

**PINTU PINTAR PENDETEKSI WAJAH KUCING SECARA REAL-TIME
MENGUNAKAN METODE HAAR CASCADE PADA
MIKROKONTROLLER ESP-32 CAM**

SKRIPSI

WAHYU HIDAYAT SIREGAR

191401027



**PROGRAM STUDI S1 ILMU KOMPUTER
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI
UNIVERSITAS SUMATERA UTARA
MEDAN
2023**

**PINTU PINTAR PENDETEKSI WAJAH KUCING SECARA REAL-TIME
MENGUNAKAN METODE HAAR CASCADE PADA MIKROKONTROLLER
ESP-32 CAM**

SKRIPSI

Diajukan untuk melengkapi tugas dan memenuhi syarat memperoleh ijazah
Sarjana Komputer

Wahyu Hidayat Siregar

191401027



**PROGRAM STUDI S1 ILMU KOMPUTER
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI
UNIVERSITAS SUMATERA UTARA
MEDAN
2023**

PERSETUJUAN

Judul : PINTU PINTAR PENDETEKSI WAJAH KUCING
SECARA REAL-TIME MENGGUNAKAN
METODE HAAR CASCADE PADA
MIKROKONTROLLER ESP- 32 CAM.

Kategori : SKRIPSI

Nama : WAHYU HIDAYAT SIREGAR

Nomor Induk Mahasiswa 191401027

Program Studi : SARJANA (S-1) ILMU KOMPUTER

Fakultas : ILMU KOMPUTER DAN TEKNOLOGI
INFORMASI UNIVERSITAS SUMATERA
UTARA

Medan, 5 Januari 2024

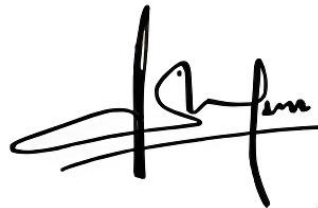
Komisi Pembimbing

Pembimbing II



Jos Timanta Tarigan, S.Kom.,M.Sc.
NIP. 198501262015041001

Pembimbing I

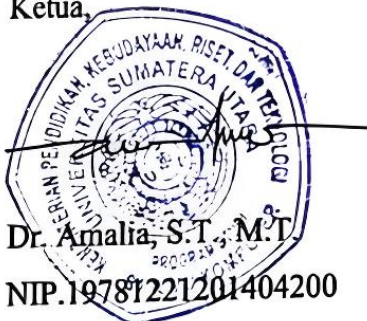


Dr. Ir. Elviawaty Muisa Zamzami, ST.,MT.
NIP. 197007162005012002

Diketahui/disetujui oleh

Program Studi S-1 Ilmu Komputer

Ketua,



Dr. Amalia, S.T., M.T.
NIP. 19781221201404200

PERNYATAAN

PINTU PINTAR PENDETEKSI WAJAH KUCING SECARA REAL-TIME MENGUNAKAN METODE HAAR CASCADE PADA MIKROKONTROLLER ESP-32 CAM

SKRIPSI

Saya mengakui bahwa skripsi ini adalah hasil karya saya sendiri, kecuali beberapa kutipan dan ringkasan yang masing-masing telah disebutkan sumbernya.

Medan, 13 Desember 2023



Wahyu Hidayat Siregar

191401027

PENGHARGAAN

Puji Syukur Alhamdulillah, penulis merasa sangat bersyukur dan berterima kasih kepada Allah SWT atas rahmat dan kekuasaan-Nya yang telah memberikan kemudahan dan kesempatan bagi penulis untuk menyelesaikan skripsi ini sebagai persyaratan untuk memperoleh gelar Sarjana Komputer dari Program Studi S-1 Ilmu Komputer Fakultas Ilmu Komputer dan Teknologi Informasi Universitas Sumatera Utara. Penulis ingin mengekspresikan rasa hormat dan terima kasih yang tulus kepada semua pihak yang telah memberikan bantuan dan dukungan dalam proses penulisan dan penyelesaian skripsi ini. Penulis menyadari bahwa tanpa adanya dukungan dari berbagai pihak, skripsi ini tidak akan pernah terwujud. Oleh karena itu ucapan terima kasih yang sebesar-besarnya kepada pihak yang telah membantu penulis menyelesaikan skripsi ini :

1. Kepada orang tua penulis Antoni Siregar dan Titin Mulya Ningsih yang selalu membnerikan doa, kasih sayang dan nasehat kepada penulis, sehingga penulis dapat berjuang menyelesaikan masa studi.
2. Bapak Dr. Muryanto Amin, S.Sos, M.Si. Selaku Rektor Universitas Sumatera Utara.
3. Ibu Dr. Maya Silvi Lydia, B.Sc., M.Sc. Selaku Dekan Fakultas Ilmu Komputer dan Teknologi Informasi Universitas Sumatera Utara.
4. Ibu Dr. Amalia, ST. MT. Selaku Ketua Program Studi S1 Ilmu Komputer Fakultas Ilmu Komputer dan Teknologi Informasi Universitas Sumatera Utara.
5. Ibu Elviawaty Muisa Zamzami selaku dosen pembimbing I yang telah memberikan arahan, kritik, nasihat dan motivasi kepada penulis selama proses pengerjaan skripsi.
6. Bapak Jos Timanta Tarigan, S.Kom., M.Sc. Selaku Dosen Pembimbing II yang telah memberikan arahan, kritik, nasihat dan motivasi kepada penulis selama proses pengerjaan skripsi.
7. Seluruh Bapak dan Ibu Dosen Program Studi S1 Ilmu Komputer yang telah memberikan waktu dan tenaga dalam transfer ilmu selama masa perkuliahan hingga kepada tahap penyusunan skripsi.

8. Seluruh Staf Pegawai Fakultas Ilmu Komputer dan Teknologi Informasi Universitas Sumatera Utara yang telah memberikan bantuan kepada penulis selama masa perkuliahan.
9. Adik kandung penulis Muhammad Zakwan Siregar dan Faiz Darmawan Siregar yang selalu membantu penulis ketika menyusun skripsi
10. Sahabat penulis semasa kuliah yang selalu bersama untuk urusan akademik yaitu Rojan Fauzan, atas bantuan kepada penulis selama masa perkuliahan terutama selama masa penyusunan skripsi.
11. Warga “Cucu Komputer” Levi Jovanatan, Rolando Febrianto, Khairul Imam, Abdurrahman Ghifari dan Matthew Hisar yang telah membantu penulis selama pembelajaran dan membantu penulis mengembangkan diri penulis hingga sekarang.
12. Keluarga besar Ilmu Komputer Laboratory Center, khususnya kepada Arian Syah Putra, Khairul Imam, Edo Harahap, Bang Arif Iskandar, Bang Alvin atas kesempatan berorganisasi di lingkungan kampus serta kerjasamanya yang baik dalam penyusunan skripsi ini.
13. Teman-teman keluarga besar angkatan 2019 Ilmu Komputer Universitas Sumatera Utara yang telah banyak memberi motivasi dan keseruan selama kuliah kepada penulis.
14. Dan semua pihak yang telah membantu dan menyemangati yang tidak dapat saya sebutkan satu per satu

Semoga hasil penelitian ini dapat memberikan kontribusi dan manfaat yang nyata bagi perkembangan ilmu pengetahuan dan teknologi, sehingga karya penulis dapat dipergunakan banyak orang. Penulis berharap penelitian ini dapat menjadi dasar untuk penelitian lanjutan dan menginspirasi penelitian lainnya. Semoga skripsi ini bermanfaat bagi pembaca dan masyarakat luas.

Terima kasih

Medan, 13 Desember 2023



Wahyu Hidayat Siregar
191401027

ABSTRAK

Penelitian ini bertujuan untuk mengembangkan sistem pengenalan wajah pada kucing kampung menggunakan kamera ESP32-Cam. Sistem ini dirancang untuk mendeteksi dan mengenali wajah kucing kampung secara otomatis dengan memanfaatkan teknologi pengolahan citra dan *Internet of things*. Tahap pengembangan mencakup perancangan perangkat keras dan perangkat lunak, termasuk integrasi sensor kamera, algoritma Haar Cascade, dan model pengenalan wajah. Pada pengujian sistem, dilakukan uji coba dengan menggunakan tiga objek kucing kampung sebagai sampel. Hasil pengujian menunjukkan kemampuan sistem dalam mendeteksi dan mengenali wajah kucing kampung dengan tingkat akurasi yang memadai. Penelitian ini memberikan kontribusi pada pengembangan sistem kecerdasan buatan untuk aplikasi pengenalan wajah pada hewan peliharaan. Pengujian yang dilakukan pada tiga ekor kucing menunjukkan bahwa sistem dapat digunakan dengan baik pada kondisi cahaya yang cukup, dan rekomendasi pendeteksian pada jarak 10 cm sampai dengan 30 cm dan pada sudut 0° sampai 30° ke kanan atau ke kiri.

Kata Kunci : Kucing, Pengenalan Wajah, ESP32-Cam, Haar Cascade, Internet of Things, Pengolahan Citra, Arduino IDE

ABSTRACT

In this research aims to develop a face recognition system for domestic cats using the ESP32-Cam camera. The system is designed to automatically detect and recognize the faces of domestic cats through the utilization of image processing technology and the Internet of Things (IoT). The development stages include the design of both hardware and software components, involving the integration of camera sensors, Haar Cascade algorithms, and face recognition models. In the system testing phase, trials were conducted using three domestic cat subjects as samples. The test results demonstrate the system's capability to detect and recognize domestic cat faces with satisfactory accuracy. This research contributes to the advancement of artificial intelligence systems for face recognition applications in pet animals. The conducted tests on three cats indicate that the system performs well under sufficient lighting conditions, with recommended detection ranges of 10cm to 30cm and angles at 0° until 30° to the right, or to the left.

Keyword : *Cat, Face Recognition, ESP32-Cam. Haar Cascade, Internet of Things, Image Processing, Arduino IDE*

DAFTAR ISI

PERSETUJUAN	Error! Bookmark not defined.
PERNYATAAN	iii
PENGHARGAAN	v
ABSTRAK	vi
ABSTRACT	viii
DAFTAR ISI	ix
DAFTAR TABEL	xii
DAFTAR GAMBAR	xiii
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah	2
1.4 Tujuan Penelitian	3
1.5 Manfaat Penelitian	3
1.6 Metodologi Penelitian	3
1.7 Sistematika Penulisan	4
BAB II LANDASAN TEORI	5
2.1 Pengolahan Citra	5
2.2 <i>Internet Of Things (IOT)</i>	5
2.3 Mikrokontroler Esp32-Cam	6
2.3.1 Arsitektur dan Konfigurasi Mikrokontroler ESP32-Cam	7
2.4 Relay	9
2.5 <i>Solenoid Door Lock</i>	10

2.6 Adaptor 12V	11
2.7 <i>Stepdown</i> LM2596	11
2.8 Algoritma Haar Cascade	12
2.8.1 Haar-Like Feature	14
2.9 Arduino IDE	15
2.9.1 Bahasa Pemrograman Arduino	15
2.10 Kucing	16
2.11 Penelitian Yang Relevan	17
BAB III ANALISIS DAN PERANCANGAN SISTEM	19
3.1 Analisis Sistem	19
3.1.1 Analisis Masalah	19
3.2 Analisis Kebutuhan	21
3.2.1 Analisis Kebutuhan Fungsional	21
3.2.2 Analisis Kebutuhan Non-Fungsional	21
3.3 Diagram Umum Sistem	22
3.3.1 Alur Mekanisme Penelitian	22
3.3.2 Arsitektur Sistem	24
3.4 Flowchart	26
3.4.1 Flowchart Sistem	27
3.4.2 Flowchart Perancangan <i>Smart Door</i>	27
3.5 Perancangan Sistem	28
3.5.1 Peralatan dan Bahan	28
3.6 Perancangan Mikrokontroler Esp32-Cam	29
3.6.1 Koneksi <i>Stepdown</i>	30
3.6.2 Koneksi Selenoid Door Lock	31

3.6.2 Keseluruhan Sistem	33
3.7 Perancangan Sistem Perangkat Lunak (<i>Software</i>)	34
BAB IV IMPLEMENTASI DAN PENGUJIAN	36
4.1 Implementasi Sistem	36
4.1.1 Implementasi Perancangan Esp32-Cam	36
4.1.2 Implementasi Perancangan Relay	37
4.1.3 Implementasi Perancangan <i>Solenoid Door Lock</i>	38
4.1.4 Implementasi Keseluruhan Sistem	38
4.2 Pengujian Sistem	39
4.2.1 Pengujian Fungsional Website	40
4.2.2 Pengujian Intensitas Cahaya	41
4.2.3 Pengujian Sudut dan Jarak	44
4.2.4 Pengujian Oklusi	48
4.2.5 Pengujian Algoritma Haar Cascade	52
4.2.6 Pengujian Filter Haar-like	52
4.2.7 Integral Image	53
4.2.8 Implementasi User Interface Website	53
BAB V PENUTUP	55
5.1 Kesimpulan	55
5.2 Saran	55
DAFTAR PUSTAKA	57

DAFTAR TABEL

Tabel 3.1 koneksi Esp-32 Cam ke Relay.....	32
Tabel 3.2 Koneksi <i>Stepdown</i> ke <i>Solenoid door lock</i>	33
Tabel 3.3 Koneksi <i>Stepdown</i> ke <i>relay</i>	33
Tabel 3.4 Koneksi <i>Stepdown</i> ke <i>Solenoid door lock</i>	34
Tabel 3.5 Koneksi <i>Solenoid Door Lock</i> ke <i>Stepdown</i>	35
Tabel 4.1 Pengujian fitur pada website.....	43
Tabel 4.2 Pengujian intensitas cahaya.....	44
Tabel 4.3 Pengujian sudut dan jarak.....	45
Tabel 4.4 Pengujian oklusi.....	47

DAFTAR GAMBAR

Gambar 2.1 <i>ESP32-Cam</i>	7
Gambar 2.2 Skema <i>ESP32-Cam</i>	8
Gambar 2.3 <i>Relay low / high level trigger</i>	11
Gambar 2.4 <i>Solenoid Door Lock</i>	11
Gambar 2.5 <i>Stepdown LM2596</i>	13
Gambar 2.6 Skema cascade pada citra.....	14
Gambar 2.7 Skema umum haar.....	16
Gambar 2.8 Tampilan Arduino IDE.....	17
Gambar 2.9 Kucing lokal.....	19
Gambar 3.1 Diagram Analisis Masalah.....	22
Gambar 3.2 alur mekanisme penelitian.....	25
Gambar 3.3 Arsitektur Sistem.....	26
Gambar 3.4. <i>Use Case Diagram</i>	27
Gambar 3.5 <i>Activity Diagram</i>	28
Gambar 3.6 <i>Flowchart Sistem</i>	29
Gambar 3.7 <i>Flowchart Pembuatan Sistem</i>	30
Gambar 3.8 koneksi Esp-32 Cam ke Relay.....	32
Gambar 3.9 Koneksi <i>Stepdown</i> ke <i>Solenoid door lock</i>	33
Gambar 3.10 Koneksi <i>Stepdown</i> ke <i>Relay</i>	34
Gambar 3.11 Koneksi <i>Solenoid Door Lock</i> ke <i>Relay</i>	34
Gambar 3.12 Koneksi <i>Solenoid Door Lock</i> ke <i>Stepdown</i>	35
Gambar 3.13 Rangkaian keseluruhan sistem.....	36
Gambar 3.14 Design antarmuka website.....	37

Gambar 4.1 Esp32 - Cam.....	39
Gambar 4.2 Relay 5v Low High Trigger.....	39
Gambar 4.3 Selenoid Door Lock.....	40
Gambar 4.4 Tampak depan dan belakang implementasi sistem.....	41
Gambar 4.4 Kucing belang putih, kucing orange, kucing belang hitam.....	42
Gambar 4.5 Halaman Utama Website.....	43

BAB I

PENDAHULUAN

1.1 Latar Belakang

Perkembangan teknologi informasi yang begitu maju mempengaruhi sikap serta perspektif kehidupan manusia, tidak lain pada segi pemahaman atau pengenalan objek. Pengenalan objek merupakan bagian penting pada manusia yang dapat digunakan untuk proses pengambilan informasi. Perkembangan teknologi kecerdasan buatan yang cepat, menyebabkan teknologi semakin mendekati atau bahkan melebihi kemampuan indera manusia (Muhammad H, 2022). Pengenalan objek juga dapat dimanfaatkan dalam berbagai bidang. Salah satu nya adalah dapat dimanfaatkan untuk mendeteksi wajah hewan peliharaan untuk dapat membuka atau menutup pintu secara otomatis dengan menggunakan kamera yang terintegrasi.

Metode *Haar like feature* atau yang dikenal sebagai *Haar Cascade Classifier* merupakan metode dengan pendekatan berbasis pembelajaran mesin yang menggunakan fungsi *cascade*, dimana fungsi ini dilatih dari berbagai citra positif dan negatif. Citra positif merupakan citra yang memiliki objek yang akan dideteksi, sedangkan citra negatif merupakan citra yang tidak memiliki objek deteksi (Sefrida, 2021). Metode ini menggunakan kumpulan *filter Haar* untuk mengenali pola wajah pada sebuah gambar. Metode ini biasanya digunakan untuk mendeteksi wajah manusia pada sebuah gambar atau video. Dan pada penelitian ini objek yang akan digunakan adalah hewan kucing, yang dimana kucing adalah hewan yang paling banyak digemari atau di pelihara orang karena lucu dan bisa dijadikan sahabat seseorang.

ESP32 adalah sebuah mikrokontroller yang populer yang dikembangkan oleh Espressif Systems. ESP32 dapat dijadikan pilihan untuk digunakan pada alat peraga *interface* mikrokontroler karena mikrokontroler ini memiliki *interface* yang lengkap, juga memiliki *WiFi* yang sudah tertanam pada mikrokontroler sehingga tepat untuk digunakan pada alat peraga atau *Internet of Things* (Hendra K, 2019). ESP32 juga mendukung pemrograman menggunakan bahasa C, C++, *MicroPython*, dan *Arduino*

IDE. Nantinya *board* ESP32 yang sudah di program menggunakan *MicroPython* akan dihubungkan dengan kamera sebagai sensor pendeteksi hewan kucing pada pintu pintar yang dibangun.

Berdasarkan latar belakang diatas dengan mengkombinasi system yang ada dari metode *Haar Cascade* dan mikrokontroller ESP32 sistem pengenalan objek dapat mudah dan cepat dikembangkan. Alat tersebut nantinya dapat mendeteksi wajah hewan kucing dari berbagai sudut pandang. Jika terdeteksi dengan benar alat ini akan memberikan output berupa pintu yang akan secara otomatis terbuka sebagai akses masuk atau keluarnya kucing, dan menutup kembali pintu ketika hewan (objek) sudah masuk ataupun keluar.

1.2 Rumusan Masalah

Salah satu tantangan utama yang dihadapi oleh pemilik hewan peliharaan yang memasang pintu khusus pada pintu utama adalah potensi terbukanya peluang bagi objek atau hewan liar lain untuk keluar dan masuk. Pemasangan pintu khusus untuk hewan peliharaan memang memberikan kenyamanan, namun juga membuka kemungkinan bagi hewan-hewan liar atau objek asing untuk dapat melewati pintu tersebut. Hal ini dapat menimbulkan kekhawatiran terkait keamanan pemilik rumah, karena hewan-hewan liar mungkin dapat memasuki rumah atau menciptakan situasi yang tidak diinginkan. Oleh karena itu, diperlukan solusi atau pengembangan sistem yang dapat mengatasi tantangan ini, memastikan keamanan hewan peliharaan, dan sekaligus mencegah akses tidak diinginkan dari hewan liar.

1.3 Batasan Masalah

Batasan masalah pada penelitian ini adalah sebagai berikut :

1. Fokus deteksi adalah pada wajah kucing
2. Penelitian tidak mempertimbangkan ekspresi wajah kucing secara mendalam dan intensitas cahaya, melainkan lebih fokus pada deteksi dalam kondisi umum.
3. Menggunakan sensor kamera pada mikrokontroller *esp32cam* sebagai platform utama, Relay 5v 1 cahnnel, Selenoid door lock 12v, dan stepdown
4. Program dirancang menggunakan *Arduino IDE 1.8.9*
5. Program yang dirancang berbasis *IoT*.

1.4 Tujuan Penelitian

Tujuan dari penelitian "Pintu Pintar Pendeteksi Wajah Kucing Secara Real-Time Menggunakan Metode Haar Cascade pada Mikrokontroler ESP-32" adalah untuk membuat sistem pintu pintar secara real time untuk mendeteksi wajah kucing.

1.5 Manfaat Penelitian

Penelitian ini memiliki beberapa manfaat ,antara lain sebagai berikut :

1. Mempermudah pengguna dalam memberikan akses ke dalam rumah secara aman tanpa harus membuka pintu secara manual atau menggunakan kunci.
2. Meningkatkan keamanan rumah dari gangguan hewan liar, terutama kucing yang dapat merusak atau mencuri makanan di dalam rumah.
3. Penelitian ini dapat menjadi dasar untuk pengembangan *Iot* serupa yang dapat mendeteksi dan mengenali wajah hewan peliharaan lainnya, seperti anjing atau burung, yang dapat digunakan untuk keperluan yang sama, seperti memberikan akses ke dalam rumah atau memantau kesehatan hewan peliharaan dari jarak jauh.

1.6 Metodologi Penelitian

Metodologi Penelitian yang digunakan adalah sebagai berikut :

1. Studi Literatur

Studi literatur dilakukan untuk mengumpulkan berbagai bahan referensi yang berkaitan dengan *Arduino Ide*, Mikrokontroller, board ESP32. Juga dari berbagai jurnal, artikel, buku, dan beberapa sumber referensi lainnya.

2. Analisis dan Perancangan

Pada tahap ini dilakukan perancangan arsitektur sistem yang sesuai dengan analisis permasalahan yang telah dilakukan sebelumnya. Pada penelitian ini, arsitektur umum dari perancangan sistem dirancang dengan membuat gambaran sistem mulai dari, flowchart algoritma, sketsa material, sensor kamera, dan mekanisme akses kontrol.

3. Implementasi

Implementasi dilaksanakan pada mikrokontroller yang telah di dipasangkan modul ESP32 yang telah dirancang, dan menancapkan kamera sebagai sensor pengenalan objek pada *board*.

4. Pengujian

Menguji apakah pintu otomatis yang di buat telah berhasil berjalan sesuai dengan keinginan dan melakukan perbaikan ataupun kesalahan jika masih terdapat kesalahan pada aplikasi.

5. Dokumentasi dan Penyusunan Laporan

Tahap terakhir yaitu membuat dokumentasi dan penyusunan laporan dari hasil penelitian yang telah dilakukan

1.7 Sistematika Penulisan

Berikut adalah sistematika penulisan dari penilitian Pintu Pintar Pendeteksi Wajah Kucing Secara Real-Time Menggunakan Metode Haar Cascade Pada Mikrokontroller Esp-32 Cam.

BAB 1 PENDAHULUAN

Latar belakang, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, metodologi penelitian, dan sistematika penulisan akan dijelaskan pada bab ini.

BAB 2 LANDASAN TEORI

Algoritma *Haar Cascade* serta *Image Processing* dan metode terkait penelitian akan dibahas dalam bab ini.

BAB 3 ANALISIS DAN PERANCANGAN

Proses menganalisis masalah dan mengembangkan solusi terbaik dijelaskan pada bagian bab ini.

BAB 4 IMPLEMENTASI DAN PENGUJIAN

Pengembangan sistem, perangkat dan pengujiannya secara lengkap terdapat pada bab ini.

BAB 5 KESIMPULAN DAN SARAN

Temuan pada penelitian yang dilakukan dibahas pada bagian ini sebagai beberapa ide masukan dan saran untuk penelitian di masa selanjutnya.

BAB II

LANDASAN TEORI

2.1 Pengolahan Citra

Pengolahan Citra (*image processing*) adalah suatu bidang studi yang mempelajari tentang bagaimana suatu citra diolah, dibentuk, dan dianalisis sehingga menghasilkan informasi yang dapat dipahami oleh manusia. Pengolahan citra meliputi berbagai teknik dan algoritma, seperti filtering, segmentasi, pengenalan pola, dan analisis statistik. Pengolahan Citra ini menggunakan bahasa pemrograman Python dan Pycharm, yang fungsinya melihat hasil perubahan citra (Ratna S, 2020). Dalam pengolahan citra juga mencakup matematika, statistik, fisika, komputer, dan teknik elektro. Beberapa sub-bidang dalam pengolahan citra antara lain pengolahan citra digital, pemrosesan video, pengenalan wajah, dan rekayasa citra.

Citra adalah representasi objek dua dimensi dari dunia visual, menyangkut berbagai macam disiplin ilmu yang mencakup seni, *human vision*, astronomi, teknik, dan sebagainya. Merupakan suatu kumpulan piksel-piksel atau titik-titik yang berwarna yang berbentuk dua dimensi (Hutahaean, Waluyo, and Rais 2019).

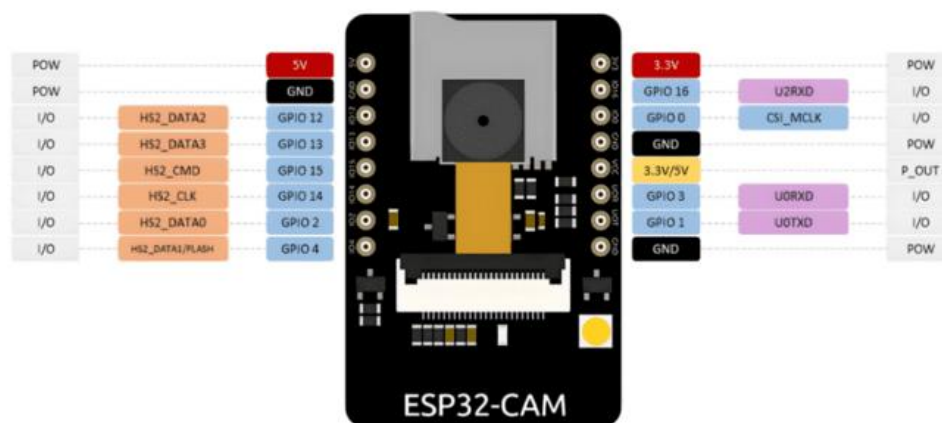
2.2 Internet Of Things (IOT)

Internet of Things (IoT) adalah sebuah konsep yang mengacu pada penghubungan perangkat yang terkoneksi ke internet, termasuk perangkat elektronik, kendaraan, peralatan rumah tangga, dan bahkan bangunan dan infrastruktur kota, sehingga mereka dapat saling berkomunikasi dan berbagi data secara otomatis. Dengan IoT, perangkat-perangkat ini dapat saling berinteraksi dan mengirim data secara real-time, sehingga memungkinkan pengguna untuk dapat mengontrol, mengawasi, dan mengumpulkan informasi dari jarak jauh. Contohnya adalah perangkat pengontrol suhu di rumah yang dapat diatur dari jarak jauh melalui ponsel pintar, atau kendaraan yang dapat mengirimkan informasi tentang kondisi mesin dan lokasi secara langsung ke pusat pemantauan.

Pada tahun 2020 perangkat yang terhubung dengan IoT diperkirakan lebih dari 20,8 miliar dan pertumbuhan eksponensial diprediksikan akan terus berlanjut di masa depan (Jie Ding, 2020). *Internet of Things (IoT)* mewakili revolusi Internet yang dapat menghubungkan hampir semua perangkat lingkungan melalui Internet untuk berbagi data mereka untuk membuat layanan dan aplikasi baru untuk meningkatkan kualitas hidup kita. (Hany F., 2019).

2.3 Mikrokontroller Esp32-Cam

Esp32 adalah sebuah mikrokontroler yang dikenalkan oleh Espressif System. Dimana kelebihan yang dimiliki oleh Esp32-Cam adalah sudah terdapat Wifi dan BLE (*Bluetooth Low Energy*) di dalamnya, yang akan sangat mempermudah pembuatan sistem IoT yang memerlukan koneksi *wireless*. Menggunakan Mikrokontroller ESP32 dapat menjadi sebuah solusi dalam upaya pemantauan dan sistem keamanan dalam suatu ruangan (Nizam M, 2022). Mikrokontroler ini juga *compatible* dengan Arduino IDE, Hal ini membuat ESP32 lebih mudah diakses dan diprogram oleh pengembang yang sudah terbiasa dengan Arduino.



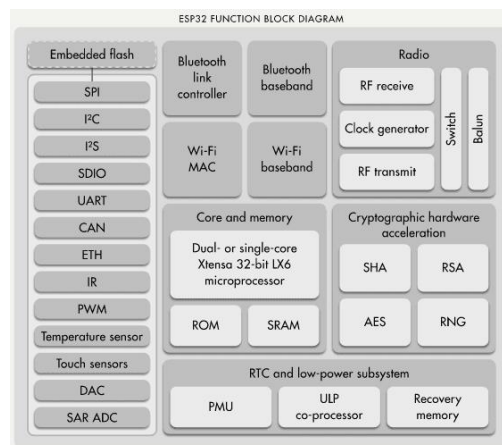
Gambar 2.1 *ESP32-Cam*

Tidak hanya memiliki dukungan konektifitas WiFi, dan Bluetooth. Modul yang berdiameter 27x 40,5 x 4,5mm ini memiliki kamera OV2640 dengan resolusi 2MP yang berfungsi sebagai photo atau video serta slot micro SD untuk menyimpan data di micro SD. Yang pada nantinya modul kamera tersebut yang akan digunakan sebagai sensor utama.

Module Esp32-Cam memiliki 3 pin GND, bisa di lihat pada gambar yang warna hitam. Sedangkan untuk power ada 2 opsi bisa menggunakan 3.3V atau 5V yang terdapat pada pin warna merah. Jadi, development board Esp32-Cam bisa menggunakan 3.3V atau 5V, tetapi lebih baik menggunakan 5V supaya tegangan lebih stabil. Memiliki 1 output pin, bisa 3.3V atau 5V bisa diatur dengan cara di jumper di boardnya.

2.3.1 Arsitektur dan Konfigurasi Mikrokontroler ESP32-Cam

ESP32-Cam adalah mikrokontroler yang memiliki arsitektur Xtensa LX6. Arsitektur ini merupakan arsitektur prosesor yang dikembangkan oleh Tensilica. Berikut ini adalah beberapa komponen pada mikrokontroler ESP32-Cam:



Gambar 2.2 Skema *ESP32-Cam*

1. Processors:

- Main processor: Tensilica Xtensa 32-bit LX6 microprocessor
- Cores: 2 or 1 (depending on variation)
- Clock frequency: up to 240 MHz
- Performance: up to 600 DMIPS
- Low-Power Dual-Core 32bit CPU

2. Wireless connectivity:

- WiFi: 802.11 b/g/n/e/i (802.11n @ 2.4 GHz up to 150 Mbit/s)
- Bluetooth: v4.2, Bluetooth Low Energy (BLE)

3. Memory:

- ROM: 448 KiB (*internal memory*)
 - SRAM: 520 KiB (*internal memory*)
 - RTC fast SRAM: 8 KiB (*internal memory*)
 - RTC slow SRAM: 8 KiB (*internal memory*)
 - eFuse: 1 Kibit (*internal memory*)
4. Embedded flash
 - 0 MiB (ESP32-D0WDQ6, ESP32-D0WD, and ESP32-S0WD chips)
 - 2 MiB (ESP32-D2WD chip)
 - 4 MiB (ESP32-PICO-D4 SiP module)
 5. External flash & SRAM: ESP32 supports up to four 16 MiB external QSPI flashes and SRAMs
 6. Spesifikasi Kamera OV2640
 - 2 Megapixel sensor
 - Array size UXGA 1622×1200
 - Output formats include YUV422, YUV420, RGB565, RGB555 and 8-bit compressed data
 - 4. Image transfer rate of 15 to 60 fps
 7. Pin Integrated
 - Module Micro SD ESP32
 - CLK GPIO 14
 - CMD GPIO 15
 - DATA0 GPIO 2
 - DATA1
 - GPIO 4
 - DATA2
 - GPIO 12
 - DATA3
 - GPIO 13

Untuk melakukan konfigurasi mikrokontroler Esp32-Cam dapat dilakukan menggunakan Arduino IDE atau dengan menggunakan bahasa pemrograman

yang mendukung ESP-IDF (*Espressif IoT Development Framework*), seperti bahasa C atau C++. Melalui konfigurasi, pengembang dapat mengatur pin GPIO, mengaktifkan dan mengatur komunikasi WiFi atau Bluetooth, dan menghubungkan dengan periferal lainnya sesuai dengan kebutuhan proyek yang sedang dikembangkan.

2.4 Relay

Relay adalah komponen elektronik yang berfungsi sebagai saklar elektronik untuk mengontrol suatu sirkuit elektronik. Cara kerja relay adalah memutus dan menyambung aliran listrik dalam rangkaian. Dalam sistem pintu pintar pendeteksi wajah kucing menggunakan Esp32-Cam ini, relay dapat digunakan untuk mengontrol *solenoid door lock* guna membuka dan menutup kunci secara otomatis. Relay terdiri dari dua bagian utama, yaitu elektromagnet dan saklar. Elektromagnet pada relay digunakan untuk menarik atau melepaskan saklar pada relay yang kemudian menghubungkan atau memutuskan sirkuit listrik.

Relay biasanya dikendalikan oleh arus listrik kecil yang dihasilkan oleh saklar, mikrokontroler, atau sensor, dan kemudian mengendalikan arus listrik yang lebih besar pada perangkat atau mesin yang dikontrol. Dan pada penelitian ini jenis relay yang digunakan adalah Relay 5v 1 Channel. Relay ini mampu menangani daya tinggi yang diperlukan untuk mengendalikan perangkat elektronik seperti *solenoid door lock* pada pintu pintar. Selain itu, relay ini sesuai dengan kebutuhan tegangan mikrokontroler, khususnya dengan mikrokontroler Esp32-Cam yang umumnya beroperasi pada tegangan 3.3V atau 5V. Keunggulan lainnya, di mana relay 1 channel umumnya lebih mudah diimplementasikan dan lebih ekonomis, yang sesuai untuk aplikasi kontrol tunggal seperti membuka atau menutup pintu. Relay ini juga mudah diintegrasikan dengan berbagai perangkat elektronik dan memiliki ukuran yang lebih kecil, mengurangi biaya dan memudahkan penempatan di dalam sistem pintu pintar. Meskipun relay ini hanya memiliki satu channel, namun itu sudah mencukupi untuk membuka dan menutup pintu dengan mudah



Gambar 2.3 *Relay low / high level trigger*

2.5 Solenoid Door Lock

Solenoid door lock (kunci pintu solenoid) digunakan sebagai motor untuk membuka dan menutup pintu pada penelitian ini. Solenoide adalah perangkat elektromagnetik yang mengubah energi listrik menjadi gerakan mekanis linear, dan kunci pintu solenoid dirancang khusus untuk penguncian dan pembukaan pintu.

Solenoid door lock memiliki konstruksi yang relatif sederhana, terdiri dari inti besi atau material ferromagnetik lainnya yang ditempatkan di dalam gulungan kawat. Ketika arus listrik dialirkan melalui gulungan kawat, tercipta medan elektromagnetik yang menyebabkan inti besi tertarik ke arah gulungan. Gerakan ini menciptakan gaya linier yang dapat dimanfaatkan untuk menggerakkan bagian mekanis dari kunci pintu, seperti penahan atau tuas penguncian.



Gambar 2.4 *Solenoid Door Lock*

Keuntungan penggunaan *solenoid door lock* dalam sistem pintu pintar melibatkan kecepatan respon yang baik. Solenoid dapat memberikan gerakan yang cepat dan akurat, membuka dan menutup pintu dengan efisien dalam waktu yang singkat. Selain itu, *solenoid door lock* dapat diendalikan dengan mudah oleh mikrokontroler seperti Esp32-Cam, memungkinkan integrasi yang lancar dengan sistem pintu pintar secara keseluruhan.

2.6 Adaptor 12V

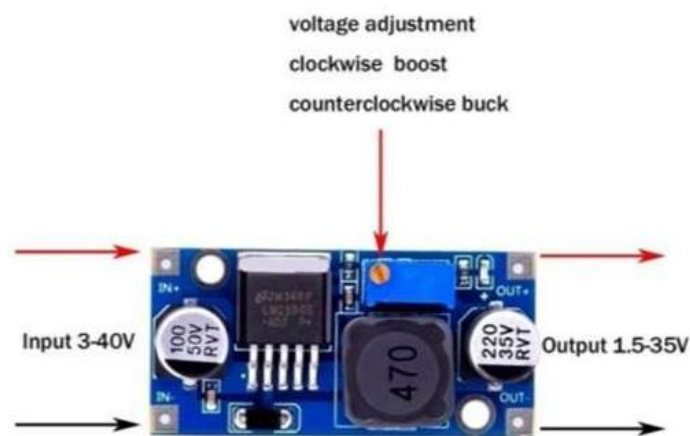
Adaptor 12V yang diintegrasikan dalam sistem ini bertujuan sebagai sumber daya eksternal yang menyediakan tegangan dan daya yang diperlukan oleh berbagai komponen sistem. Penggunaan adaptor ini memastikan kelancaran operasional mikrokontroler Esp32-Cam, solenoid door lock, relay, dan komponen lainnya. Dengan menyediakan tegangan yang stabil dan daya yang cukup, adaptor 12V memenuhi kebutuhan energi sistem secara keseluruhan, memastikan bahwa seluruh komponen dapat berfungsi optimal.

Adaptor 12V ini tidak hanya menyediakan tegangan yang diperlukan, tetapi juga memastikan daya yang mencukupi untuk komponen-komponen sistem. Beberapa adaptor 12V dilengkapi dengan regulasi tegangan dan arus, meningkatkan keamanan dan melindungi komponen sistem dari fluktuasi tegangan atau arus yang berlebihan. Pemilihan adaptor yang sesuai dengan standar keamanan dan sertifikasi yang relevan juga menjadi pertimbangan penting untuk menjaga keandalan dan keselamatan operasional sistem pintu pintar. Dengan adanya adaptor 12V yang handal, sistem dapat beroperasi secara optimal, menjadikan solusi ini cocok untuk aplikasi pintu pintar dengan kebutuhan daya dan tegangan yang spesifik.

2.7 Stepdown LM2596

Step-down adalah suatu proses atau perangkat yang digunakan untuk menurunkan (*down*) tegangan listrik dari tingkat yang lebih tinggi menjadi tingkat yang lebih rendah. Proses ini dikenal sebagai regulasi tegangan atau buck conversion. Dalam konteks tegangan listrik, step-down umumnya dilakukan untuk menyesuaikan tegangan sumber daya dengan kebutuhan perangkat atau sistem yang memerlukan tegangan yang lebih rendah untuk beroperasi secara efisien dan aman.

Step-down LM2596 disini, berperan sebagai regulator tegangan yang mengubah tegangan masukan yang lebih tinggi dari adaptor 12V menjadi tegangan keluaran yang sesuai dengan kebutuhan mikrokontroler Esp32-Cam dan komponen lainnya dalam sistem. Fungsi utamanya adalah menurunkan tegangan agar sesuai dengan persyaratan tegangan yang diinginkan oleh komponen-komponen tersebut. Dengan adanya LM2596, perbedaan tegangan antara adaptor dan kebutuhan komponen sistem dapat diatasi, sehingga sistem dapat beroperasi secara stabil dan sesuai dengan spesifikasi tegangan yang dibutuhkan.



Gambar 2.5 *Stepdown LM2596*

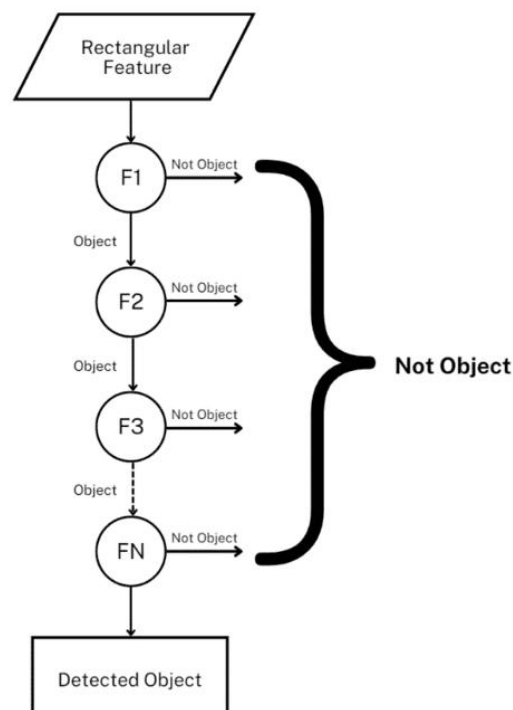
2.8 Algoritma Haar Cascade

Algoritma Haar Cascade merupakan salah satu metode deteksi objek yang paling populer dalam bidang pengolahan citra. Algoritma ini ditemukan oleh Viola dan Jones pada tahun 2001. Ide dasar di balik algoritma Haar Cascade berasal dari konsep fitur Haar. Fitur Haar adalah pola piksel yang dibentuk oleh perbedaan intensitas antara area piksel tertentu di atas dan di bawah garis horizontal atau vertikal. Fitur-fitur Haar ini dapat digunakan untuk membedakan bagian wajah dan bukan wajah. Setelah ditemukan, algoritma Haar Cascade menjadi sangat populer dan banyak digunakan dalam berbagai aplikasi deteksi objek, terutama deteksi wajah. Kecepatan dan akurasi deteksi yang tinggi menjadikan algoritma ini sangat berguna dalam pengenalan wajah, deteksi emosi, pengenalan gerakan, dan banyak lagi. Sejak ditemukan, algoritma Haar Cascade terus mengalami perkembangan dan

perbaikan. Dan telah banyak digunakan dalam pembuatan aplikasi deteksi wajah, deteksi mata, dan deteksi objek lainnya.

Algoritma Haar Cascade didasarkan pada penggunaan fitur Haar. Fitur Haar adalah sebuah kelompok filter khusus yang dapat digunakan untuk mengukur intensitas suatu area pada citra. Fitur Haar dapat digunakan untuk mengidentifikasi bagian-bagian penting pada sebuah objek. Algoritma tersebut mampu mendeteksi dengan cepat dan real-time sebuah benda ataupun objek (Geantika, 2020).

Cara Algoritma Haar Cascade memanfaatkan konsep pemilihan fitur yang adaptif dan pemindaian berjenjang (cascade). Pemindaian berjenjang memungkinkan proses deteksi dilakukan secara bertahap, dengan mengurangi jumlah subwindow yang harus dianalisis pada setiap tahap. Fitur-fitur Haar yang tidak relevan dihilangkan pada setiap tahap, sehingga mempercepat proses deteksi. Setelah itu, model klasifikasi dapat digunakan untuk melakukan deteksi pada citra masukan dengan memindai citra dengan berbagai ukuran window.

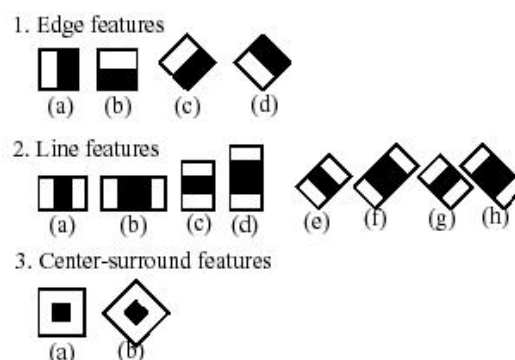


Gambar 2.6 Skema cascade pada citra

Hasil deteksi Haar-like Feature pada tingkatan filter yang tinggi dapat meningkatkan akurasi tetapi memerlukan waktu lebih lama. Oleh karena itu, Haar Cascade menggunakan classifier cascade, suatu urutan seleksi citra, untuk mengoptimalkan deteksi dalam waktu yang efisien. Setiap citra yang terdeteksi secara salah akan dibuang, sementara citra yang benar akan diteruskan ke tahap berikutnya dalam proses cascade. Dengan pola cascade ini, Haar Cascade dapat meningkatkan ketelitian deteksi dalam waktu yang relatif singkat, meminimalkan jumlah citra yang perlu diproses, dan menghasilkan hasil deteksi yang lebih akurat.

2.8.1 Haar-Like Feature

Haar like Feature merupakan metode yang sudah tidak asing lagi digunakan dalam pendeteksian sebuah objek. Nama Haar sendiri mengacu pada Haar Wavelet, sebuah fungsi matematika yang berbentuk kotak dan memiliki prinsip seperti pada fungsi Fourier (Nabila, 2020). Haar-like features (fungsi persegi), yang memberikan indikasi secara spesifik pada sebuah gambar atau image. Prinsip Haar-like features adalah mengenali objek berdasarkan dari nilai sederhana dari fitur tetapi bukan merupakan nilai piksel objek tersebut. Metode ini memiliki kelebihan yaitu komputasinya sangat cepat, karena hanya bergantung pada jumlah piksel dalam persegi bukan setiap nilai piksel dari sebuah image. Haar Like Feature akan memproses gambar dalam kotak-kotak, yang dimana dalam satu kotak terdapat beberapa pixel. Tiap kotak itu kemudian diproses dan didapatkan perbedaan nilai (threshold) untuk mendapatkan daerah gelap dan terang dengan mengurangi daerah yang gelap dan daerah yang putih, jika perbedaannya berada di atas threshold maka dikatakan fitur itu ada. Pola Haar dapat dilihat pada Gambar 2.7.



Gambar 2.7 Skema umum haar

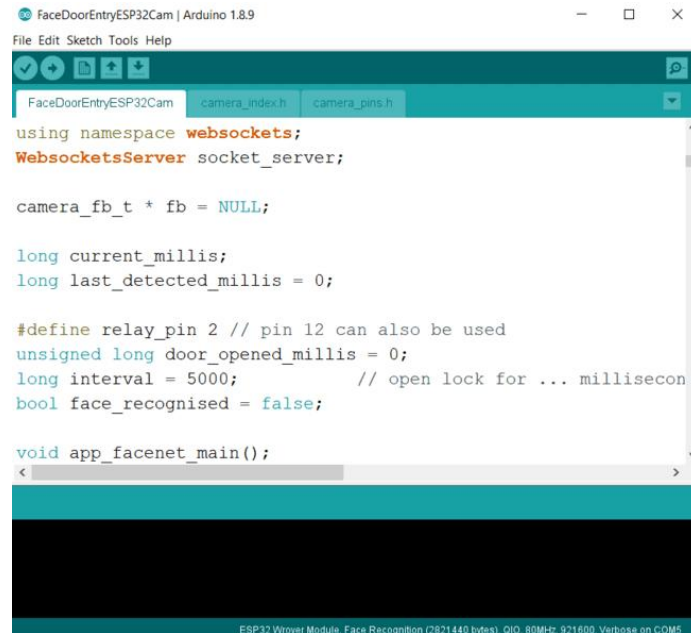
2. 9 Arduino IDE

Arduino IDE (Integrated Development Environment) adalah perangkat lunak yang digunakan untuk memprogram dan mengembangkan aplikasi pada platform Arduino. Arduino IDE dilengkapi dengan banyak library dan fungsi bawaan yang dapat digunakan untuk mempermudah pengembangan proyek. Library-library ini menyediakan fungsi-fungsi yang umum digunakan seperti komunikasi serial, kontrol PWM, operasi matematika, dan banyak lagi. Software yang dapat di unduh untuk berbagai sistem operasi ini juga kompatibel dengan berbagai jenis board Arduino, baik yang resmi maupun yang kompatibel dengan Arduino. Ini memungkinkan pengguna untuk mengembangkan aplikasi menggunakan berbagai jenis board dengan memanfaatkan fitur dan library yang tersedia.

2.9.1 Bahasa Pemrograman Arduino

Arduino merupakan *open source* yang ditujukan kepada siapa saja yang ingin membuat prototype peralatan elektronik interaktif berdasarkan hardware maupun software yang fleksibel dan mudah digunakan. Mikrokontroler diprogram menggunakan bahasa pemrograman arduino yang memiliki kemiripan syntax dengan bahasa pemrograman C.

Arduino Programming Language, dirancang untuk memudahkan pemula dengan syntax sederhana dan bersifat tingkat tinggi, menggunakan berkas program utama (sketch) yang memiliki fungsi utama `setup()` dan `loop()`, mendukung berbagai tipe data dan struktur kontrol seperti `if`, `else`, `while`, dan `for`, menyediakan library untuk kontrol perangkat keras, dan memiliki fungsi-fungsi khusus mikrokontroler yang mempermudah akses dan kontrol terhadap perangkat keras, dengan keseluruhan pengembangan dilakukan melalui Arduino IDE (Integrated Development Environment). Pada penelitian ini, bahasa pemrograman Arduino dipilih untuk mengimplementasikan algoritma pendeteksi wajah menggunakan metode Haar Cascade pada mikrokontroller ESP32-Cam.



Gambar 2.8 Tampilan Arduino IDE

2.10 Kucing

Kucing adalah hewan yang paling banyak digemari orang karena lucu dan bisa dijadikan sahabat seseorang. Hewan yang mempunyai nama latin *Felis Catus* ini juga merupakan hewan yang paling banyak di pelihara oleh manusia. Karena sifatnya yang memiliki pendengaran yang baik, tak heran jika kucing mempelajari suara pemiliknya dan lama-kelamaan tahu bahwa suara itu adalah suara yang berasal dari pemiliknya.

Menurut kepercayaan penduduk Jepang dan China, kucing dianggap sebagai hewan pembawa hoki. Kucing juga telah ada semenjak masa kerajaan-kerajaan kuno selaku fauna yang menemani seorang raja ataupun istri raja. Di Islam, kucing merupakan salah satu binatang kesayangan Nabi Muhammad.

Pada umumnya kucing hidup dengan pola liar dan dapat bertahan dengan mengandalkan sisa-sisa makanan manusia atau menjalankan insting berburunya terhadap tikus. Meskipun awalnya hidup di jalur independen, kucing juga memiliki potensi untuk diadopsi menjadi hewan peliharaan di rumah, memberikan kehangatan dan keceriaan dalam kehidupan sehari-hari. Keberagaman perilaku dan karakteristik

unik kucing membuat mereka menjadi bagian yang tak terpisahkan dari kehidupan di berbagai masyarakat di Indonesia.



Gambar 2.9 Kucing lokal

Keberadaan kucing tidak hanya menciptakan keseimbangan ekosistem dengan menjalankan peran alamiah sebagai pemburu tikus, tetapi juga memberikan nuansa kehangatan dalam hubungan antara manusia dan hewan. Meskipun sering ditemui di lingkungan pemukiman manusia, kucing menawarkan sisi independen yang memancarkan keanggunan dan ketangguhan. Selain itu, potensi adopsi kucing sebagai hewan peliharaan di rumah menjadi sebuah pilihan yang menarik, menciptakan ikatan emosional yang kuat antara pemilik dan hewan peliharaan. Keunikan dan keragaman perilaku kucing mencerminkan kedalaman interaksi antara manusia dan hewan, menciptakan kehidupan yang lebih berwarna dan bermakna bagi manusia.

2.11 Penelitian Yang Relevan

Berikut ini adalah beberapa penelitian yang relevan terkait dengan Implementasi Algoritma Haar Cascade Dan Mikrokontroler Esp32 Cam yaitu :

1. Penelitian yang dilakukan oleh Stefanus Adhie Nugroho, Rancang Bangun Sistem Deteksi Label Kardus Berbasis Model Kecerdasan Buatan YOLO dan EasyOCR serta ESP32-CAM, menghasilkan akurasi terendah sebesar 58,002%, dan rata-rata eror pembacaan 2,25%
2. Pada penelitian yang dilakukan oleh Nabila Dayu, Farida, dan Muchamad Kurniawan yang berjudul “Analisis Fitur Haar Menggunakan Algoritma Haar-Like Feature Pada Citra Kendaraan Bermotor”. Menghasilkan Nilai akurasi terbesar yaitu 100% dengan hasil pelabelan sebanyak 1 pada objek kendaraan mobil. Sedangkan pada pelabelan terbanyak 4 objek terdiri dari 3 kendaraan mobil dan 1 kendaraan bisa memperoleh hasil akurasi sebesar 90%. Nilai akurasi terendah sebesar 0% karena tidak adanya objek yang dikenali pada saat proses pelabelan.
3. Pada penelitian yang dilakukan Yusuf Fauzan yang berjudul “Kotak Penerima Paket Berbasis IoT menggunakan modul ESP32-Cam” menghasilkan tingkat kesesuaian fungsi sebesar 100% berfungsi sesuai sesuai target yang diinginkan penulis.
4. Penelitian yang dilakukan oleh Fini Febriani yang berjudul “Perancangan Alat Posisi Pada Hewan Peliharaan” memberikan akurasi 58%-98% dengan alat pelacak posisi hewan untuk mendeteksi posisi hewan peliharaan yang sebenarnya.
5. Selanjutnya penelitian tentang “Door Security System for Home Monitoring Based on ESP32”. Hasil evaluasi menunjukkan bahwa sensor pendeteksi gerak dapat mendeteksi pergerakan secara akurat hingga jarak 1,6 meter ke depan, dan pesan yang dipublikasikan antara smartphone dan kunci pintu terenkripsi dengan baik sehingga pesan terkirim dengan aman.
6. Penelitian lain tentang “Wi-Fi Door Lock System Using ESP32 CAM Based on IoT”. Dalam penelitian ini, ketika seseorang menekan bel pintu, pemilik menerima notifikasi di ponselnya dengan foto orang tersebut. Pemiliknya juga bisa membuka kunci pintu dari ponsel setelah memeriksa foto.

BAB III

ANALISIS DAN PERANCANGAN SISTEM

3.1 Analisis Sistem

Sistem yang dibuat bertujuan untuk mengembangkan sistem pintu pintar yang mampu mendeteksi wajah kucing secara real-time. Sistem ini didasarkan pada penggunaan metode Haar Cascade dan Esp32-Cam untuk mendeteksi wajah kucing dalam gambar atau video yang diperoleh melalui kamera. Mikrokontroler ESP32 digunakan sebagai otak sistem, yang mengendalikan proses deteksi wajah dan mengambil tindakan untuk membuka atau menutup pintu secara otomatis. Dalam analisis sistem, penelitian ini akan mengevaluasi performa sistem dalam hal akurasi pendeteksian wajah kucing, kecepatan deteksi real-time, dan responsifitas aksi pembukaan pintu. Selain itu juga akan dilakukan analisis penggunaan sumber daya sistem seperti penggunaan memori dan kebutuhan daya mikrokontroler Esp32-Cam. Hasil analisis ini akan memberikan pemahaman yang lebih mendalam tentang kinerja dan efektivitas sistem pintu pintar pendeteksi wajah kucing menggunakan metode Haar Cascade pada mikrokontroler Esp32-Cam, serta memberikan dasar untuk pengembangan sistem yang lebih optimal dan efisien di masa depan.

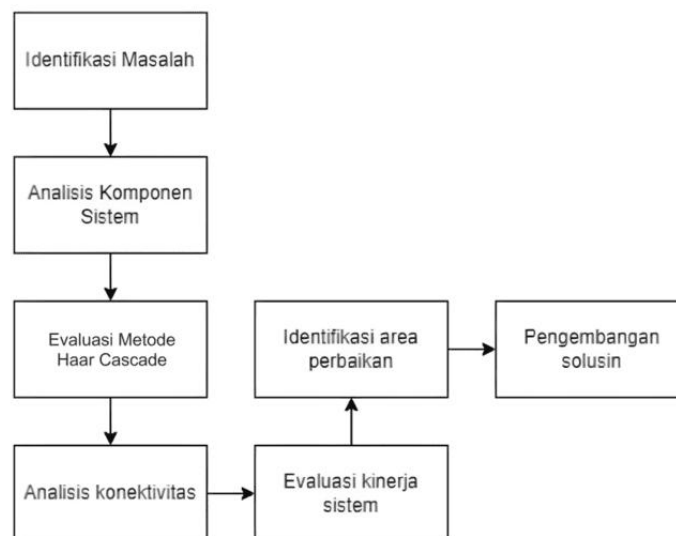
Namun, terdapat beberapa keterbatasan dalam sistem ini, seperti keakuratan deteksi wajah kucing yang dapat dipengaruhi oleh kualitas kamera, cahaya lingkungan, dan pose kucing. Selain itu, sistem ini memerlukan penggunaan tegangan yang cukup besar, seperti daya mikrokontroler Esp32-Cam dan memori. Oleh karena itu, diperlukan pengaturan yang tepat pada sistem agar dapat bekerja secara optimal dan efisien.

3.1.1 Analisis Masalah

Dengan kemajuan teknologi dalam Internet of Things (IoT), telah banyak muncul produk-produk yang dikembangkan, salah satunya adalah sistem *face recognition*. Sistem otomatisasi ini menggunakan sensor kamera untuk mendeteksi wajah yang mampu mencocokkan wajah manusia atau objek dari gambar digital atau bingkai video dengan database wajah untuk mengonfirmasi identitas seseorang. Namun, dalam pengembangan nya, masi sedikit penelitian

yang mengaplikasikan sistem tersebut pada wajah hewan. Oleh karena itu, pengembangan sistem ini akan menggunakan wajah kucing sebagai objek untuk mengautentikasi dan mengidentifikasi untuk mengolah input menjadi output.

Pada identifikasi masalah ini, akan dilakukan penelitian untuk memahami tantangan dan hambatan yang dihadapi dalam pembuatan sistem pintu pintar pendeteksi wajah kucing secara real-time menggunakan metode haar cascade pada mikrokontroller Esp32-Cam. Fokus penelitian ini adalah untuk mengidentifikasi kelemahan dan kekurangan yang mungkin terjadi dalam implementasi sistem saat ini, seperti kesalahan dalam akurasi, pencahayaan, atau masalah ketidakstabilan konektivitas.



Gambar 3.1 Diagram Analisis Masalah

Pada Diagram di atas menunjukkan langkah-langkah analisis masalah dalam sistem pintu pintar pendeteksi wajah kucing secara real-time menggunakan metode haar cascade pada mikrokontroller Esp32-Cam. Diagram tersebut mencakup identifikasi permasalahan, analisis komponen sistem, evaluasi metode Haar Cascade, analisis konektivitas, evaluasi kinerja sistem, identifikasi area perbaikan, dan pengembangan solusi. Setiap langkah dalam diagram dihubungkan dengan panah yang menggambarkan urutan analisis yang dilakukan. Hal ini membantu memvisualisasikan proses analisis yang terstruktur

dan mempermudah pemahaman tentang bagaimana masalah sistem diidentifikasi dan bagaimana solusi dapat dihasilkan.

3.2 Analisis Kebutuhan

Analisis kebutuhan bertujuan untuk mengidentifikasi dan mengevaluasi masalah dalam kerangka sistem. Di dalam analisis kebutuhan mencakup kebutuhan fungsional dan non-fungsional. Tujuan lain dari analisis kebutuhan adalah untuk menentukan tindakan yang tepat untuk memenuhi kebutuhan sistem serta memastikan sistem yang dibuat sesuai dengan kebutuhan yang sebenarnya.

3.2.1 Analisis Kebutuhan Fungsional

Kebutuhan fungsional sistem merupakan proses yang harus diikuti sistem agar dapat memenuhi tujuan dan fungsinya dengan baik dan benar. Berikut ini adalah persyaratan fungsional sistem :

1. Sistem dapat dijalankan dengan algoritma *Haar Cascade*
2. Sistem dapat mengenali fitur khusus yang membedakan wajah kucing dari objek lain
3. Sistem memiliki kemampuan untuk mengambil keputusan secara otomatis ,membuka atau mengunci pintu berdasarkan hasil dengan kemampuan pendeteksian dan klasifikasi wajah kucing.
4. Sistem yang diimplementasikan dapat berjalan pada mikrokontroler ESP-32-Cam.

3.2.2 Analisis Kebutuhan Non-Fungsional

Analisis kebutuhan nonfungsionalitas ini bertujuan untuk menentukan persyaratan yang tidak berkaitan langsung dengan fitur dan fungsi sistem, tetapi penting dalam mencapai kinerja, keandalan, dan pengalaman pengguna yang baik. Kebutuhan non-fungsional berisi batasan-batasan sistem, atau karakteristik dari sistem yang menjadikan sistem yang dibuat dapat digunakan oleh pengguna dengan baik. selain itu kebutuhan non-fungsional juga berisi batasan pengembangan sistem selanjutnya. Kebutuhan non-fungsional dari sistem yang dibuat adalah :

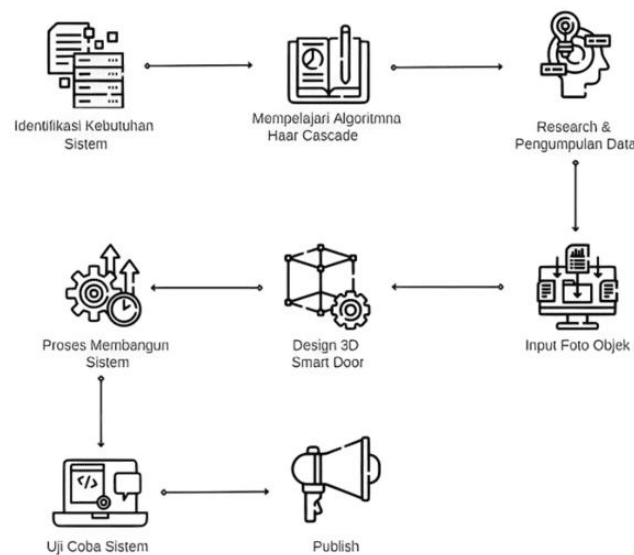
1. Batasan kucing yang dapat kenali pada penelitian ini adalah jenis kucing Kampung
2. Batasan objek yang dikenali adalah wajah, dan bentuk tubuh,
3. Alat dibuat minimalis dan juga sederhana agar tidak menghilangkan nilai estetika pintu utama rumah.
4. Sistem memiliki akurasi yang tinggi jika kondisi lingkungan memiliki pencahayaan yang mendukung.
5. Sistem harus dapat beroperasi secara konsisten dan handal dalam jangka waktu yang lama.
6. Sistem harus dapat dengan mudah ditingkatkan dan dapat diakses oleh pengguna atau perangkat IoT kapan pun diperlukan

3.3 Diagram Umum Sistem

Pada bagian diagram umum sistem ini, berisi alur mekanisme penelitian yang dimulai dari pemahaman konsep dan diagram sistem yang menjelaskan cara kerja sistem secara keseluruhan.

3.3.1 Alur Mekanisme Penelitian

Dalam alur mekanisme penelitian ini akan memberikan gambaran dari tahapan tahapan yang dilakukan dalam penelitian untuk mendapatkan hasil akhir yang diharapkan. Berikut adalah gambaran alur mekanisme penelitian ini :

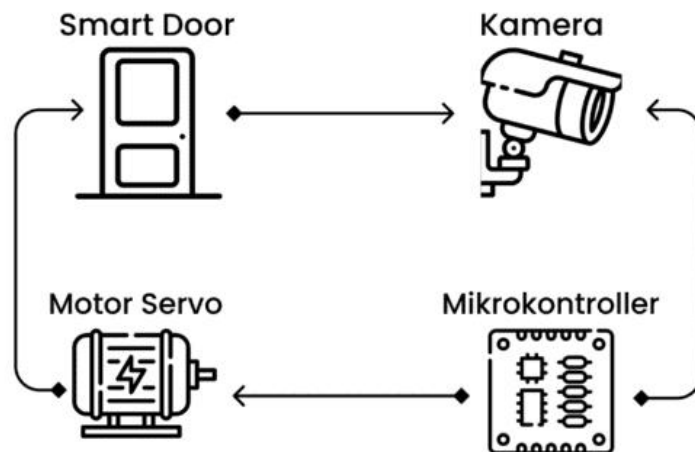


Gambar 3.2 alur mekanisme penelitian

1. Identifikasi kebutuhan sistem diperlukan untuk menyiapkan segala hal yang mendukung kinerja dan perancangan sistem. Pada tahap ini akan dilakukan analisis masalah yang akan dipecahkan oleh sistem, menentukan kebutuhan fungsional dan non-fungsional, dan mendokumentasikan kebutuhan ini dalam sebuah dokumen spesifikasi kebutuhan sistem. Dokumen ini akan digunakan sebagai pedoman selama pengembangan sistem untuk memastikan bahwa sistem memenuhi kebutuhan pengguna nantinya.
2. Mempelajari penggunaan mikrokontroller ESP32 dengan Algoritma *Haar Cascade* dari beberapa sumber yang *actual* juga *factual* seperti jurnal, buku, artikel, dan juga termasuk penelitian sebelumnya yang berkaitan dengan penelitian ini.
3. Pengumpulan data dilakukan dengan mengumpulkan berbagai foto kucing yang menjadi target dalam penelitian ini
4. *Design* sistem yang dibangun akan menggunakan *software Proteus*. Dimana tujuan penggunaan *software* adalah untuk merancang, simulasi, dan verifikasi rangkaian elektronik. Proteus juga memiliki fitur desain *PCB (Printed Circuit Board)* yang memungkinkan pengguna untuk merancang dan membuat layout rangkaian elektronik pada papan sirkuit cetak.
5. Sistem akan dibangun menggunakan Algoritma *Haar Cascade* yang dan menggunakan *software* Arduino IDE.
6. Pengujian sistem secara keseluruhan bertujuan untuk mengetahui apakah *Smart Door* dapat berfungsi dengan baik atau belum. Jika ditemukan adanya kesalahan akan dilakukan perbaikan terlebih dahulu. Dan jika sudah tidak ada lagi masalah maka akan lanjut ke tahap berikutnya.
7. Pada tahap ini sistem yang sudah selesai dirancang dan dibangun dapat digunakan oleh user sesuai dengan fungsinya.

3.3.2 Arsitektur Sistem

Arsitektur sistem menggambarkan alur, proses dan interaksi dari masing-masing sistem komponen sistem yang dibangun. Perancangan smart door diuraikan melalui gambar berikut :



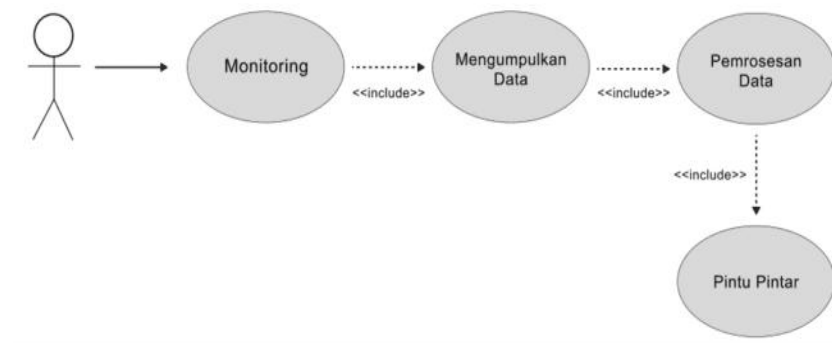
Gambar 3.3 Arsitektur Sistem

1. Sistem yang dibangun diletakkan tepat di bawah daun pintu rumah.
2. Kamera digunakan sebagai sumber inputan untuk mendeteksi hewan kucing secara *real-time*.
3. Sistem ini didesain berbasis mikrokontroler ESP32 sebagai inti dari arsitektur. Mikrokontroler ESP32 bertanggung jawab atas pengendalian operasi keseluruhan sistem, termasuk pendeteksian wajah kucing, klasifikasi, pengendalian pintu, dan interaksi dengan pengguna. Sistem memiliki akurasi yang tinggi jika kondisi lingkungan memiliki pencahayaan yang mendukung.
4. Setelah wajah kucing terdeteksi dan terdeteksi valid, maka sistem memberikan sinyal kepada motor servo untuk membuka pintu. Jika wajah kucing tidak dikenali, sistem akan memberikan perintah kepada motor servo untuk menutup pintu dan mencegah akses yang tidak sah.

3.3.2.1 Use Case Diagram

Use case diagram adalah sebuah diagram yang digunakan untuk menggambarkan interaksi antara pengguna dan sistem yang dibuat. Use case diagram menggambarkan peran pengguna (user) dalam menggunakan sistem

dan fungsionalitas yang disediakan oleh sistem tersebut. Gambar 3.3 menunjukkan contoh use case diagram yang digunakan untuk menggambarkan hubungan antara pengguna dan sistem dalam suatu konteks.

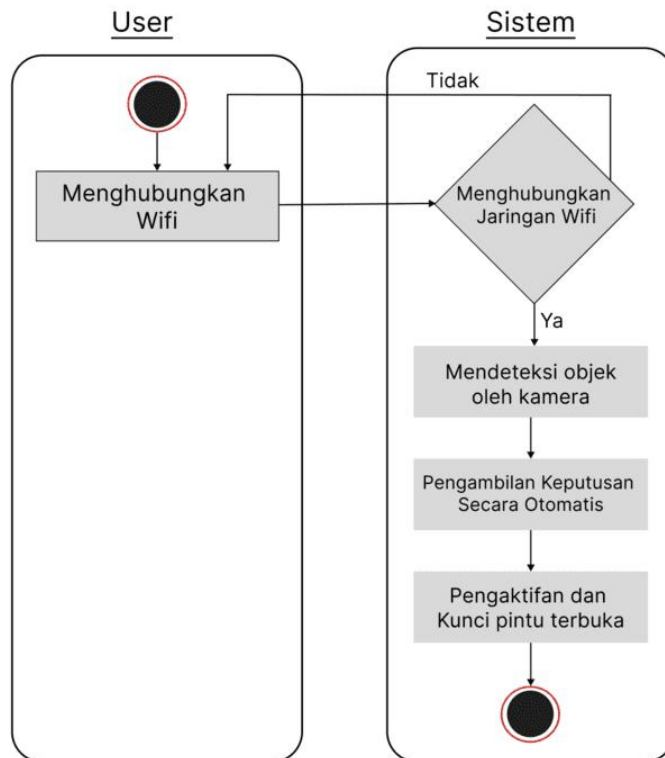


Gambar 3.4. *Use Case Diagram*

Pada gambar diagram use case diatas dapat diketahui bahwa pengguna dapat memonitoring pemberian pupuk yang sudah di proses dari pengumpulan data sensor.

3.3.2.2 Activity Diagram

Activity diagram adalah salah satu jenis diagram dalam UML (Unified Modeling Language) yang digunakan untuk menggambarkan aliran aktivitas atau urutan kejadian dalam suatu proses atau aktivitas. Diagram ini menggambarkan serangkaian tindakan atau aktivitas yang terjadi dalam sistem atau proses tertentu. Activity diagram memberikan pemahaman yang lebih baik tentang alur kerja sistem, langkah-langkah yang terlibat dalam suatu proses, keputusan yang diambil, dan bagaimana aktivitas-aktivitas tersebut saling terkait.



Gambar 3.5 Activity Diagram

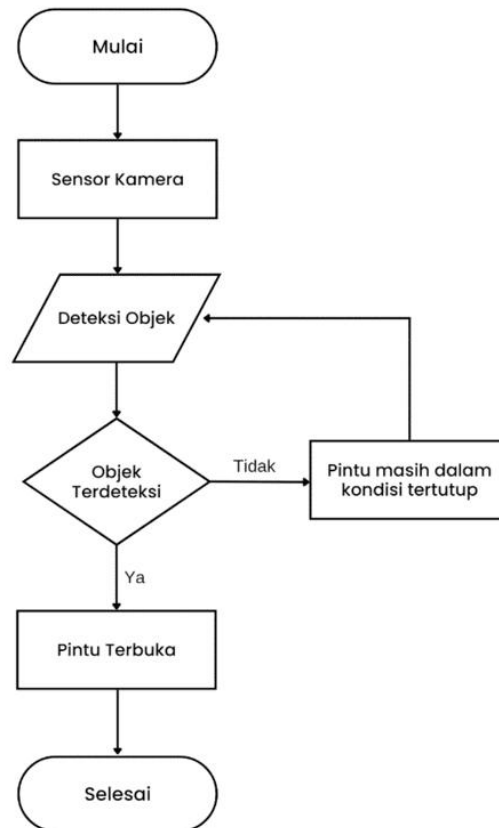
Dalam activity diagram di atas, terdapat langkah-langkah interaksi antara pengguna dengan sistem. Proses dimulai dengan user menghubungkan sistem ke WiFi dan sumber power. Setelah terhubung, sistem akan melanjutkan dengan pendeteksian objek oleh sensor untuk. Nilai yang terbaca kemudian akan diolah menggunakan metode Haar Cascade. Setelah objek ditentukan, dan objek terdeteksi benar maka sistem akan membuka door lock (alat untuk mengunci dan membuka pintu) dengan bantuan relay.

3.4 Flowchart

Diagram alir atau yang biasa disebut dengan flowchart digunakan untuk memberi gambaran terhadap jalannya suatu sistem dengan memberikan arah dari suatu proses satu ke proses lainnya. Flowchart umumnya menggunakan simbol-simbol grafis untuk menggambarkan setiap tahap dalam urutan yang ditentukan, serta menunjukkan bagaimana informasi atau data diproses atau dihubungkan antara satu tahap dengan tahap lainnya.

3.4.1 Flowchart Sistem

Gambar 5 berikut ini akan menggambarkan alur kerja dari smart door yang dibuat :

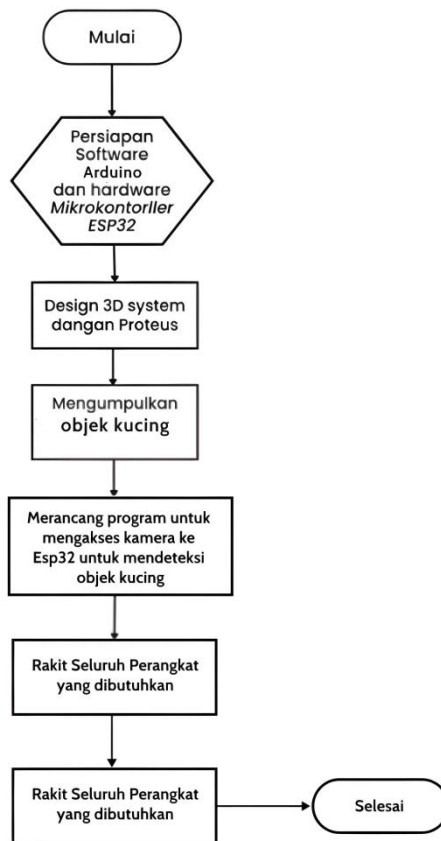


Gambar 3.6 *Flowchart Sistem*

Pada gambar diatas, dijelaskan bahwa alur sistem dimulai dari kamera (inputan) sebagai sensor pada sistem ini. Kemudian selanjutnya kamera akan mendeteksi objek. Jika objek terdeteksi salah atau tidak terdeteksi sama sekali maka pintu akan terus dalam kondisi tertutup. Namun jika objek terdeteksi benar atau dapat dikenali dengan benar maka, smart door akan membuka pintu secara otomatis dengan durasi yang telah ditentukan.

3.4.2 Flowchart Perancangan *Smart Door*

Berikut adalah flowchart proses *development* sistem smart door yang dapat dilihat pada gambar dibawah ini :



Gambar 3.7 *Flowchart Pembuatan Sistem*

3.5 Perancangan Sistem

Penelitian ini melibatkan perancangan hardware yang terintegrasi dan dibangun menggunakan komponen elektronika dan membentuk sebuah sistem elektronika yang sederhana. Tujuan dari desain hardware ini adalah untuk mempermudah pengoperasian oleh pengguna. Sistem hardware ini mencakup beberapa komponen, seperti main board dan sensor, yang diintegrasikan bersama untuk mengumpulkan data yang diperlukan.

3.5.1 Peralatan dan Bahan

A. Peralatan

Dalam perancangan sistem pintu pintar pendeteksi wajah kucing menggunakan metode haar cascade pada mikrokontroler Esp32-Cam, terdapat beberapa peralatan yang digunakan yaitu :

- Solder
- Timah
- Gunting

- Tang potong
- Obeng
- Multimeter

B. Bahan

Dalam perancangan sistem pintu pintar pendeteksi wajah kucing menggunakan metode haar cascade pada mikrokontroller Esp32-Cam, terdapat beberapa bahan yang digunakan yaitu :

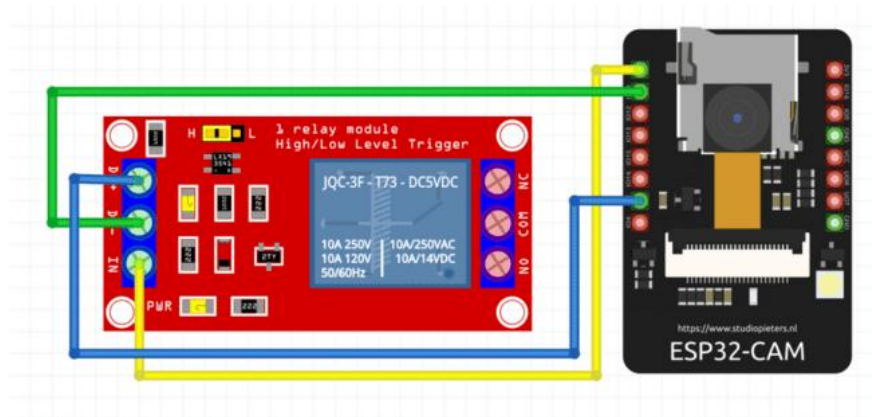
- Esp32-Cam
- Relay 5v 1 channel
- Adaptor 12v 2a
- Solenoid Door Lock
- Kabel
- Micro Usb
- Jumper
- Stepdown

3.6 Perancangan Mikrokontroller Esp32-Cam

Skema perancangan mikrokontroller Esp32-Cam adalah dengan menghubungkan Esp32-Cam kebel relay, kemudian dari modul relay dilanjutkan ke *solenoid door lock*. Untuk menghubungkan Esp32-Cam ke relay dibutuhkan 3 kabel sesuai dengan yang tercantum pada Tabel 3.1 dan Gambar 3.8.

Tabel 3.1 koneksi Esp-32 Cam ke Relay

Pin Esp-32 Cam	Relay
5v	IN
GND	DC-
IO2	DC+



Gambar 3.8 Koneksi Esp-32 Cam ke Relay

3.6.1 Koneksi *Stepdown*

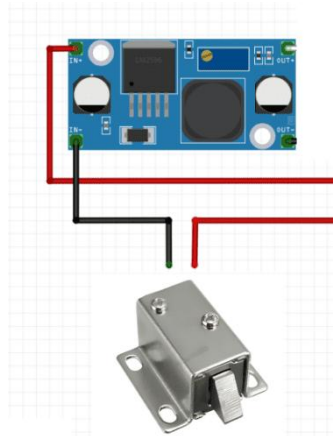
Alat ini menggunakan modul *stepdown* LM2596 DC-DC yang melibatkan penggunaan regulator tegangan buck seperti LM2596, adalah untuk menurunkan tegangan masukan yang diberikan oleh adaptor 12V menjadi tingkat tegangan yang sesuai dengan kebutuhan komponen-komponen sistem. Dalam hal ini, khususnya, tegangan tersebut akan diturunkan menjadi tingkat yang dapat diakomodasi oleh mikrokontroler ESP32-Cam dan komponen lainnya.

a. Koneksi *Stepdown* ke *Solenoid door lock*

Berikut adalah tabel koneksi *Stepdown* ke *Solenoid Door Lock*

Tabel 3.2 Koneksi *Stepdown* ke *Solenoid door lock*

<i>Stepdown</i>	<i>Solenoid Door Lock</i>
In -	(+)



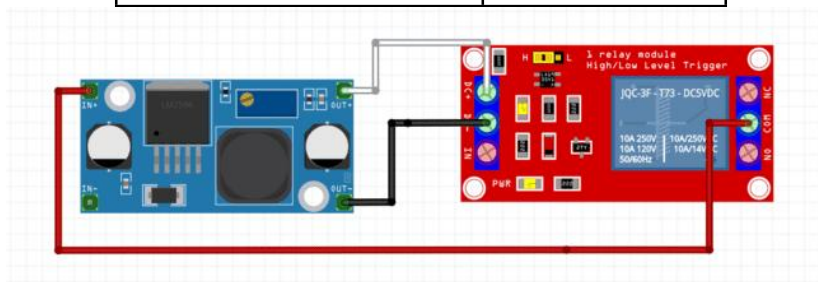
Gambar 3.9 Koneksi *Stepdown* ke *Selenoid door lock*

b. Koneksi *Stepdown* ke *relay*

Berikut adalah koneksi antara *Stepdown* dengan *relay*

Tabel 3.3 Koneksi *Stepdown* ke *relay*

<i>Stepdown</i>	<i>Relay</i>
Out +	In
Out -	DC -



Gambar 3.10 Koneksi *Stepdown* ke *Relay*

3.6.2 Koneksi Selenoid Door Lock

Solenoid door lock dalam sistem penelitian ini memiliki peran sentral sebagai motor elektromagnetik yang bertugas mengendalikan mekanisme penguncian dan pembukaan pintu. Solenoide menciptakan medan elektromagnetik yang memungkinkannya menarik bagian mekanis, mengunci pintu dengan aman. Fungsionalitas elektronik solenoide ini memungkinkan pengintegrasian dengan mikrokontroler seperti ESP-32 CAM, memungkinkan

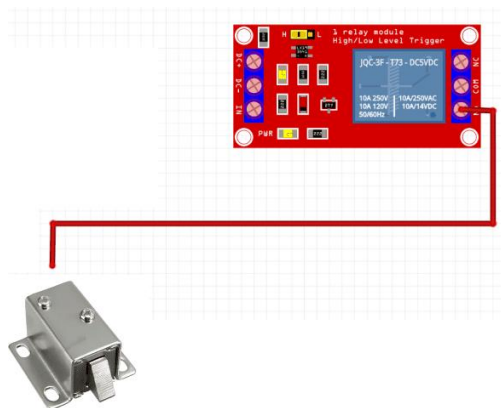
kontrol pintu dari jarak jauh atau otomatisasi yang terkait dengan deteksi wajah kucing.

a. Koneksi *Solenoid door lock* ke *Relay*

Berikut adalah Koneksi antara *Solenoid door lock* dan *Relay*

Tabel 3.4 Koneksi *Stepdown* ke *Solenoid door lock*

<i>Solenoid Door Lock</i>	<i>Relay</i>
(-)	On



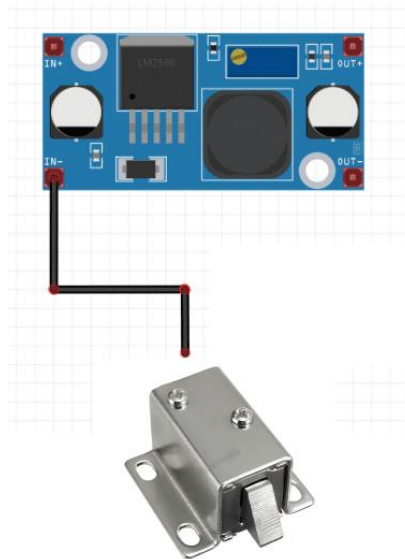
Gambar 3.11 Koneksi *Solenoid Door Lock* ke *Relay*

b. Koneksi *Solenoid door lock* ke *Stepdown*

Berikut adalah Koneksi *Solenoid door lock* dan *Stepdown*

Tabel 3.5 Koneksi *Solenoid Door Lock* ke *Stepdown*

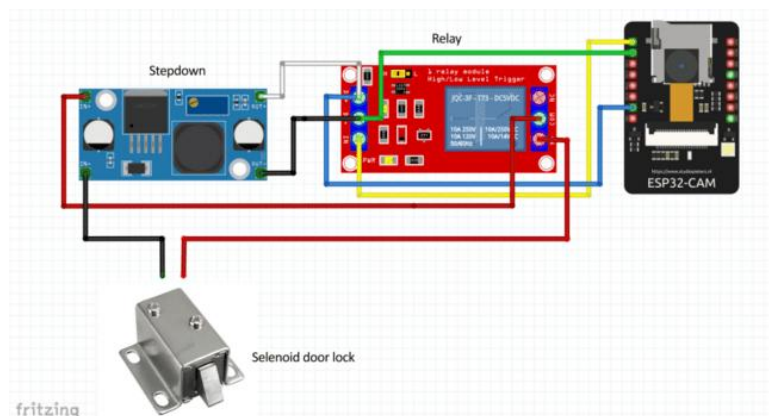
<i>Solenoid Door Lock</i>	<i>Stepdown</i>
(+)	In -



Gambar 3.12 Koneksi *Selenoid Door Lock* ke *Stepdown*

3.6.2 Keseluruhan Sistem

Dalam sistem ini, Esp32-Cam berperan sebagai komponen utama yang bertanggung jawab untuk memproses input dan output dalam sistem. Esp32-Cam berfungsi sebagai unit pemrosesan yang menerima input dari sensor kamera yang menjadi sensor utama, dan kemudian menghasilkan output untuk mengontrol komponen lain, seperti relay dan *selenoid door lock*.



Gambar 3.13 Rangkaian keseluruhan sistem

Keterangan :

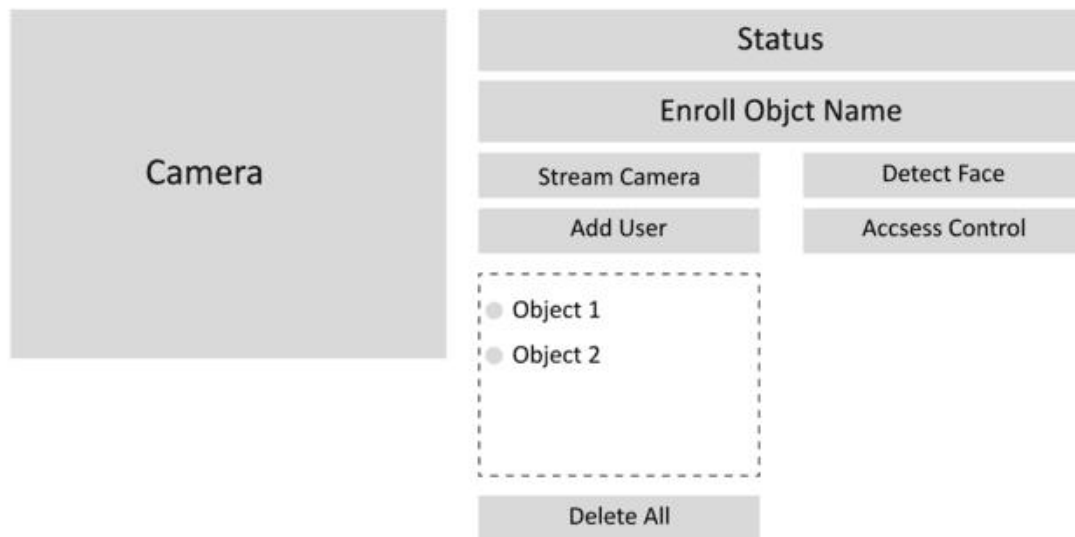
- Pin 5v (Esp32-Cam) - Pin Input (Relay)

- Pin Grund (Esp32-Cam) - Pin ((DC -) Relay)
- Pin Input Output 2 (Esp32-Cam) - Pin ((DC +) Relay)
- Pin Input (-) (Stepdown) - Pin (+) (Solenoid Door Lock)
- Pin Output (+) (Stepdown) - Pin Input (Relay)
- Pin Output (-) - Pin DC (-) (Relay)
- Pin(-) (Solenoid Door Lock) - Pin NO (Relay)
- Pin (+) (Solenoid Door Lock) - Pin Input (-) (Stepdown)

Sesuai pada gambar 3.9, semua komponen saling terhubung dan berinteraksi satu sama lain. Setiap alat memiliki peran yang berbeda untuk dapat menjalankan sistem.

3.7 Perancangan Sistem Perangkat Lunak (*Software*)

Pengembangan skema perancangan software melibatkan langkah-langkah lebih lanjut dalam merinci dan memperluas fungsionalitas sistem. Langkah berikutnya mencakup desain dan implementasi program yang menjalankan operasi pada sistem web, dengan fokus pada integrasi yang efisien antara perangkat ESP32-Cam dan website. Selanjutnya, pengembangan mencakup aspek pengoptimalan kinerja, keamanan, dan antarmuka pengguna agar sistem dapat berfungsi dengan lebih lancar dan sesuai dengan kebutuhan pengguna. Desain perangkat lunak dapat dilihat pada Gambar 3.14.



Gambar 3.14 Design antarmuka website

Dalam konteks Gambar 3.14, tampilan tersebut menggambarkan antarmuka sistem kontrol yang dirancang untuk mengoperasikan sistem pendeteksian wajah kucing. Dari tampilan tersebut, pengguna diberikan kemampuan untuk menambahkan (enroll) banyak objek yang diinginkan, dan memberikan nama pada setiap objek yang dimasukkan. Selain itu, antarmuka ini memberikan opsi bagi pengguna untuk menghapus objek yang tidak diperlukan, memberikan fleksibilitas dan pengelolaan yang optimal. Pengguna juga dapat melihat jumlah objek yang telah dimasukkan ke dalam sistem. Hal ini memberikan pemahaman yang jelas kepada pengguna tentang sejauh mana sistem telah terintegrasi dengan objek atau wajah kucing yang telah didaftarkan.

BAB IV

IMPLEMENTASI DAN PENGUJIAN

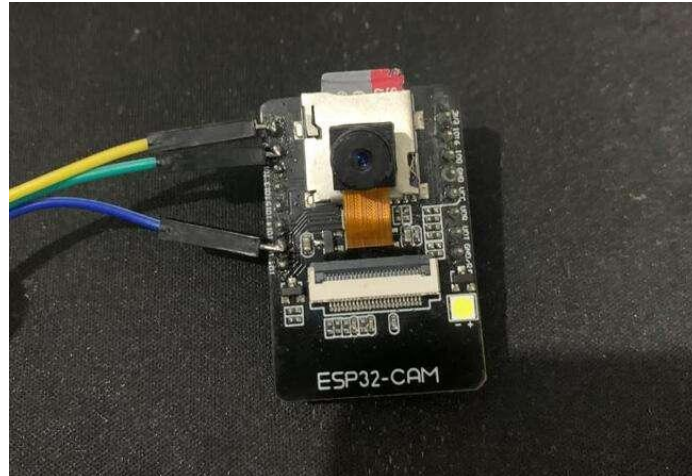
Pada tahap ini, sistem akan diuji untuk mengevaluasi kinerja pendeteksian wajah kucing secara real-time menggunakan metode Haar Cascade pada mikrokontroler ESP32-Cam. Uji coba ini bertujuan untuk mengoptimalkan parameter deteksi dan memastikan respons yang cepat terhadap kehadiran wajah kucing. Data hasil pendeteksian tersebut akan dapat dipantau secara langsung melalui sebuah antarmuka website yang telah dirancang sebelumnya untuk memberikan pemantauan real-time terhadap deteksi wajah kucing.

4.1 Implementasi Sistem

Tata letak rangkaian dalam implementasi sistem pintu pintar pendeteksi wajah kucing secara real-time menggunakan metode Haar Cascade pada mikrokontroler ESP32-Cam merupakan langkah awal dalam menerapkan deteksi wajah kucing secara otomatis. Uji coba sistem dilakukan di lokasi dengan lingkungan yang representatif, memastikan respons dan akurasi deteksi pada berbagai kondisi. Pada tahap pengujian ini, rangkaian alat akan dipasang pada pintu utama rumah, memperhitungkan variasi kondisi pencahayaan dan posisi kucing untuk mengevaluasi performa sistem secara menyeluruh.

4.1.1 Implementasi Perancangan ESP32-Cam

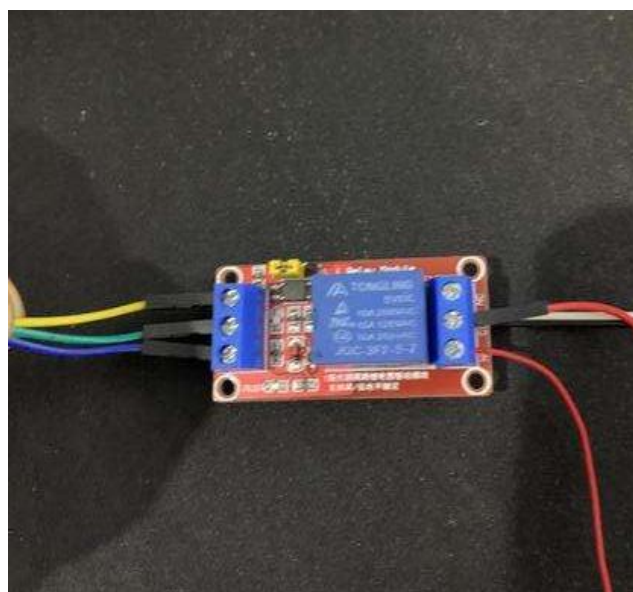
Sensor kamera yang terletak pada ESP32-Cam memegang peran kunci dalam implementasi sistem pintu pintar pendeteksi wajah kucing menggunakan mikrokontroler Esp32-Cam. Sensor ini berfungsi untuk mendeteksi setiap kali ada objek yang lewat. Hasil deteksi wajah dapat digunakan untuk mengontrol *Solenoid Door Lock*. Kondisi *Solenoid Door Lock* akan selalu terkunci (*false*) jika tidak adanya objek yang terdeteksi, sebaliknya jika objek terdeteksi benar (*true*) maka ESP32-Cam berkomunikasi dengan sistem pintu pintar untuk membuka atau menutup pintu berdasarkan deteksi wajah kucing.



Gambar 4.1 Esp32 - Cam

4.1.2 Implementasi Perancangan Relay

Dalam implementasi perancangan pintu pintar pendeteksi wajah kucing menggunakan mikrokontroler ESP32-Cam, Relay low-high trigger 5v 1 *channel* memiliki peran penting sebagai saklar elektronik yang dikendalikan oleh ESP32-Cam untuk mengontrol *solenoid door lock*. Relay dihubungkan dengan ESP32-Cam melalui pin atau port yang telah ditentukan. Mikrokontroler mengirimkan sinyal kontrol ke Relay, mengatur statusnya antara ON (aktif) dan OFF (non-aktif) berdasarkan hasil deteksi objek.



Gambar 4.2 Relay 5v Low High Trigger

4.1.3 Implementasi Perancangan *Solenoid Door Lock*

Solenoid berfungsi sebagai motor elektronik yang merespons instruksi dari mikrokontroler ESP-32 CAM untuk membuka atau menutup pintu sesuai dengan hasil deteksi wajah kucing. Dengan beroperasi pada tegangan 12V, solenoid dirancang untuk menghasilkan gaya elektromagnetik untuk menggerakkan bagian mekanis yang membuka atau menutup pintu sesuai instruksi yang diberikan oleh relay 5v. Keamanan sistem ditingkatkan dengan penggunaan solenoid yang dikontrol secara elektronik, memastikan bahwa akses hanya diberikan ketika wajah kucing yang diotorisasi terdeteksi.



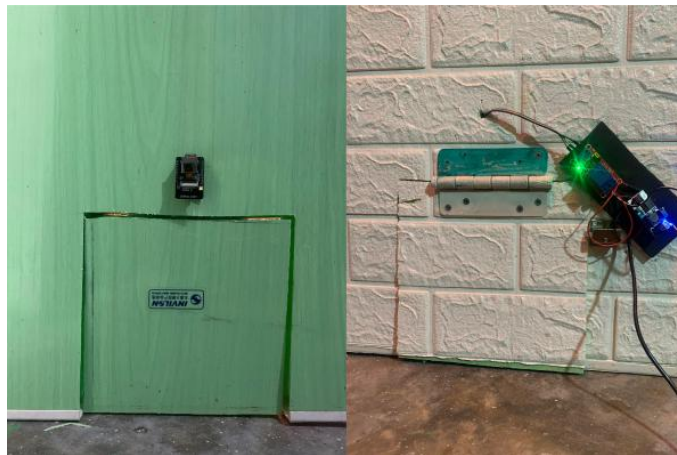
Gambar 4.3 *Solenoid Door Lock*

4.1.4 Implementasi Keseluruhan Sistem

Implementasi keseluruhan perangkat dalam penelitian pintu pintar pendeteksi wajah kucing mengintegrasikan beberapa komponen utama untuk menciptakan sistem yang responsif dan aman. Mikrokontroler utama yang digunakan adalah ESP-32 CAM, yang bertanggung jawab atas fungsi deteksi wajah menggunakan metode Haar Cascade. ESP32-Cam berinteraksi dengan solenoid door lock 12V untuk mengendalikan mekanisme pembukaan dan penutupan pintu otomatis.

Relay low-high trigger 5V digunakan sebagai saklar elektronik yang dikendalikan oleh ESP-32 CAM untuk mengatur daya listrik menuju solenoid door lock. Relay berperan dalam mengamankan dan mengelola daya listrik

yang digunakan oleh solenoide. Solenoide door lock 12V, sebagai motor elektronik utama, membuka atau menutup pintu berdasarkan instruksi dari ESP-32 CAM. Pengoperasian solenoide dikendalikan secara elektronik untuk memastikan bahwa akses hanya diberikan ketika wajah kucing yang diotorisasi terdeteksi. Seluruh sistem dapat dipantau melalui sebuah antarmuka web yang telah dirancang sebelumnya, memungkinkan pengguna untuk memantau status pintu pintar dan proses pembukaan atau penutupan pintu secara real-time.



Gambar 4.4 Tampak depan dan belakang implementasi sistem

4.2 Pengujian Sistem

Pengujian sistem dilakukan untuk memastikan bahwa sistem yang telah dikembangkan sesuai dengan spesifikasi yang telah ditetapkan dan dapat berfungsi dengan optimal sesuai dengan tujuan yang diinginkan. Pada pengujian ini, penulis secara langsung melakukan uji coba dengan menggunakan tiga objek kucing kampung atau kucing lokal sebagai sampel. Pengujian dilakukan pada kucing yang memiliki warna bulu yang berbeda dan tidak memiliki kemiripan antara kucing satu dengan kucing lainnya. Pendekatan ini bertujuan untuk menguji kehandalan sistem dalam mengenali dan memproses gambar dari objek yang umum dijumpai di lingkungan sekitar.



Gambar 4.5 Kucing belang putih, kucing orange, kucing belang hitam

Dengan menggunakan tiga objek kucing sebagai sampel, penelitian ini dapat memberikan gambaran yang lebih komprehensif tentang kinerja sistem dalam menghadapi variasi karakteristik dan postur kucing kampung yang berbeda. Hasil dari pengujian ini diharapkan dapat memberikan informasi yang lebih akurat mengenai kemampuan sistem dalam mendeteksi dan mengenali wajah kucing kampung secara efektif.

4.2.1 Pengujian Fungsional Website

Pengujian fungsional aplikasi dilakukan untuk melihat apakah seluruh fungsi baik berupa inputan ataupun terjadi secara langsung berjalan dengan baik sesuai dengan seharusnya. Berikut hasil pengujian tombol menu pada website :



Gambar 4.6 Halaman Utama Website.

Tabel 4.1 Pengujian fitur pada website

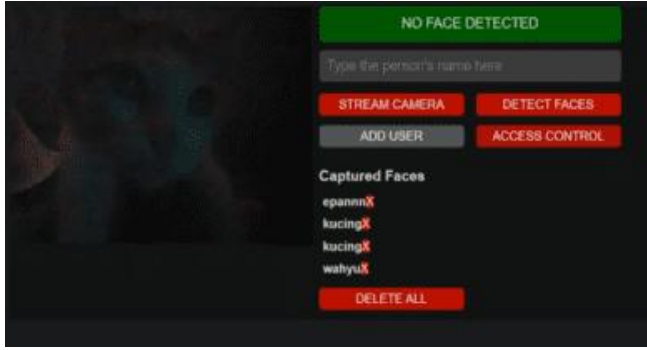
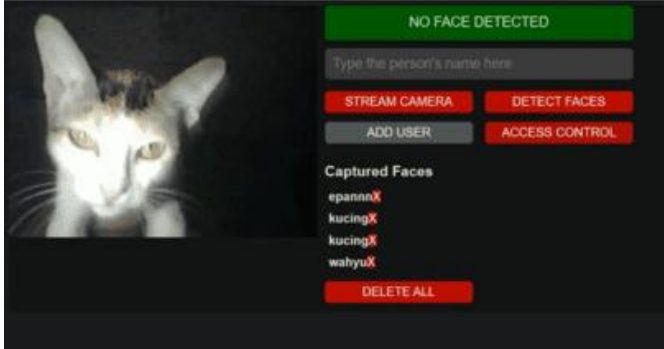
<i>Input</i>	<i>Output yang diharapkan</i>	Hasil Pengamatan	Keterangan
Menekan Tombol Stream Kamera	Sistem dapat melakukan streaming	Sistem melakukan streaming	[✓] Berhasil [] Gagal
Menekan Tombol Detect faces	Sistem dapat melakukan detecting faces	Sistem melakukan detecting face	[✓] Berhasil [] Gagal
Menekan tombol Add user	Sistem dapat menambahkan objek baru	Sistem menambahkan objek baru	[✓] Berhasil [] Gagal
Menekan tombol Access control	Sistem dapat menjalankan kamera untuk mendeteksi objek	Sistem menjalankan kamera untuk mendeteksi objek	[✓] Berhasil [] Gagal

4.2.2 Pengujian Intensitas Cahaya

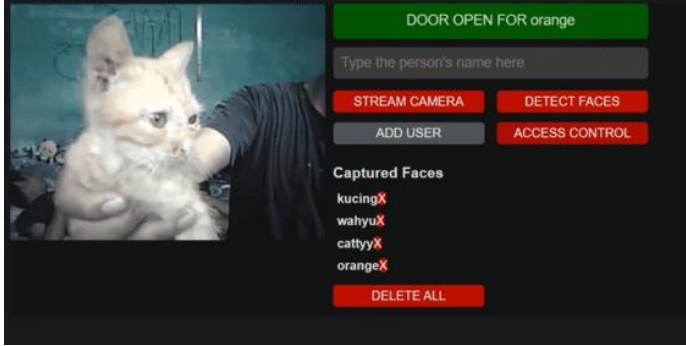
Pengujian dengan intensitas cahaya dilakukan untuk mengetahui respon dan akurasi sistem dalam pendeteksian objek pada kondisi cahaya tertentu dalam kondisi penggunaan kamera sehari-hari.

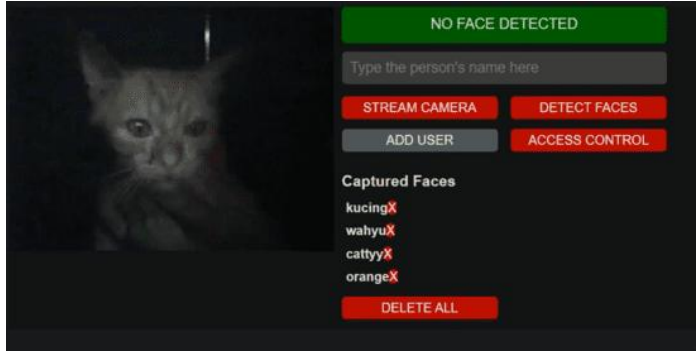

Tabel 4.6 Pengujian intensitas cahaya kucing 1

<i>Input</i>	Gambar	Keterangan
Kondisi siang hari, cahaya cukup		[✓] Berhasil [] Gagal


Kondisi malam hari, cahaya kurang		<input type="checkbox"/> Berhasil <input checked="" type="checkbox"/> Gagal
Kondisi malam hari, dengan flash		<input type="checkbox"/> Berhasil <input checked="" type="checkbox"/> Gagal

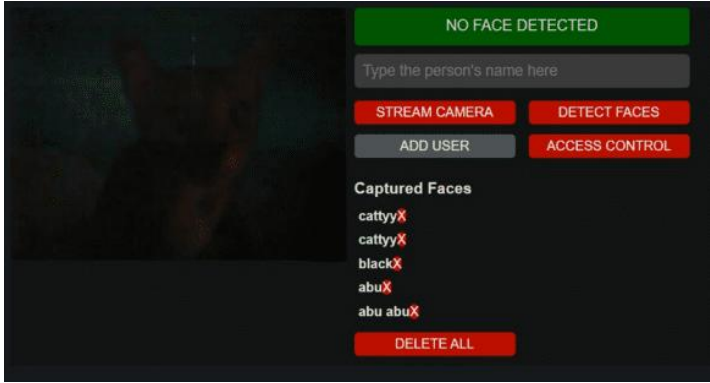

Tabel 4.3 Pengujian intensitas Cahaya kucing 2

<i>Input</i>	Gambar	Keterangan
Kondisi siang hari, cahaya cukup		<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Gagal

Kondisi malam hari, cahaya kurang		<input type="checkbox"/> Berhasil <input checked="" type="checkbox"/> Gagal
Kondisi malam hari, dengan flash		<input type="checkbox"/> Berhasil <input checked="" type="checkbox"/> Gagal

Tabel 4.4 Pengujian intensitas cahaya kucing 3

Input	Gambar	Keterangan
Kondisi siang hari, cahaya cukup		<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Gagal

Kondisi malam hari, cahaya kurang		<input type="checkbox"/> Berhasil <input checked="" type="checkbox"/> Gagal
Kondisi malam hari, dengan flash		<input type="checkbox"/> Berhasil <input checked="" type="checkbox"/> Gagal

4.2.3 Pengujian Sudut dan Jarak

Pengujian Pengujian sudut dan jarak dilakukan untuk mengetahui respon sistem terhadap perubahan situasi pendeteksian objek pada jarak dan sudut tertentu. Pengujian akurasi ini dilakukan dengan interval jarak 5-10cm, 10-20cm, 20-30cm, 30-40cm, dan >40cm serta pada sudut dengan rotasi 60°, 45°, dan 30°.

Tabel 4.5 Pengujian sudut dan jarak kucing 1.

Jarak (cm)	Sudut	Output	Keterangan	Hasil Uji
10cm	0°	Informasi Muncul	Objek terdeteksi	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Gagal
	30° ke-kanan	Informasi Muncul	Objek terdeteksi	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Gagal
	30° ke-kiri	Informasi Muncul	Objek terdeteksi	<input checked="" type="checkbox"/> Berhasil

				<input type="checkbox"/> Gagal
10cm - 20cm	0°	Informasi Muncul	Objek terdeteksi	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Gagal
	30° ke-kanan	Informasi Muncul	Objek terdeteksi	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Gagal
	30° ke-kiri	Informasi Muncul	Objek terdeteksi	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Gagal
20cm - 30cm	0°	Informasi Muncul	Objek terdeteksi	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Gagal
	30° ke-kanan	Informasi Muncul	Objek terdeteksi	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Gagal
	30° ke-kiri	Informasi Muncul	Objek terdeteksi	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Gagal
30cm - 40cm	0°	Informasi Muncul	Objek tidak terdeteksi	<input type="checkbox"/> Berhasil <input checked="" type="checkbox"/> Gagal
	30° ke-kanan	Informasi Muncul	Objek tidak terdeteksi	<input type="checkbox"/> Berhasil <input checked="" type="checkbox"/> Gagal
	30° ke-kiri	Informasi Muncul	Objek tidak terdeteksi	<input type="checkbox"/> Berhasil <input checked="" type="checkbox"/> Gagal
> 40cm	0°	Informasi Muncul	Objek tidak terdeteksi	<input type="checkbox"/> Berhasil <input checked="" type="checkbox"/> Gagal
	30° ke-kanan	Informasi Muncul	Objek tidak terdeteksi	<input type="checkbox"/> Berhasil <input checked="" type="checkbox"/> Gagal
	30° ke-kiri	Informasi Muncul	Objek tidak terdeteksi	<input type="checkbox"/> Berhasil <input checked="" type="checkbox"/> Gagal

Tabel 4.6 Pengujian sudut dan jarak kucing 2

Jarak (cm)	Sudut	Output	Keterangan	Hasil Uji
10cm	0°	Informasi Muncul	Objek terdeteksi	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Gagal
	30° ke-kanan	Informasi Muncul	Objek terdeteksi	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Gagal
	30° ke-kiri	Informasi Muncul	Objek terdeteksi	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Gagal
10cm - 20cm	0°	Informasi Muncul	Objek terdeteksi	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Gagal
	30° ke-kanan	Informasi Muncul	Objek terdeteksi	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Gagal
	30° ke-kiri	Informasi Muncul	Objek terdeteksi	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Gagal
20cm - 30cm	0°	Informasi Muncul	Objek terdeteksi	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Gagal
	30° ke-kanan	Informasi Muncul	Objek terdeteksi	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Gagal
	30° ke-kiri	Informasi Muncul	Objek terdeteksi	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Gagal
30cm - 40cm	0°	Informasi Muncul	Objek tidak terdeteksi	<input type="checkbox"/> Berhasil <input checked="" type="checkbox"/> Gagal
	30° ke-kanan	Informasi Muncul	Objek tidak terdeteksi	<input type="checkbox"/> Berhasil <input checked="" type="checkbox"/> Gagal
	30° ke-kiri	Informasi Muncul	Objek tidak terdeteksi	<input type="checkbox"/> Berhasil <input checked="" type="checkbox"/> Gagal

> 40cm	0°	Informasi Muncul	Objek tidak terdeteksi	<input type="checkbox"/> Berhasil <input checked="" type="checkbox"/> Gagal
	30° ke-kanan	Informasi Muncul	Objek tidak terdeteksi	<input type="checkbox"/> Berhasil <input checked="" type="checkbox"/> Gagal
	30° ke-kiri	Informasi Muncul	Objek tidak terdeteksi	<input type="checkbox"/> Berhasil <input checked="" type="checkbox"/> Gagal

Tabel 4.7 Pengujian sudut dan jarak kucing 3.

Jarak (cm)	Sudut	Output	Keterangan	Hasil Uji
10cm	0°	Informasi Muncul	Objek terdeteksi	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Gagal
	30° ke-kanan	Informasi Muncul	Objek terdeteksi	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Gagal
	30° ke-kiri	Informasi Muncul	Objek terdeteksi	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Gagal
10cm - 20cm	0°	Informasi Muncul	Objek terdeteksi	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Gagal
	30° ke-kanan	Informasi Muncul	Objek terdeteksi	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Gagal
	30° ke-kiri	Informasi Muncul	Objek terdeteksi	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Gagal
20cm - 30cm	0°	Informasi Muncul	Objek terdeteksi	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Gagal
	30° ke-kanan	Informasi Muncul	Objek terdeteksi	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Gagal

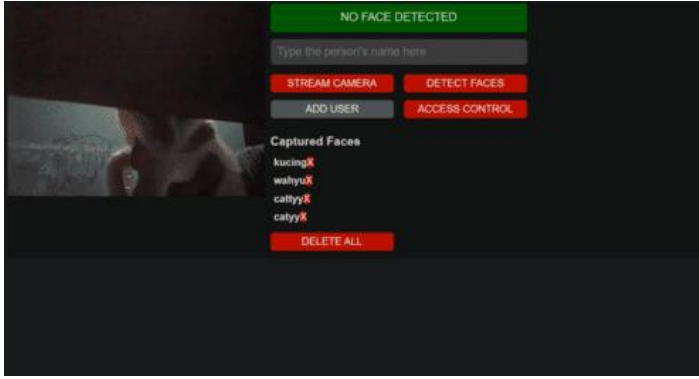
	30° ke-kiri	Informasi Muncul	Objek terdeteksi	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Gagal
30cm - 40cm	0°	Informasi Muncul	Objek tidak terdeteksi	<input type="checkbox"/> Berhasil <input checked="" type="checkbox"/> Gagal
	30° ke-kanan	Informasi Muncul	Objek tidak terdeteksi	<input type="checkbox"/> Berhasil <input checked="" type="checkbox"/> Gagal
	30° ke-kiri	Informasi Muncul	Objek tidak terdeteksi	<input type="checkbox"/> Berhasil <input checked="" type="checkbox"/> Gagal
> 40cm	0°	Informasi Muncul	Objek tidak terdeteksi	<input type="checkbox"/> Berhasil <input checked="" type="checkbox"/> Gagal
	30° ke-kanan	Informasi Muncul	Objek tidak terdeteksi	<input type="checkbox"/> Berhasil <input checked="" type="checkbox"/> Gagal
	30° ke-kiri	Informasi Muncul	Objek tidak terdeteksi	<input type="checkbox"/> Berhasil <input checked="" type="checkbox"/> Gagal

4.2.4 Pengujian Oklusi

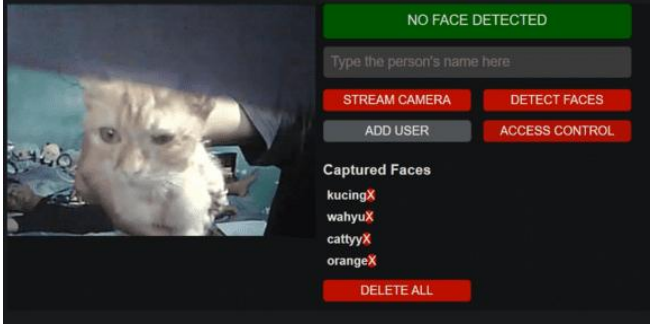
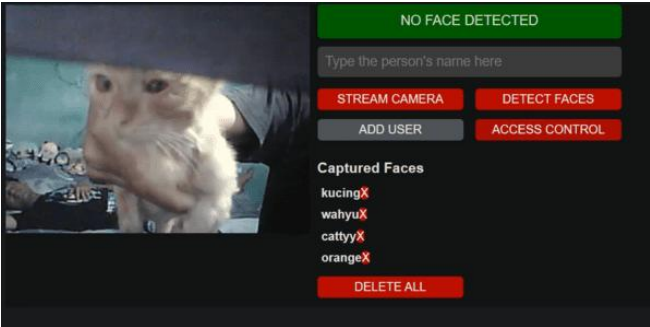
Pengujian oklusi adalah pengujian pendeteksian objek apabila objek terhalang sesuatu ataupun terpotong saat pendeteksian oleh kamera. Pengujian ini dilakukan dengan cara menutup objek dengan interval 10%, 30%, 50%, dan 70%.

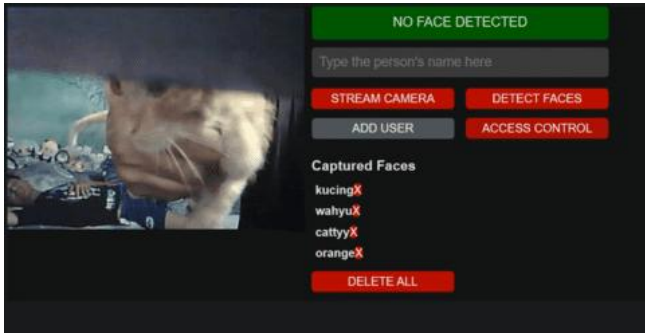
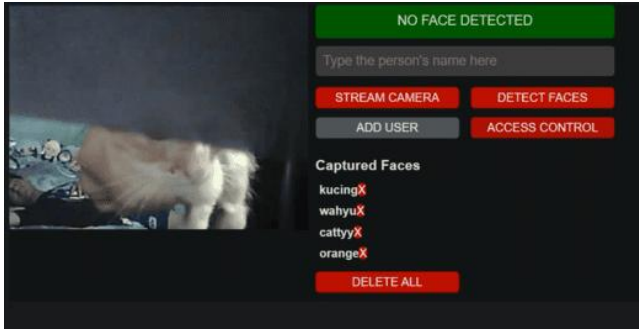
Tabel 4.8 Pengujian oklusi kucing 1

Kondisi Objek	Gambar	Keterangan
Tertutup 10%		[] Berhasil [✓] Gagal
Tertutup 30%		[] Berhasil [✓] Gagal
Tertutup 50%		[] Berhasil [✓] Gagal

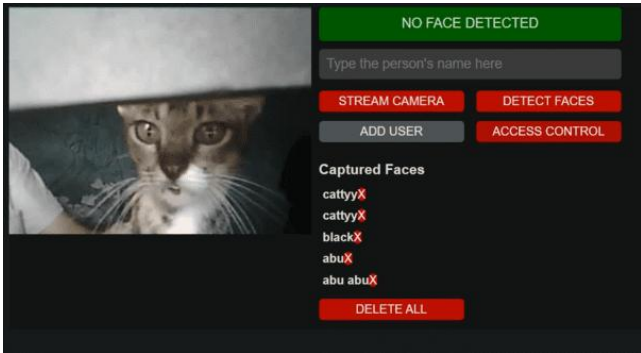
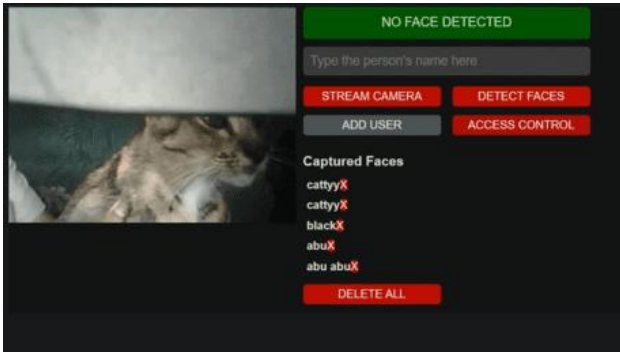
Tertutup 70%		<input type="checkbox"/> Berhasil <input checked="" type="checkbox"/> Gagal
--------------	--	--

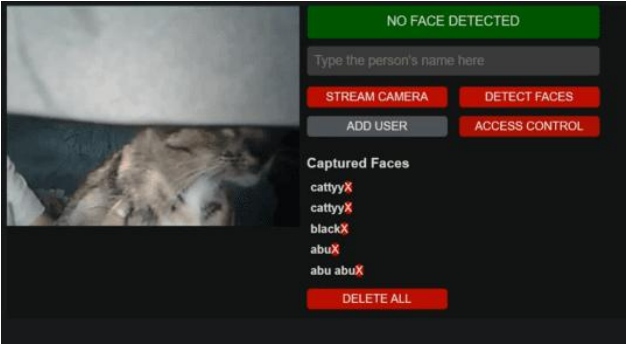
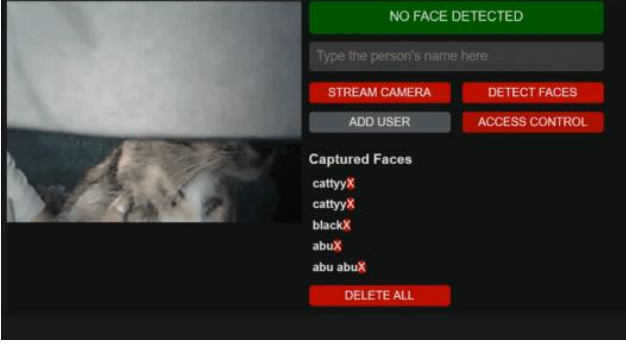
Tabel 4.9 Pengujian oklusi kucing 2.

Kondisi Objek	Gambar	Keterangan
Tertutup 10%		<input type="checkbox"/> Berhasil <input checked="" type="checkbox"/> Gagal
Tertutup 30%		<input type="checkbox"/> Berhasil <input checked="" type="checkbox"/> Gagal

Tertutup 50%		<input type="checkbox"/> Berhasil <input checked="" type="checkbox"/> Gagal
Tertutup 70%		<input type="checkbox"/> Berhasil <input checked="" type="checkbox"/> Gagal

Tabel 4.10 Pengujian oklusi kucing 3

Kondisi Objek	Gambar	Keterangan
Tertutup 10%		<input type="checkbox"/> Berhasil <input checked="" type="checkbox"/> Gagal
Tertutup 30%		<input type="checkbox"/> Berhasil <input checked="" type="checkbox"/> Gagal

Tertutup 50%		[] Berhasil [✓] Gagal
Tertutup 70%		

4.2.5 Pengujian Algoritma Haar Cascade

Pengujian Haar Cascade merupakan contoh dari pengimplementasian computer vision yang dapat digunakan untuk berbagai tujuan seperti pengenalan citra, dan pengenalan objek. Pola cascade ini dirancang sebagai serangkaian tahap atau level classifier yang diurutkan secara hierarkis. Pada setiap tahap, citra masukan diuji, dan jika berhasil melewati uji di tahap tertentu, citra tersebut dianggap mengandung objek yang dicari. Jika tidak, citra tersebut langsung dibuang, memungkinkan penolakan cepat terhadap citra yang tidak relevan.

4.2.6 Pengujian Filter Haar-like

Filter Haar-like adalah satu set filter yang digunakan untuk mendeteksi fitur tertentu pada citra. Filter ini terdiri dari dua kotak (hitam dan putih) yang disusun secara berurutan. Ukuran filter ($w \times h$) di mana, w adalah lebar filter dan h adalah tinggi filter. Filter Haar-like dengan $w=2$ dan $h=1$ dapat direpresentasikan sebagai berikut:

$$\text{Filter} = \begin{bmatrix} -1 & 1 \end{bmatrix}$$

Perhitungan nilai Haar-like $H(x,y)$ pada suatu posisi (x,y) dalam citra dilakukan dengan mengalikan elemen filter dengan nilai piksel pada posisi yang sesuai dalam citra dan kemudian menjumlahkannya:

$$H(x,y) = \sum_{i=1}^w \sum_{j=1}^w \text{Filter}(i, j) \times \text{Citra}(x+i, y+j)$$

Proses ini diulangi untuk setiap posisi dalam citra, menghasilkan citra yang telah difilter, dengan nilai Haar-like yang mencerminkan perbedaan antara bagian hitam dan putih filter di setiap posisi.

4.2.7 Integral Image

Integral Image adalah representasi citra yang memungkinkan perhitungan nilai total intensitas piksel dalam suatu area tertentu dengan cepat. Integral Image (II) dihitung dengan menjumlahkan nilai piksel pada setiap titik dalam citra dari kiri atas ke kanan bawah. Misalkan citra dengan piksel intensitas (x,y) . Proses perhitungan Integral Image dimulai dari piksel paling kiri atas dan diiterasikan secara berurutan ke kanan dan ke bawah. Rumus perhitungan nilai Integral Image pada suatu piksel (x,y) adalah sebagai berikut:

$$II(x,y) = I(x,y) + II(x-1,y) + II(x,y-1) - II(x-1,y-1)$$

Ketika objek (kucing) terdeteksi oleh kamera dengan menggunakan Haar Cascade, proses yang terjadi pada Integral Image melibatkan penggunaan classifier cascade. Setiap classifier pada cascade menguji nilai Haar-like pada area tertentu dalam Integral Image. Jika nilai Haar-like pada suatu area dalam Integral Image ($II(x,y)$) melebihi ambang batas tertentu, maka objek dianggap terdeteksi. Proses ini diulangi untuk setiap tahap classifier dalam cascade untuk memastikan keakuratan dan kehandalan deteksi objek.

4.2.8 Implementasi User Interface Website

Pada sistem pintu pintar pendeteksi wajah kucing ini menyediakan antarmuka pengguna yang dapat dimonitoring dan mengakses kontrol melalui sebuah website. Website ini menampilkan tiga informasi penting yaitu dapat

menambahkan objek (*add user*), mendeteksi objek (*face detecting*), dan menghapus objek (*deleting user*). Pengguna dapat mengklik tombol akses kontrol untuk memulai mendeteksi objek, Gambar tampilan website dapat dilihat pada gambar 4.3.

BAB V

PENUTUP

5.1 Kesimpulan

Dari hasil penelitian dan pengujian yang telah penulis laksanakan, dapat diambil beberapa kesimpulan yaitu :

1. Walaupun kurang akurat sistem pintu pintar yang menggunakan metode Algoritma Haar Cascade terbukti efektif dalam mendeteksi ciri-ciri khusus pada wajah kucing dan memberikan hasil identifikasi yang cepat.
2. Hasil pengujian intensitas cahaya memperoleh kesimpulan bahwa, kamera hanya dapat mendeteksi secara maksimal jika objek mendapat pencahayaan yang cukup.
3. Dari hasil pengujian jarak dan sudut dapat disimpulkan bahwa objek hanya dapat terdeteksi dalam radius 10cm sampai dengan 30cm, dengan sudut 0° dan 30° (ke-kanan dan ke-kiri) oleh kamera.
4. Hasil pengujian oklusi juga menunjukkan hasil dimana kamera tidak dapat mendeteksi objek jika terdapat benda yang menghalangi objek tersebut.

5.2 Saran

Untuk pengembangan pada penelitian selanjutnya, berikut adalah saran yang dapat dipertimbangkan oleh penulis, yaitu :

1. Untuk menghasilkan hasil yang lebih akurat disarankan Menggunakan algoritma SSD (Single Multishot Detector) dan algoritma MTCNN (Multi-task Cascaded Convolutional Networks).
2. Kamera yang tertanam pada Esp32-Cam hanya beresolusi 2mp, Disarankan untuk mempertimbangkan penggunaan kamera dengan resolusi yang lebih tinggi seperti 5mp atau diatas 5mp, guna meningkatkan detail dalam pengenalan wajah kucing.

3. Mengoptimalkan pencahayaan lingkungan tempat sistem diimplementasikan. Penelitian lebih lanjut terkait cara mengelola kondisi pencahayaan yang berbeda dapat meningkatkan daya adaptasi sistem terhadap lingkungan sekitarnya.
4. Memasukan banyak dataset dari objek kucing dari berbagai variasi posisi dan sudut yang mungkin terjadi dalam penggunaan sehari-hari agar mendapat hasil yang lebih akurat

DAFTAR PUSTAKA

- Abidin Suhepy. (2018). *“Deteksi Wajah Menggunakan Metode Haar Cascade Classifier Berbasis Webcam Pada Matlab”* e-ISSN 2656-0143. Universitas Negeri Ujung Padang
- al, N. D. (2020). *Analisis Fitur Haar Menggunakan Algoritma Haar-Like Feature Pada Citra Kendaraan Bermotor*. Universitas Trunojoyo Madura.
- Andika Willy, Maulana Rizal, Utaminingrum Fitri. (2020). *“Implementasi Sistem Otomatisasi Pintu Dengan Face Recognition Menggunakan Metode Haar-Cascade Dan Local Binary Pattern Pada Raspberry Pi”*. Universitas Brawijaya
- Baatrix, R. A. (2021). *Monitoring Laju Kendaraan Berbasis IP Camera Menggunakan Metode Haar Sitanggang*. Universitas Sumatera Utara.
- Fikri, M. (2022). *Perancangan Sistem Penyiraman Dan Pemupukan Otomatis (Smart Garden) Berbasis Iot (Internet Of Things) Menggunakan Nodemcu Esp8266*. Media Komunikasi Ilmiah Dibidan Teknik Vol 23 No 1.
- K, H. (2019). *PENERAPAN TRAINER INTERFACING MIKROKONTROLER DAN INTERNET OF THINGS BERBASIS ESP32*. Redaksi Journal.
- Kiran, K. d. (2020). *Implementation of Online Python Learning Platform Using Raspberry Pi*. Procedia Computer Science, vol. 171. Hal. 504-509. doi: 10.1016/j.procs.2020.04.066.
- Mouhammad, C. S.-R. (2019). *BLE Indoor Localization based on Improved RSSI and Trilateration. Proceedings of the International Japan-Africa Conference on Electronics, Communications and Computations (Vols. JAC-ECC 2019, 17–21)*.
- Nemati, J. D. (2020). *IoT Connectivity Technologies and Applications: A Survey*. IEEE Xplore.
- Nugroho, S. A. (2022). *Rancang Bangun Sistem Deteksi Label Kardus Berbasis Model Kecerdasan Buatan YOLO dan EasyOCR serta ESP32-CAM*. UNESA.

Prafanto Anton, Budiman Edy. (2021). *“PENDETEKSI KEHADIRAN MENGGUNAKAN ESP32 UNTUK SISTEM PENGUNCI PINTU OTOMATIS”*. Universitas Mulawarman

Sulistiyo Wahyu, Suyanto Budi, Hestinationsih Idhawati, Mardiyono, Sukanto. (2014). *“Rancang Bangun Prototipe Aplikasi Pengenalan Wajah untuk Sistem Absensi Alternatif dengan Metode Haar Like Feature dan Eigenface”*. Polsri. Politeknik Negeri Semarang.

Wills, H. F. (2020). *IoT Security, Privacy, Safety and Ethics*. Springer Nature Switzerland AG .

Y, G. (2020). *IMPLEMENTASI ALGORITMA FACE DETECTION PADA SISTEM CCTV DI LABORATORIUM TELEKOMUNIKASI*. Polsri.