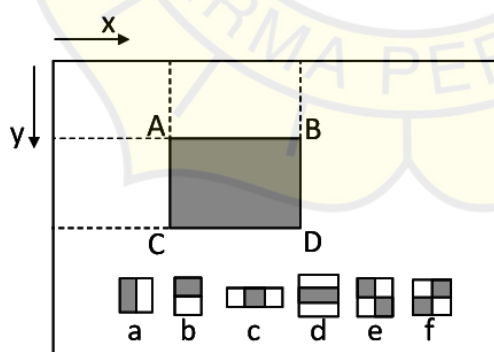
Integral image dibuat dengan cara sebagai berikut:

$$ii(x,y) = \sum_{x_0 \leq x, y_0 \leq y} i(x_0, y_0)$$

Dalam rumus ini:

- $ii(x,y)$ merupakan integral image pada lokasi piksel (x,y) .
- $i(x_0,y_0)$ adalah gambar asli

Penggunaan integral image untuk menghitung jumlah area persegi panjang mana pun sangat efisien. Hal ini secara signifikan mengurangi beban komputasi. Berikut merupakan ilustrasi dari integral image :



Gambar 2. 2 Ilustrasi bagaimana cara kerja Integral Image

Dalam hal integral image, jika nilai di titik 1 adalah A, di titik 2 adalah A+B, di titik 3 adalah A+C, dan di titik 4 adalah A+B+C+D, maka jumlah piksel di daerah D pada gambar di atas dapat dihitung dengan rumus:

$$\sum_{(x,y) \in ABCD} i(x,y) = ii(D) + ii(A) - ii(B) - ii(C)$$

Rumus diatas dapat diturunkan menjadi $4+1-(2+3)$.

3. *Adaptive Boosting (Adaboost)*

Pelatihan *Adaboost* adalah proses pembelajaran mesin di mana model klasifikasi dibangun dengan menggabungkan serangkaian model klasifikasi sederhana yang disebut klasifier lemah. *Adaboost* berfokus pada pengembangan model yang kuat dengan cara mengubah bobot atau pentingnya setiap sampel dalam set data latihan. Dalam pelatihan *Adaboost*, langkah pertama adalah menyediakan sebuah set fitur dan sebuah set data latihan yang terdiri dari citra positif dan negatif. Ada banyak pendekatan machine learning yang dapat digunakan untuk mempelajari fungsi klasifikasi. Varian dari *Adaboost* digunakan baik untuk memilih satu set kecil fitur maupun melatih klasifier. Dalam bentuk aslinya, algoritma pembelajaran *Adaboost* digunakan untuk meningkatkan kinerja klasifikasi dari sebuah algoritma pembelajaran yang sederhana (kadang-kadang disebut sebagai lemah).

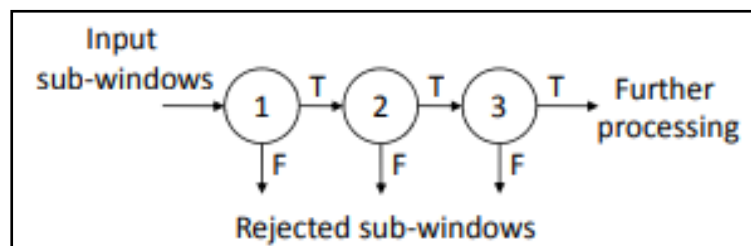
Untuk setiap fitur, *weak learner* mencari fungsi ambang batas yang dapat meminimalkan jumlah kesalahan klasifikasi. Weak classifier (klasifier lemah) dihasilkan dari proses ini dan terdiri dari sebuah fitur (feature f_j), sebuah ambang batas (threshold θ_j), sebuah polaritas (polarity p_j) yang menunjukkan arah tanda ketidaksetaraan dalam fungsi tersebut dan dimensi pixel yang digunakan pada gambar (x). (Viola & Jones, 2001)

$$h_j(x) = \begin{cases} 1 & \text{if } p_j f_j(x) < p_j \theta_j \\ 0 & \text{otherwise} \end{cases}$$

Rumus tersebut membantu dalam menjelaskan bagaimana klasifier lemah dalam *Adaboost* membuat prediksi terhadap sebuah contoh tertentu dengan tujuan akhirnya menemukan klasifier lemah yang tepat untuk digunakan dalam pembentukan model yang lebih kuat.

4. Cascade Classifier

Pada penelitian yang dilakukan oleh Septyanto et al., 2020 yang berjudul “APLIKASI PRESENSI PENGENALAN WAJAH DENGAN MENGGUNAKAN ALGORITMA HAAR CASCADE CLASSIFIER” dijelaskan dengan menggunakan banyak fitur yang ditentukan oleh *adaboost*, *cascade classifier* melakukan proses penyaringan subcitra dengan membentuk tingkatan penyaringan. Untuk menentukan keberadaan fitur objek wajah pada fitur yang dipilih, terdapat beberapa tingkatan klasifikasi. Selama proses klasifikasi, bagian subcitra melewati beberapa tahapan penyaringan. Pada tahap awal, subcitra yang tidak mengandung objek wajah akan ditolak, sementara yang mengandung objek positif akan dijadikan inputan untuk tahapan penyaringan berikutnya. Proses ini berlanjut hingga didapatkan subcitra yang diyakini sebagai objek wajah.



Gambar 2. 3 Skematik deteksi Haar Cascade

Klasifikasi filter dilakukan secara bertahap, pada filter pertama setiap subcitra akan dianalisis dengan menggunakan hanya satu. Ketika nilai fitur nya tidak sesuai dengan kriteria fitur maka akan ditolak dan proses klasifikasi filter dilanjutkan. Algoritma terus berlanjut ke sub window berikutnya dan mengevaluasi kembali nilai fitur seperti pada filter pertama. Jika nilai fitur sesuai dengan *threshold* yang ditetapkan, proses akan melanjutkan ke tahap filter berikutnya. Proses akan terus dilakukan berulang – ulang sehingga jumlah sub window yang lolos klasifikasi akan berkurang, mendekati citra yang terdapat pada sampel.

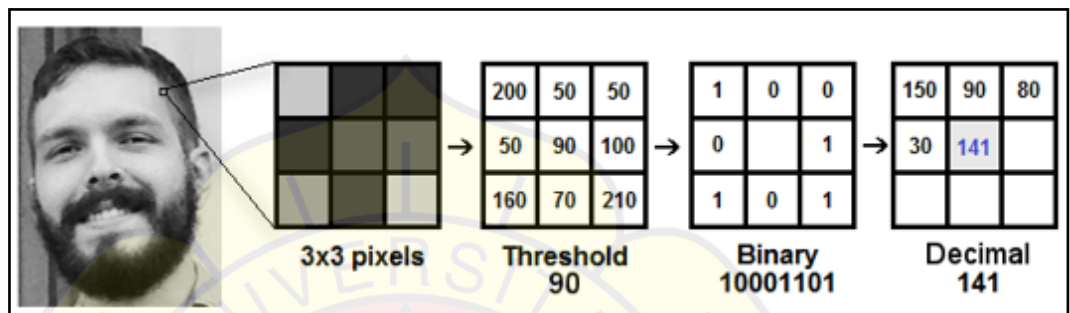
2.1.6 Local Binary Pattern Histogram

LBPH merupakan gabungan algoritma antara *Local Binary Pattern (LBP)* dengan *Histogram of Oriented Gradients (HOG)*. Dalam sistem ini, citra wajah yang diambil secara real-time menggunakan webcam akan dibandingkan dan disesuaikan dengan histogram yang telah diekstraksi dari citra wajah yang ada dalam basis data. Keberhasilan pengenalan wajah dalam sistem ini sangat dipengaruhi oleh faktor pencahayaan, sudut pandang wajah, dan keberadaan aksesoris pada wajah. (Tarigan & Kurniawan, 2022).

Local Binary Pattern Histogram (LBPH) adalah metode populer dalam pengolahan citra yang digunakan untuk ekstraksi fitur. Teknik dasarnya, *Local Binary Pattern (LBP)*, mengonversi setiap piksel citra menjadi kode biner berdasarkan perbandingan intensitas dengan tetangganya. Dengan membagi citra menjadi blok-blok kecil, histogram LBP dihitung untuk setiap blok, merekam frekuensi kemunculan setiap kode biner. LBPH kemudian menggabungkan histogram ini menjadi satu vektor fitur tunggal, yang dapat digunakan dalam

aplikasi pengenalan wajah, deteksi objek, dan klasifikasi citra. Keunggulan LBPH termasuk kestabilannya terhadap variasi pencahayaan karena hanya memperhitungkan perbedaan relatif antara intensitas piksel tetangga dan pusatnya.

LBPH memiliki beberapa tahap agar dapat mengidentifikasi wajah individu, berikut adalah tahap – tahapnya:



Gambar 2. 4 Prosedur dari LBPH

1. Pendefinisian Area Lokal

Langkah pertama adalah pendefinisian area lokal di sekitar setiap piksel dalam gambar ditentukan. Area lokal ini akan digunakan untuk mengekstraksi pola tekstur yang khas untuk setiap wilayah wajah dalam gambar. Pendefinisian area lokal ini memungkinkan algoritma LBPH untuk memperhitungkan konteks lokal saat mengidentifikasi wajah.

2. Penghitungan Nilai Threshold

Langkah selanjutnya adalah menentukan nilai threshold untuk binerisasi gambar. Nilai *threshold* ini akan digunakan untuk memisahkan piksel menjadi dua kelas berdasarkan perbedaan intensitas dengan piksel tetangganya. Nilai threshold ini penting untuk mengidentifikasi tepi dan detail tekstur dalam gambar. Nilai – nilai dihasilkan berdasarkan intensitas cahaya yang dihasilkan pada suatu piksel, ketika intensitas nya cerah atau

putih maka akan menghasilkan angka mendekati 255, sedangkan jika intensitas gelap atau hitam maka akan menghasilkan angka mendekati 0.

3. Konversi Citra Biner

Setelah nilai *threshold* ditentukan, gambar grayscale biner dihasilkan dengan mengaplikasikan nilai *threshold* pada setiap piksel dalam gambar. Piksel yang intensitasnya lebih tinggi dari nilai *threshold* akan diubah menjadi piksel putih, sedangkan piksel yang intensitasnya lebih rendah dari nilai *threshold* akan diubah menjadi piksel hitam. Gambar biner ini akan memperkuat pola tekstur dalam gambar.

4. Konstruksi Pola Biner

Dalam langkah ini, Local Binary Patterns (LBP) dihitung untuk setiap piksel dalam gambar biner. LBP adalah representasi pola tekstur lokal yang terdiri dari urutan biner yang menggambarkan hubungan intensitas piksel tetangga dengan piksel yang sedang diproses. LBP memungkinkan untuk mengidentifikasi pola tekstur lokal yang khas dalam gambar, termasuk pola tekstur khas yang mungkin terdapat pada wajah.

5. Menghitung Frekuensi pada Histogram

Langkah terakhir adalah menghitung histogram dari pola biner yang dihasilkan. Histogram ini merepresentasikan distribusi frekuensi kemunculan berbagai pola biner dalam gambar. Histogram ini digunakan sebagai fitur untuk melatih model atau untuk membandingkan pola tekstur antara gambar yang berbeda. Dengan menggunakan histogram LBPH, kita dapat mengidentifikasi wajah berdasarkan pola tekstur unik yang dimilikinya.

2.1.7 ESP32-CAM

ESP32-CAM adalah modul yang menggabungkan kemampuan mikrokontroler ESP32 dengan kamera OV2640, sehingga memungkinkan penggunaannya untuk aplikasi pengambilan gambar dan video. Mikrokontroler ESP32 yang terdapat pada modul ini memiliki kemampuan *WiFi* dan *Bluetooth*, serta dilengkapi dengan berbagai fitur tambahan seperti pemrosesan sinyal digital dan antarmuka I/O yang luas. Sementara itu, kamera OV2640 memiliki resolusi 2 megapiksel dan mendukung berbagai mode pengambilan gambar dan video. ESP32-CAM dapat diprogram menggunakan berbagai bahasa pemrograman seperti Arduino IDE dan MicroPython, serta dapat diintegrasikan dengan berbagai platform IoT untuk aplikasi pemantauan jarak jauh, pengawasan keamanan, dan pengenalan objek. Dengan kombinasi kemampuan tersebut, ESP32-CAM menjadi pilihan yang populer untuk proyek-proyek IoT yang membutuhkan kemampuan pengambilan gambar dan video secara mudah dan terjangkau.

Modul ini merupakan modul Wi-Fi yang dilengkapi dengan kamera. Dari modul ini, dapat digunakan untuk berbagai tujuan, misalnya untuk CCTV, pengambilan gambar, dan sebagainya. Fitur lainnya adalah bahwa ia dapat mendeteksi wajah dan pengenalan wajah. ESP32-CAM memiliki modul kamera ukuran kecil yang sangat kompetitif yang dapat beroperasi secara mandiri dengan sistem minimum. ESP32-CAM dapat digunakan secara luas dalam berbagai aplikasi IoT. Cocok untuk perangkat rumah pintar, kontrol nirkabel industri, pemantauan nirkabel, identifikasi nirkabel QR, sinyal sistem penempatan nirkabel, dan aplikasi IoT lainnya. (Rusimamto et al., 2021)

2.1.8 Sensor PIR (*Passive Infrared*)

Sensor PIR (*Passive Infrared*) adalah sensor yang digunakan untuk mendeteksi gerakan. Salah satu jenis sensor PIR yang populer adalah HC-SR501 seperti yang akan digunakan pada penelitian ini, memiliki tiga pin: VCC, GND, dan OUT. Sensor ini dapat dihubungkan ke pin digital pada mikrokontroler. Ketika sensor mendeteksi gerakan, ia akan mengirim sinyal listrik berupa "*HIGH*", dan "*LOW*" ketika tidak mendeteksi gerakan. (Kadir, 2018)

Sensor PIR (*Passive Infrared*) bekerja berdasarkan prinsip pendeteksian perubahan radiasi inframerah yang dihasilkan oleh benda-benda di sekitarnya. Ketika sebuah objek bergerak masuk ke dalam jangkauan sensor, perubahan pola radiasi inframerah akan terdeteksi, menyebabkan sensor menghasilkan sinyal keluaran. Sensor PIR sangat sensitif terhadap perubahan suhu, sehingga sering digunakan dalam aplikasi keamanan, pencahayaan otomatis, dan penghematan energi.

2.1.9 RFID

Radio Frequency Identification atau biasa disebut RFID adalah sistem yang menggunakan gelombang radio untuk membaca data secara nirkabel pada tag yang berbentuk seperti kartu kredit atau ktp. Terdapat 2 tipe tag yang berbeda yaitu tag aktif dan tag pasif. Tag aktif berbentuk seperti gantungan kunci dan memiliki baterai di dalamnya oleh karena itu jarak antara pembaca RFID dan tag aktif bisa lebih jauh dalam pembacaan datanya, sedangkan tag pasif memiliki bentuk seperti kartu kredit dan memiliki harga yang relatif murah dan biasanya digunakan untuk identifikasi palang pintu mobil atau sebagai kartu presensi pegawai, lalu tag pasif

ini harus berada dekat pembaca RFID agar pembaca mampu membaca data yang tersimpan pada kartu tag. (Sumber: Abdul Kadir, 2018, “Arduino dan Sensor”).

2.1.10 Google Drive

Google Drive adalah layanan penyimpanan berbasis cloud yang dikembangkan Google, Inc yang memungkinkan pengguna untuk menyimpan file secara online, sehingga file-file tersebut dapat diakses dari berbagai perangkat yang terhubung ke internet. *Google Drive* memberikan layanan gratis kepada penggunanya dengan kapasitas penyimpanan 15 GB dan akses mudah 24/7 dari perangkat apapun. (Safitri & Nasution, 2023). Kelebihan utama Google Drive adalah kemudahannya dalam penggunaan, kapasitas penyimpanan yang besar, dan ketersediaannya secara cross-platform, sehingga menjadi salah satu layanan penyimpanan cloud yang populer dan sering digunakan di seluruh dunia.

2.1.11 Jupyter

Jupyter Notebook adalah sebuah aplikasi web open-source yang memungkinkan pengguna untuk membuat dan berbagi dokumen interaktif yang menggabungkan kode komputasi, teks naratif, gambar, dan visualisasi data. Pengguna dapat menulis dan menjalankan kode dalam berbagai bahasa pemrograman, termasuk Python, R, Julia, dan banyak lainnya, langsung dari dalam notebook. Fitur utama dari Jupyter Notebook adalah kemampuannya untuk menggabungkan kode yang dieksekusi dengan hasilnya yang ditampilkan secara interaktif, memungkinkan analisis data dan eksplorasi konsep menjadi lebih intuitif dan terarah.

Jupyter Notebook merupakan tools untuk menyusun code – code Python dengan kelebihan *user friendly* dan mudah digunakan. Dengan jupyter anda dapat

mendokumentasikan sebuah pekerjaan, dimana coding dan dokumentasi bisa dilakukan dalam satu page dan disimpan dalam bentuk presentasi yang menarik. (Wahyono, 2021)

2.1.12 Firebase Realtime Database

Dalam penelitian yang dilakukan oleh Al-Kababji et al., 2019 dengan judul *“IoT-based fall and ecg monitoring system: Wireless communication system based firebase realtime database”* menjelaskan tentang Platform Firebase yang menawarkan sistem basis data yang di-hosting di cloud yang disebut *Database Realtime* yang mengikuti mekanisme NoSQL yang lebih sederhana meskipun kurang terstruktur dibandingkan basis data relasional tradisional. Ini memungkinkan penyimpanan data dalam waktu nyata dari set data besar yang dapat disinkronkan ke semua klien yang terhubung dalam akses basis data tersebut. Meskipun database baru bernama Firestore telah diluncurkan oleh Firebase, *Database Realtime* dipilih karena memiliki laten rendah, yang meningkatkan implementasi data visualisasi dalam waktu nyata.

2.1.12 Heroku

Heroku adalah platform PaaS cloud computing publik yang memungkinkan penyiapan konfigurasi server privat cloud computing, deployment layanan web service, serta pemanfaatan layanan web service tersebut melalui program berbasis web. (Wagito, 2022).

Heroku mendukung aplikasi yang ditulis dalam bahasa pemrograman populer seperti Ruby, Node.js, Python, Java, PHP, Go, dan Scala. Pengembang dapat mendorong (push) kode mereka ke Heroku menggunakan sistem kontrol versi Git,

dan Heroku akan secara otomatis mengelola siklus hidup aplikasi tersebut, termasuk penyediaan server, manajemen dependensi, dan skalabilitas.

2.1.13 Unified Modeling Language (UML)

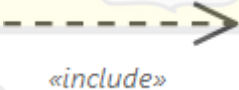
UML adalah sebuah bahasa yang berbasis grafik dan gambar yang digunakan untuk memvisualisasikan, menyesuaikan, membangun, dan mendokumentasikan sistem pengembangan software berbasis objek. UML adalah bahasa formal dengan komponen yang memiliki arti khusus untuk menghindari kesalahpahaman. UML mampu menggambarkan komponen penting sistem secara menyeluruh, berkat notasi yang jelas dan sederhana. UML adalah bahasa standar yang tidak terikat pada produk tertentu karena mudah digunakan dan fleksibel untuk menangani sistem besar atau kecil. (Sumirat et al., 2023)

UML atau Unified Modeling Language, adalah bahasa standar yang digunakan untuk memodelkan, merancang, dan mendokumentasikan sistem perangkat lunak, terutama dalam pengembangan perangkat lunak berbasis objek. UML memiliki manfaat untuk menyediakan bahasa yang dapat dipahami oleh semua orang yang terlibat dalam pengembangan perangkat lunak, seperti pengembang, analis bisnis, arsitek perangkat lunak, dan manajer proyek.

2.1.14 Use Case

Use case adalah sebuah komponen yang dapat membuat pembuat software dapat berkomunikasi dengan baik dengan konsumen. Use case berbentuk sebuah alur dengan simbol – simbol yang menggambarkan fungsi dari sebuah sistem yang akan atau yang sudah dibuat. (Sumirat et al., 2023)

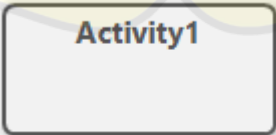
Tabel 2. 1 Komponen Usecase Diagram

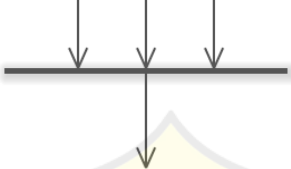
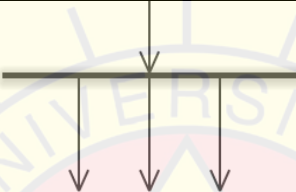
No	Nama	Simbol	Keterangan
1	Aktor		Simbol yang mewakili peran seseorang seperti jabatan pekerjaan, atau sistem, atau alat.
2	Use Case		Berisi interaksi atau hubungan yang dilakukan antara aktor dengan sistem.
3	Association		Merupakan garis penghubung antara actor dengan usecase.
4	Generalisasi		Merupakan garis yang menunjukkan spesialisasi actor untuk berpartisipasi dengan use case.
5	Include		Garis yang menunjukkan bahwa use case yang sedang dipertimbangkan memerlukan fungsionalitas yang didefinisikan di dalam use case lain untuk menyelesaikan tugasnya.
6	Extend		Garis yang menunjukkan bahwa use case merupakan fungsi tambahan dari use case lainnya ketika terjadi kondisi tertentu.

2.1.15 Activity Diagram

Activity diagram adalah salah satu jenis diagram dalam *Unified Modeling Language* (UML) yang digunakan untuk menggambarkan alur kerja atau aktivitas dalam suatu sistem atau algoritma. Diagram ini terdiri dari serangkaian aktivitas (tindakan) yang dihubungkan dengan aliran kontrol (garis panah) yang menunjukkan urutan atau alur kerja dari satu aktivitas ke aktivitas lainnya. Activity diagram dapat digunakan untuk menggambarkan proses bisnis dari awal hingga akhir, mengidentifikasi cabang dan pengulangan dalam proses, serta memvisualisasikan alur kerja yang melibatkan banyak entitas atau objek yang berinteraksi. Ini memungkinkan pemodel untuk memahami dengan jelas bagaimana suatu sistem berperilaku dan berinteraksi dengan elemen lainnya.

Tabel 2. 2 Komponen Activity Diagram

No	Nama	Simbol	Keterangan
1	<i>Initial Node</i>		Titik dimulainya aktivitas yang dilakukan dalam sistem aplikasi.
2	<i>Final Node</i>		Titik berakhirnya diagram aktivitas dalam sistem aplikasi.
3	<i>Activity</i>		Berisi tentang aktivitas yang dilakukan aktor atau sistem lainnya dalam aplikasi.
4	<i>Decision</i>		Sebuah simbol yang digunakan untuk membuat kondisi sebagai acuan

			pengambilan aksi atau keputusan.
5	<i>Line Connector</i>		Garis yang menghubungkan satu simbol dengan simbol lainnya.
6	<i>Join</i>		Sebuah simbol yang digunakan untuk menggabungkan aktivitas atau aksi.
7	<i>Fork</i>		Sebuah simbol yang digunakan untuk memecah aktivitas atau aksi
8	<i>Swimlane</i>		Memisahkan aktor yang bertanggung jawab atas aksi atau aktivitas yang ada didalam masing – masing swimlane.

2.2 Penelitian Terkait

Tabel 2. 3 Penelitian Terkait

No	Judul	Author	Tahun	Klasifikasi
1	<i>IMPROVED FACE DETECTION ACCURACY USING HAAR CASCADE CLASSIFIER METHOD AND ESP32-CAM FOR IOT-BASED HOME DOOR SECURITY</i>	Nauval Muhammad, Endro Ariyanto, Yogi Anggun Saloko Yudo	2023	JUPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika) Vol. 8, No. 1, Maret 2023, Pp. 154-161 SINTA 3
Hasil : Pada paper dengan judul “<i>IMPROVED FACE DETECTION ACCURACY USING HAAR CASCADE CLASSIFIER METHOD AND ESP32-CAM FOR IOT-BASED HOME DOOR SECURITY</i>” yang disusun oleh Muhammad et al., 2023, peneliti memilih untuk menerapkan algoritma <i>Haar Cascade Classifier</i> sebagai kerangka utama dalam studi mereka. Dalam eksperimen ini, sebanyak 12 data wajah yang berbeda diambil, dimana 6 diantaranya terdaftar ke dalam basis data yang digunakan untuk pelatihan algoritma. Proses pengenalan wajah dilakukan dengan menggunakan variasi jarak, yakni 30 cm, 40 cm, dan 50 cm, serta pada tingkat intensitas cahaya sebesar 130 lux. Hasil dari percobaan ini menghasilkan rata-rata akurasi sebesar 96,6%, mencerminkan tingkat keandalan yang tinggi dalam proses identifikasi wajah menggunakan metode yang diusulkan.				

2	<i>FACE RECOGNITION MENGGUNAKAN METODE HAAR CASCADE CLASSIFIER DAN LOCAL BINARY PATTERN HISTOGRAM</i>	Nurul Mega Saraswati, Rito Cipta Sigitta Hariyono, David Chandra	2023	Jurnal MEDIA ELEKTRIK, Vol. 20, No. 3, Agustus 2023 p-ISSN:1907-1728, e-ISSN:2721-9100 SINTA 4
Hasil : Pada paper yang berjudul “<i>FACE RECOGNITION MENGGUNAKAN METODE HAAR CASCADE CLASSIFIER DAN LOCAL BINARY PATTERN HISTOGRAM</i>” oleh Saraswati et al., 2023, Peneliti memanfaatkan algoritma <i>Haar Cascade Classifier</i> serta <i>Local Binary Patterns Histogram</i> (LBPH) dalam mengembangkan sistem pengenalan wajah. Dalam penelitian ini, terdapat 5 kelas wajah yang dilibatkan, dengan keseluruhan 1250 gambar wajah yang dikumpulkan. Setiap kelas memiliki 250 gambar wajah, di mana 200 gambar di antaranya digunakan sebagai data latih (training) dan 50 gambar lainnya digunakan sebagai data uji (test). Kemudian, dilakukan pengujian secara real-time sebanyak 75 kali dengan jarak 30 cm, 50 cm, dan 100 cm, yang menghasilkan tingkat akurasi sebesar 92%.				
3	<i>Face Recognition untuk Smart Door Lock menggunakan Metode Haar-Cascades Classifier dan LBPH</i>	Arvita Agus Kurniasari, Muhammad Farizul Imami Sudirman, Asif Mahardhika Ramadan, Firdaus Firmansyah	2023	SINTA 4

		, Nur Hakiki Damayant		
Hasil : Pada penelitian dengan judul “<i>Face Recognition</i> untuk <i>Smart Door Lock</i> menggunakan Metode <i>Haar-Cascades Classifier</i> dan LBPH” yang dilakukan oleh (Kurniasari et al., 2023), peneliti mengambil gambar sebanyak 100 data wajah yang diambil menggunakan ESP32-CAM secara otomatis menggunakan perulangan <i>while</i> pada program pendaftaran wajah. Setelah dilakukan training dan pengujian menggunakan 4 data wajah dan tingkat intensitas cahaya yang berbeda, dicapai akurasi dalam pengenalan wajah sebesar 90.25% pada intensitas cahaya 40 Lux dan dihasilkan akurasi sebesar 79.75% pada intensitas cahaya 20 Lux. Berdasarkan hasil pengujian, dapat disimpulkan bahwa kombinasi antara algoritma Haar Cascade Classifier dan LBPH menghasilkan sistem pengenalan wajah yang efektif. Namun, keberhasilan sistem ini sangat dipengaruhi oleh kualitas intensitas cahaya yang diterima oleh objek yang dideteksi. Hal ini menunjukkan bahwa intensitas cahaya memegang peranan penting dalam proses pendeteksian dan pengenalan wajah.				
4	<i>Face recognition</i> dengan metode <i>Haar Cascade</i> dan <i>Facenet</i>	Dewangga Mantara Saktia, Wahyu Sudoro Murtia, Ayu Kurniasaria, Jaml Rosida	2022	Indonesian Journal of Data and Science (IJODAS) ISSN: 2715-9930 Vol 3, No 1, March 2022, pp. 30-34 DOAJ

Hasil :

Berdasarkan hasil penelitian dan pembahasan tentang penggunaan metode *Haar Cascade* dan *FaceNet* dalam mendeteksi citra wajah, disimpulkan bahwa tingkat akurasi *face recognition* mencapai 80%, menunjukkan bahwa program ini mampu berfungsi dengan baik dan cukup akurat. Pengujian menggunakan enam dataset juga menghasilkan temuan bahwa program dapat mendeteksi dan mengenali wajah dengan baik.

5	Penerapan <i>Haar Cascade Classifier</i> dalam Mendeteksi Wajah dan Transformasi Citra Grayscale Menggunakan <i>OpenCV</i>	Syefrida Yulina	2021	Jurnal Komputer Terapan Vol. 7, No. 1, Mei 2021, 100 – 109 SINTA 4
---	--	--------------------	------	---

Hasil :

Hasil pengujian pada deteksi wajah menggunakan metode *Haar Cascade Classifier*, diperoleh akurasi keseluruhan sebesar 100%. Ketika objek wajah pada citra input berada dalam posisi frontal, akurasi mencapai tingkat optimal, sedangkan citra input dengan wajah dalam posisi non-frontal menghasilkan akurasi di bawah 50%. Perbedaan ini disebabkan oleh variasi posisi wajah dalam citra input. Tingkat kesalahan deteksi mencapai 24%. Sistem ini berhasil mengubah citra berwarna menjadi citra keabuan dengan menggunakan library *OpenCV*. Deteksi wajah sangat dipengaruhi oleh posisi wajah dalam citra.