

BAB II DASAR TEORI

2.1 TINJAUAN PUSTAKA

Penelitian dilakukan dengan beberapa jurnal yang menjadi referensi penulis. Penelitian pertama yang dilakukan oleh Efa Setiyoko pada tahun 2021. Penelitian tersebut melakukan pengujian secara *real-time* menggunakan Rasberri Pi 3, citra yang tersimpan pada database diuji menggunakan posisi wajah dengan *library dilb* dengan perintah untuk mendapatkan *crop* dari posisi wajah. Dalam penggunaan *library dilb* dan metode *eigenface*, sample wajah akan diubah dalam warna *grayscale* (hitam putih) dan dari metode *eigenface* akan mereduksi gambar dari high dimension menjadi *low dimension*. Pada 10 percobaan yang di lakukan didapat rata-rata waktu yang diperlukan untuk mengolah database sebesar 0,623 detik dan akurasi rata-rata untuk *face recognition* sebesar 72,5%. Penelitian tersebut masih memiliki kelemahan diantaranya, *library* yang banyak menjadikan waktu *booting* dari sistem berjalan lama. Dengan metode *eigenface* juga memerlukan Langkah-langkah yang lumayan Panjang dan proses deteksi wajah perlu memerlukan waktu yang lama [6].

Pada penelitian kedua yang di ambil dari Jurnal Ilmiah Fifo yang di lakukan pada tahun 2021. Peneliti membuat sistem keamanan melalui pengenalan wajah yang mengambil objek dari CCTV. Dalam perancangan sistem tersebut peneliti mengambil gambar untuk dimasukan kedalam dataset dan menggunakan metode *haar cascade classifier*. Hasil dari *face recognition* berupa pelabelan pada nama yang memiliki akurasi sebesar 40-69%. Pada penelitian tersebut pengambilan dataset menggunakan *hardware rasberri pi camera*. Ketika brankas tersebut tidak ada yang mengawasi dan terdapat seseorang yang tidak dikenal mendekati brankas, maka akan terdapat notifikasi yang di kirim ke telegram. Pada penelitian kedua terdapat kelemahan diantaranya *smart cctv* hanya mendeteksi wajah untuk dikenali dengan jarak 350 cm. Pada metode yang diambil masih terdapat kesalahan untuk mengenali identitas gambar masukan sebesar 1 : 40 yaitu 0,025% [7].

Penelitian ketiga yang dilakukan oleh Agung Yoke Basuki tahun 2019 [8] yang menggunakan pengenalan wajah dengan algoritma *eigenface* dan menggunakan *library OpenCv 2.4.9*. Database yang digunakan pada penelitian tersebut terdiri dari 10 wajah yang memiliki 2 sub folder positif, masing-masing citra wajah mengambil posisi yang berbeda. Pada Pengujian pembacaan foto dengan 4 kali percobaan menunjukkan hasil di atas *threshold* dimana hasil dari pembacaan tersebut adalah 3086.88, 3785.80, 3821.42, 2873,19. Berdasarkan pengujian pengiriman informasi ke telegram sebesar 3.71 detik dan rata-rata *feedback* telegram sebesar 3.786 detik.

Penelitian berikutnya yang keempat dilakukan oleh penulis Putra Lumbanraja E pada tahun 2023 [9] yang menggunakan mikokontroler ESP32-CAM yang dilengkapi dengan camera yang berjenis OV2640 sebagai perangkat utama dalam penelitian tersebut. Penelitian tersebut juga menggunakan LED untuk pencahayaan dalam meningkatkan akurasi pada pengenalan wajah. Metode dari penelitian yang dilakukan menggunakan penerapan PCA (*Principal Component Analysis*) pada *Eigenface*. PCA tersebut mereduksi dimensi pada citra yang dimensinya tinggi menjadi dimensi yang lebih rendah. Mikrokontroler ESP32-Cam sudah dilengkapi dengan *wifi* dan *bluetooth* yang digunakan sebagai monitoring yang terhubung dengan *blynk* untuk notifikasi keamanan.

Penelitian kelima yang diteliti oleh Prasetyo N tahun 2021 [10] dengan menggunakan pengenalan wajah dengan metode *Viola Jones*. Metode tersebut menggunakan proses klasifikasi dari suatu citra yang dapat dilihat dari nilai-nilai fitur melewati sebuah *classifier* yang sudah dihasilkan dari data *training*. Penelitian tersebut menggunakan Raspberry Pi 3B+ sebagai hardware utama dalam pemrosesan. Dalam pengujian pengenalan wajah yang dijalankan oleh sistem tersebut memperoleh akurasi pengenalan sebesar 60% dari 24 kali percobaan. Untuk pengujian pengaruh intensitas cahaya dengan ukuran 8 Lux memperoleh hasil akurasi sebesar 30% dan intensitas cahaya dengan 40 Lux memperoleh hasil akurasi sebesar 90%.

Penelitian berikutnya yang keenam dari Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer tahun 2022 [11] mengenai pengolahan pada *deep learning* untuk mempelajari data sebagai inputan dan melakukan pengolahan

transformasi lapisan *non-linear* dalam mendapatkan hasil output. Penelitian yang dilakukan tersebut menerapkan metode *deep learning Convolutional Neural Network* untuk melakukan pengenalan wajah dengan objek menggunakan 1386 gambar dataset. Pada hasil dengan 30 pengujian dan didapatkan hasil akurasi sebesar 83%, nilai presisi sebesar 81% dan hasil nilai recall 86%. Sedangkan waktu rata rata komputasi sebesar 8,19 detik.

Penelitian ketujuh yang dilakukan oleh Syafeeza A pada tahun 2020 [12] Penelitian tersebut menggunakan metode *Deep Learning* dan teknik pengenalan wajah *Convolutional Neural Network* (CNN). Arsitektur tersebut dibangun dengan beberapa lapisan diantaranya *Convolution, Maxpooling, Flatten, Dense, Activation and Dropout*. Sistem tersebut diimplementasikan dalam bahasa pemrograman python dan *Library Keras* dan menggunakan mikrokontroler ESP8266 yang telah dihubungkan dengan internet dan aplikasi *blynk*. Pada hasil akhir penelitian tersebut melakukan kombinasi antara *face recognition* dan IoT berhasil dilakukan dan mengirimkan notifikasi pengguna apabila terdapat wajah yang tidak dikenali. Pada kesimpulan penelitian tersebut memperoleh kesimpulan yang pertama dalam melakukan pemrosesan gambar memakan waktu yang lama dikarenakan penelitian tersebut menggunakan hardware raspberry pi 3.

Penelitian Berikutnya yang ke delapan di tulis oleh Arifin Mohammad tahun 2022 [13]. Pada penelitian tersebut menggabungkan antara sidik jari dan *face recognition*. Dalam pengenalan wajah tersebut metode yang digunakan adalah HOG (*Histogram Of Oriented Gradient*) dalam melakukan verifikasi wajah berdasarkan histogram lokal yang dari orientasi *gradien* yang telah di beri bobot dengan *magnitude gradient*. Dalam metode pengenalan tersebut menunjukkan tingkatan akurasi pengenalan sebesar 80%. Cara kerja sistem tersebut adalah dengan mengidentifikasi ciri-ciri dari suatu wajah individu dan kemudian melakukan ekstraksi menggunakan metode DCT (*Discrete Cosine Transform*) dan akan dilakukan klasifikasi menggunakan metode *decision tree*.

Pada penelitian kesembilan yang dilakukan oleh Yulianto pada tahun 2023 [14]. Penelitian tersebut menggunakan dataset wajah sebanyak 50 gambar yang berbeda. *Hardware* yang digunakan dalam penelitian tersebut menggunakan Arduino UNO yang terhubung dengan LCD sebagai tampilan monitor dan *solenoid*

door lock sebagai penguncian brankas tersebut. Pada hasil yang diperoleh dalam penggunaan metode CNN. Di dalam CNN, fungsi dari aktivasi (*activation function*) digunakan untuk melakukan *mapping* terhadap nilai pada *feature map*. Peneliti menyimpulkan untuk hasil perbedaan wajah dengan kategori pengguna utama memiliki akurasi 90% sedangkan kategori pengguna kedua memiliki akurasi 0% dan kategori wajah tidak dikenal "*unknown*" memiliki akurasi 71.43 %. Kemungkinan dari hasil tersebut dikarenakan masalah pada sistem CNN mengalami overfitting dan web camera tidak fokus pada bagian wajah. Pengujian komunikasi antara Jetson Nano dan Arduino UNO memiliki delay sebesar 10 detik.

Penelitian kesepuluh yang ditulis oleh Fadly E tahun 2021 [15]. Penelitian menggunakan perpaduan pendeteksian wajah dan sensor RFID sebagai kunci keamanan dengan menambahkan sensor ultrasonik dan ESP32-CAM. Penelitian tersebut menggunakan perangkat ESP32-CAM sebagai akses utama untuk deteksi wajah sedangkan sensor RFID digunakan sebagai backup apabila terjadi eror pada ESP32-CAM. Pada pengujian Sensor *Ultrasonic* menunjukkan akurasi pendeteksian sebesar 89%. Deteksi wajah hanya menggunakan 1 citra wajah pemilik dan waktu respon untuk pemberitahuan telegram sebesar 6-8 detik. Pengujian EKTP yang dilakukan dengan sensor RFID terdapat 10 kali percobaan dan sensor RFID dapat membaca RFID dengan keakuratan sebesar 100%.

Dalam penelitian ini menggunakan metode *library OpenCV* yang terhubung langsung dengan kunci brankas. *Library OpenCV* merupakan salah satu sebuah API (*Application Programming Interface*) yang sangat familiar dalam pengolahan citra gambar *Computer Vision*. Citra gambar akan di proses melalui dataset yang dibuat yang nantinya algoritma ini terlebih dahulu akan diuji dengan gambar dataset dari wajah orang dan pengujian dilakukan dengan gambar yang diambil secara real time yang dihubungkan dengan perangkat camera. Sementara *Computer Vision* merupakan salah satu dari cabang bidang ilmu pengolahan citra gambar (*Image Processing*) yang memungkinkan computer dapat melihat seperti manusia layaknya mata manusia. Dengan adanya *Computer Vision*, komputer tersebut mampu mengambil keputusan, melakukan sebuah aksi, dan dapat mengenali sebuah objek tertentu. Salah satu pengeimpelentasian untuk penelitian ini adalah *Face recognition* [16].

Tabel 2.1 Penelitian Sebelumnya

Penulis	Judul	Tahun	Metode
Efa Setiyoko	Perancangan Pengaman Brankas Berbasis <i>Face recognition</i> Dengan Metode <i>Eigenface</i> Yang Terkoneksi Dengan Handphone	2021	Pengenalan Wajah PCA (<i>Eigenface</i>), <i>Library Dlib</i> dan Raspberry Pi 3
Anggraini Nenny, Martunus Fikriansyah, Marzuki Shofi Imam, Wardhani Luh Kesuma	Implementasi <i>Face Recognition</i> Dengan <i>OpenCV</i> Pada “ <i>Smart CCTV</i> ” Untuk Keamanan Brankas Berbasis IOT	2021	<i>Library OpenCV</i> , <i>Expression Classification</i> dan Raspberry Pi 4
Agung Yoke Basuki	Perancangan <i>Door Lock Face Recognition</i> Dengan Metoda <i>Eigenfaces</i> Menggunakan <i>Opencv2.4.9</i> Dan Telegram Messenger Berbasis Raspberry Pi	2019	Pengenalan Wajah <i>Eigenface</i> , <i>OpenCV 2.4.9</i> dan Raspberry Pi

Penulis	Judul	Tahun	Metode
Putra Lumbanraja E, Saniman, Tugiono	“Sistem Monitoring Keamanan Brankas Menggunakan <i>Face Recognition</i> Berbasis Mikrokontroler ESP32-CAM”.	2023	Pengenalan wajah <i>eigenface</i> dan ESP32-CAM
Prasetyo N, Gozali F, Dan J, Rambung R	Sistem Pengaman Brankas Menggunakan Pengenalan Wajah Berbasis Raspberry Pi	2021	<i>Alogaritma Viola Jones, Haar Like Feature</i> , dan Raspberry Pi 3B+
Fauzan Alfiandi M, Utaminingrum F, Rosana Widasari E	Perancangan Sistem Pengamanan Ganda pada Brankas menggunakan <i>Convolutional Neural Network</i> berbasis Raspberry Pi	2022	Deep Learning <i>Convolutional Neural Network (CNN)</i> dan Raspberry Pi
Syafeeza A, Mohd Fitri Alif M, Mohammad Kasai,	“ <i>Iot Based Facial Recognition Door Access Control Home Security System Using Raspberry Pi</i> ”.	2021	<i>Deep Learning Convolutional Neural Network (CNN)</i> dengan ESP8266 dan Raspberry Pi

Penulis	Judul	Tahun	Metode
Arifin Mohammad	Sistem Pengamanan Mesin ATM Dengan Menggunakan Pengenalan Sidik Jari Dan Wajah <i>Face Recognition</i> Untuk Meminimalisir <i>Cyberbanking Crime</i>	2022	Pengenalan Wajah <i>Histogram of oriented Gradient (HOG)</i> dan <i>decision tree</i>
Yulianto A, Andreas Willy, dan Baloi Sei Ladi M	Perancangan <i>Prototype</i> Brankas Menggunakan Sistem Pengenalan Wajah dengan Metode <i>Convolutional Neural Network (CNN)</i>	2023	<i>Convolutional Neural Network (CNN)</i> dan Arduino UNO
Fadly E, Wibowo S, Sasmito A	Sistem Keamanan Pintu Kamar Kos Menggunakan <i>Face Recognition</i> Dengan Telegram Sebagai Media Monitoring Dan <i>Controlling</i>	2021	Pendeteksian Wajah Menggunakan ESP32-CAM dan Sensor RFID

2.2 DASAR TEORI

Terdapat dasar teori yang mendasari penulis dan sebagai acuan dalam mengembangkan penelitian ini.

2.2.1. Brankas

Brankas merupakan sebuah alat yang biasa terbuat dari besi dan digunakan sebagai penyimpanan suatu barang dan aset-aset barang pribadi seperti surat-surat penting, uang dan hak berharga pribadi lainnya [17]. Kunci pengaman pada brankas terdapat dua macam diantaranya digital dan analog, tidak jarang sistem kunci pada brankas tersebut dipadukan.

2.2.2 Artificial Intelligence (AI)

Secara umum *Artificial Intelligence* (Kecerdasan Buatan) mencakup pengembangan mesin yang memiliki kecerdasan tertentu, yang melakukan fungsi seperti layaknya manusia. Fungsi tersebut termasuk kognitif, belajar, membuat keputusan dan mampu beradaptasi dengan lingkungan [18]. Di dalam Kecerdasan Buatan terdapat bagian didalamnya terdapat metode yaitu *Deep Learning* dan *Machine Learning*, Pada *Machine Learning* membantu menerapkan kecerdasan buatan pada sistem sedangkan *Deep Learning* membantu sesuatu mencapai tujuan pembelajaran mesin pada sistem yang lebih sistematis. Salah satu aplikasi dari *Machine Learning* seperti *Computer Vision* [19].

2.2.2.1 Machine Learning

Machine Learning adalah ilmu yang mempelajari tentang algoritma komputer yang dapat mengenali pola-pola di dalam data, bertujuan untuk mengubah berbagai data menjadi satu tindakan yang nyata dengan sedikit adanya campur tangan manusia. Dengan adanya *Machine Learning* manusia dapat menciptakan mesin (Komputer) yang dapat belajar dengan data yang ada, selanjutnya dapat membuat keputusan sendiri tanpa perlu adanya pemrograman ulang [20].

2.2.2.2 Deep Learning

Deep Learning berasal dari jaringan saraf konvensional tetapi *Deep Learning* lebih unggul dibanding pendahulunya. Pembelajaran *Deep Learning* membangun Model *multi-layer* dan menggunakan transformasi dan teknologi grafik secara bersamaan. Algoritma ini memiliki arsitektur representasi dan *multi-layer*, dimana untuk *layer* pertama mengekstraksi fitur tingkat rendah sedangkan untuk lapisan terakhir mengekstraksi fitur tingkat tinggi [21].

2.2.3 Computer Vision

Visi Computer atau yang sering disebut sebagai sub dari bidang *Artificial Intelligence* (Kecerdasan Buatan) dan *Machine Learning* (Pembelajaran Mesin) yang mampu melatih computer untuk memvisualisasikan, menafsirkan, dan merespon Kembali ke dunia visual dengan cara yang sama halnya seperti yang dipresentasikan pada penglihatan manusia. Tujuan mendasari *Computer Vision* adalah untuk menginterpretasikan nilai dari sebuah gambar, dengan melibatkan prosedur terbaru yang mencoba untuk meniru dari kemampuan penglihatan manusia. Dalam mengenali gambar *Computer Vision* juga memiliki cara yang sama seperti otak manusia. Otak manusia memperoleh makna semantik dari kumpulan gambar sedangkan CV tersendiri memvisualkan sebuah gambar menjadi penggambaran digital dari computer melihat objek dalam pixel. Sistem kerja mendasar dari *Computer Vision* adalah untuk mengklasifikasikan gambar dan bagaimana *Computer Vision* mengenali suatu objek [22].

2.2.4 Face recognition

Face recognition (Pengenal Wajah) adalah salah satu tugas pemrosesan dari gambar, dan mengidentifikasi sebuah gambar atau video tertentu. *Open-Source Library* adalah salah satu library yang digunakan pada umumnya dengan library dari *Dlib*. Salah satu algoritma dari *Dlib* adalah CNN (*Convolutional Neural Network*) dan HoG (*Histogram of Oriented Gradients*) [23]. Sedangkan teknik dari algoritma dari *OpenCV* untuk mengenali wajah atau mengekstrak ciri-ciri suatu

citra wajah diantaranya, dengan menggunakan metode Pengenalan Wajah *Eigen Faces*, *Fisher Faces* dan *Local Binary Pattern Histogram* (LBPH).

2.2.4.1 Metode *Eigenface*

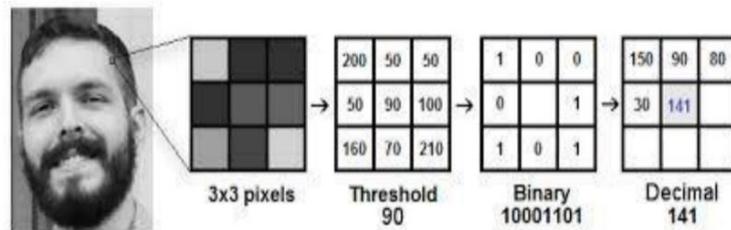
Asal kata *eigenface* berasal dari bahasa Jerman dengan penyebutan "*eigenwert*" yang dimana kata "*eigen*" berarti karakteristik dan kata "*wert*" berarti nilai. Oleh karena itu *Eigenface* adalah pola wajah algoritma pengenalan berdasarkan *Principal Component Analysis* (PCA). PCA merupakan salah satu metode reduksi fitur. Metode PCA ini digunakan sebelum melakukan proses klasifikasi, dalam hal tersebut PCA berfungsi sebagai mempercepat proses komputasi, oleh karena itu mengapa PCA dapat melakukan reduksi fitur pada setiap citra wajah. Setiap citra wajah yang telah direduksi oleh PCA akan diperoleh ciri atau karakteristik dari citra yang pada dasarnya memiliki warna dasar yang terdiri dari RGB (*Red, Green, Blue*) [24]. Ekstrak metode *eigenface* informasi yang relevan dari gambar wajah kemudian mengubahnya menjadi satu set kode wajah yang disebut vektor eigen. Kode wajah ini kemudian dibandingkan dengan dataset wajah yang menyimpan kode wajah dari pelatihan sebelumnya. Karakteristik wajah dapat juga dinyatakan sebagai vektor eigen, oleh karena itulah metode ini disebut dengan *eigenface*. Setiap wajah direpresentasikan secara linear kombinasi *eigenface*. Beberapa cara perhitungan *Eigenface* dibagi menjadi dua tahap, diantaranya tahap pelatihan (*training*) dan tahap pengenalan wajah [25].

2.2.4.2 Metode *Fisherface*

Metode dari Algoritma *Fisherface* merupakan penyempurnaan dari Metode *Eigenface* yang dikembangkan oleh Peter N. Belhumeur, algoritma *Fisherface* memberikan variasi kecerahan pada citra wajah. Algoritma ini mentransformasikan vektor dari ruang citra yang berdimensi- n ke ruang citra yang berdimensi- m dengan nilai $m < n$. Algoritma *Fisherface* bekerja lebih baik dibandingkan dengan *Eigenface* dalam situasi tersebut dimana kondisi pencahayaannya diubah [26].

2.2.4.3 Metode *Local Binary Patterns Histograms* (LBPH)

LBPH merupakan salah satu metode dalam penelitian kali ini dan sering digunakan pada *Computer Vision*, *Image Processing* dan Pengenalan Pola. Metode LBPH ini sesuai untuk ekstraksi sebuah fitur dikarenakan menggambarkan tekstur dan struktur dari gambar. Metode LBPH memiliki dasar LBP (*Local Binary Patterns*) dilakukan pembagian citra menjadi beberapa bagian. Kemudian dilakukan proses ekstraksi untuk mengambil satu pixel pada citra menjadi thresholding 8x8 untuk mendapatkan nilai histogramnya. Nilai dari *grayscale* akan menjadi pembanding dengan citra yang berdekatan, jika nilai pixel yang dihasilkan dari citra nilainya lebih besar dari nilai *grayscale*, maka nilai akan diubah menjadi 1, sebaliknya jika nilai pixel yang dihasilkan lebih kecil dari *grayscale*, maka nilainya diubah menjadi 0 [27]. Proses dalam ekstraksi citra di jelaskan pada gambar 2.1.



Gambar 2.1 Proses Ekstraksi LBP [27]

Pada gambar 2.1 pada citra wajah grayscale menunjukkan awal LBP menghasilkan 3x3 pixel pusat matriks dikenal sebagai *threshold* (ambang). Nilai-nilai tersebut dibandingkan dengan pernyataan di atas untuk mendapatkan biner 8 bit. Urutan bilangan biner pada hasil akhir akan diubah menjadi bilangan desimal yang digunakan sebagai tekstur deskriptor pixel tengahnya [26]. Terdapat beberapa tahapan dalam metode LBPH dalam pembuatan sistem diantaranya.

1. Mendefinisikan parameter (radius, tetangga, kisi x dan kisi y) menggunakan struktur parameter dari paket LBPH. Fungsi ini dipanggil untuk melewati struktur dengan parameter. Jika parameter tidak diatur, parameter akan diatur default.

2. Melatih metode dengan menjalankan fungsi *train* dari sebuah gambar dan sepotong label berdasarkan parameteranya. Semua gambar harus memiliki ukuran yang sama. Label digunakan sebagai ID untuk gambar tersebut, ketika memiliki lebih dari satu gambar dengan tekstur atau subjek yang sama, label harus sama.
3. Fungsi *train* pada awalnya akan memeriksa apakah semua gambar memiliki ukuran yang sama. Jika setidaknya satu gambar tidak memiliki ukuran yang sama, fungsi *train* akan mengembalikan kesalahan dan tidak akan dilatih.
4. Setiap piksel gambar yang sudah diambil berdasarkan jumlah *grid* (x dan y) yang dilewati oleh parameter. Setiap histogram (dari setiap kisi) hanya akan berisi 256 posisi (0~255) yang mewakili kemunculan setiap intensitas piksel. Setiap histogram perlu digabungkan untuk membuat histogram baru yang lebih besar. Misalnya *grid* yang dimiliki sebesar 8x8, maka 8x8x256 menjadi 16,384. Nilai tersebut adalah posisi yang akan terbentuk di histogram baru. Histogram terakhir mewakili karakteristik gambar asli [28]

Dari hasil analisis dan pembahasan dalam penelitian tersebut dapat ditarik kesimpulan bahwa algoritma LBPH memiliki akurasi yang akurat dan efisien. Algoritma LBPH memiliki rata-rata akurasi sebesar 83%, Algoritma *Eigenface* memiliki rata-rata akurasi sebesar 46% dan algoritma *Fisherface* rata-rata akurasi sebesar 54% [29].

2.2.5 Internet Of Things (IoT)

Istilah Iot (*Internet Of Things*) pada tahun 1999 pertama kali digunakan oleh Kevin Ashton selama presentasi Procter dan Gamble. Di dalam presentasinya ada beberapa penjelasan pemanfaatan teknologi RFID dalam pengelolaan barang dengan melengkapi barang dan perangkat khusus. Pengenalan sebenarnya awal mula dari IoT menurut perkiraan Cisco yaitu pada periode 2008 – 2009. Berkat adanya pengembangan teknologi IoT ini memudahkan manusia mempertoleh informasi tentang status mereka dan sekitarnya tetapi dengan cara yang lebih efisien IoT mengacu pada sistem perangkat yang saling terhubung dari satu perangkat ke perangkat lain dan memungkinkan untuk mengirim atau mentransfer data melalui jaringan tanpa adanya interaksi dari manusia [30].

2.2.5 Software

Adapun *software* yang digunakan dalam penelitian ini, untuk membuat sistem dari keamanan brankas. Diantaranya *software* yang ada di dalam raspberry *Thonny Python IDE*, Bahasa pemrograman *Python*, *Mailgun* dan *VNC Viewer*.

2.2.5.1 Python

Python merupakan salah satu bahasa pemrograman yang banyak digunakan oleh perusahaan untuk mengembangkan berbagai aplikasi berbasis mobile, desktop, maupun web, *Python* di ciptakan pada tahun 1990 oleh Guido van Rossum di Belanda [30]. *Python* salah satu bahasa pemrograman tingkat tinggi yang memiliki sifat interpreter, interaktif, *object-oriented*, dapat beroperasi di berbagai platform seperti Windows, Linux MacOS dan platform lainnya. *Python* banyak digunakan karena *python* juga termasuk bahasa pemrograman yang mudah di pelajari dikarenakan sintaks yang mudah dan jelas juga elegan, dapat dikombinasikan dengan modul yang memiliki struktur program data yang tergolong tingkatannya lebih tinggi. *Source Code* dari program *Python* pada biasanya akan di kompilasi dan dijadikan format perantara yang biasa disebut *bytecode*. Yang nantinya *bytecode* tersebut selanjutnya akan di eksekusi[31].

2.2.5.2 Library OpenCV

OpenCV merupakan pustaka yang berbasis “*Open Source*” yang memiliki lebih dari 500 fungsi. *Library* ini memiliki lisensi BSD (Distribusi Perangkat Lunak Berkeley). *OpenCV* dibangun menggunakan bahasa pemrograman C tetapi membungkinkan untuk menggunakan bahasa pemrograman python untuk mengakses *library OpenCV* tersebut [32]. Adapun beberapa Modul yang terdapat dalam *OpenCV* dapat dijelaskan sebagai berikut.

1. *core* : Modul ini mendefinisikan data dasar, struktur dan fungsi dasar yang digunakan oleh semua modul.
2. *imgproc* : pemrosesan gambar yang mencakup penyaringan, transformasi, geometris, konversi warna dan histogram pada gambar.
3. *objdetec* : Deteksi objek kelas dengan permisalan (wajah, mata, orang dan mobil)

4. *ml* : *Machine Learning Library* (MLL) adalah suatu sekumpulan kelas yang digunakan sebagai regresi, klasifikasi, dan pengelompokkan.
5. *shape* : dapat digunakan sebagai pencocokan, pengambilan atau perbandingan suatu bentuk [33].

2.2.5.3 Mailgun

Mailgun merupakan layanan email SMTP (*Simple Mail Transfer Protocol*) yang dirancang khusus untuk pengembangan perangkat lunak yang ingin menggunakan API (*Application Programming Interface*) untuk mengirim email massal. Layanan Mailgun tersebut juga dapat terintegrasi dengan pemrograman *Python* dan pemrograman lainnya. Dapat digunakan secara gratis hingga batas maksimal adalah 5.000 pengiriman email.

2.2.5.4 VNC Viewer

VNC adalah sebuah singkatan dari *Virtual Network Computing* yang memungkinkan komputer dapat mengakses komputer lainnya dengan terhubung dengan jaringan yang sama. VNC (*Virtual Network Computing*) juga dapat digunakan pada perangkat *hardware* seperti smartphone, laptop, komputer maupun raspberry pi yang sudah terpasang *software* tersebut.

2.2.6 Hardware

Adapun *hardware* yang diperlukan dalam penelitian kali ini diantaranya, Power Bank, Raspberry Pi 4, Kartu Memory 16 GB, Web Camera Usb, *Solenoid Door Lock*, Relay 2x9V Batrai.

2.2.6.1 Raspberry Pi 4

Raspberry Pi adalah sebuah perangkat seperti computer tetapi dengan versi yang lebih kecil. Komputer versi kecil ini mampu melakukan tugas secara efektif seperti versi computer yang asli. Modul dari raspberry pi 4 memiliki berbagai jenis prosesor dan hanya dapat menginstall sistem operasi dari sumber terbuka di dalamnya. Raspberry Pi 4 dapat mendukung bahasa pemrograman seperti *Python*,

C,C++,Perl dan Ruby [34]. Pada gambar 2.2 merupakan tampilan *hardware* dari raspberry pi 4.



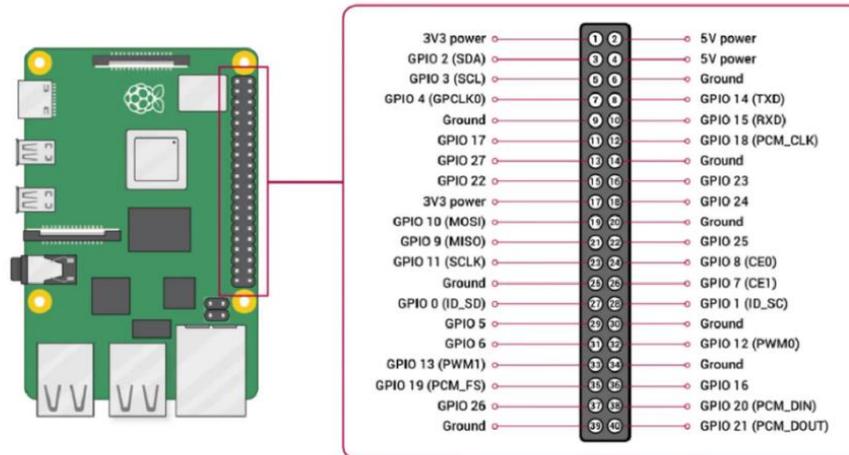
Gambar 2.2 Raspberry Pi 4 [34]

Pada tabel 2.2 merupakan spesifikasi dari Raspberry Pi 4 yang umum digunakan sebagai pengembangan dan pembelajaran dalam pemrograman komputer.

Tabel 2.2 Spesifikasi Raspberry Pi 4

Spesifikasi	Keterangan
Arsitektur	Broadcom BCM2711
Processor	Quad core Cortex-A72 (ARM v8) 64-bit SoC @ 1.5GHz
Memory/RAM	8GB LPDDR4-3200 SDRAM
GPU	Broadcom VideoCore VI
Wireless Adapter / LAN	2.4 GHz and 5.0 GHz IEEE 802.11ac wireless
Bluetooth	Bluetooth 5.0, BLE (Bluetooth Low Energy)
GPIO	40 Pin
Port USB	2 USB 3.0 ports; 2 USB 2.0 ports.
Card Storage	Slot Kartu untuk memuat sistem operasi dan penyimpanan data file
Jaringan	Gigabit Ethernet
Operating System	Debian GNU/Linux,Fedora, Arch Linux ARM, RISC OS

Pada Raspberry Pi 4 juga terdapat bagian pin seperti pada gambar 2.3 merupakan urutan pemetaan atau nama-nama pin yang terdapat dalam Raspberry Pi 4. Pemetaan dari pin dapat berguna dalam menggunakan *source code* yang akan dihubungkan dengan komponen seperti relay dan perangkat elektronik lainnya.



Gambar 2.3 Penjelasan *Port* Raspberry Pi 4 [34]

2.2.6.1.2 Web Camera Usb

Web Camera Xiaovv adalah sebuah perangkat external yang memiliki resolusi pixel HD (*High Definition*) yang dapat mengambil sebuah objek secara real time. Web Camera dalam penelitian ini digunakan sebagai pengambilan citra wajah dan pendeteksian wajah yang dapat terhubung dengan port USB dari Rasberri Pi 4. Pada gambar 2.4 merupakan tampilan *hardware* dari web camera.



Gambar 2.4 Xiaovv HD Web Camera

Pada tabel 2.3 merupakan spesifikasi dari Xiaovv HD Web Camera yang digunakan dalam pengambilan objek.

Tabel 2.3 Spesifikasi Xiavv HD Web Camera

Spesifikasi	Keterangan
Resolusi	1920 x 1080 px
Microphone	Built-in Microphone
Koneksi	USB 2.0
Support Operating System	Windows 7/8/10, Linux, MacOS
Format Media	H.264/ H.265/ MJPG/ NV12/YUY 12
Daya Input	5V / 1 A

2.2.6.1.3 Solenoid Door Lock

Solenoid door lock terbuat dari besi dan terdapat gulungan kawat tembaga yang di jadikan elektromagnetik dari logam di dalamnya. Saat tembaga diberi tegangan, gulungan tembaga akan menarik logam tersebut sebaliknya, ketika tidak diberi tegangan maka logam akan kembali seperti semula. *Solenoid door lock* biasa digunakan untuk sistem penguncian otomatis pada pintu rumah, brankas dan lainnya. *Solenoid Door Lock* Ini dapat dijalankan menggunakan daya sebesar 9-12 volt. Pada gambar 2.5 merupakan tampilan dari *solenoid door lock*.



Gambar 2.5 Solenoid Door Lock [35].

Pada tabel 2.4 merupakan spesifikasi yang terdapat pada *solenoid door lock* yang digunakan.

Tabel 2.4 Tabel Spesifikasi Solenoid Door Lock

Spesifikasi	Keterangan
Voltage	12 VDC (Dapat Digunakan 9-12 DC volt)
Waktu Aktivasi	1-10 detik
Dimensi	23,57 mm / 0,92 " x 67,47 mm / 2,65" x 27,59 mm / 1,08 "
Panjang Kabel	222.25mm / 8.75"