

PERANCANGAN SISTEM DETEKSI DAN KLASIFIKASI INTRUSI SATWA LIAR MENGGUNAKAN ESP32-CAM BERBASIS MACHINE LEARNING TENSORFLOW JS

Riansyah Akhwan¹, Firahmi Rizky²

Email: ^{1,2}Universitas Muhammadiyah Sumatera Utara, Indonesia

Email: riansyahakwan001@gmail.com

ABSTRAK

Meningkatnya aktivitas ekowisata dan bumi perkemahan di kawasan konservasi Sumatera Utara memicu tingginya intensitas Konflik Manusia dan Satwa Liar (KMSL), khususnya melibatkan spesies monyet ekor panjang (*Macaca fascicularis*) dan babi hutan (*Sus scrofa*). Metode pengamanan konvensional seperti patroli jagawana bersifat reaktif dan berisiko tinggi, sementara kamera jebak (*camera trap*) komersial tidak mengirimkan data secara *real-time*. Penelitian ini bertujuan untuk merancang dan membangun sistem deteksi dan klasifikasi intrusi satwa liar berbasis *Internet of Things* (IoT) menggunakan mikrokontroler ESP32-CAM dan *machine learning* TensorFlow.js sebagai sistem peringatan dini (*early warning system*). Penelitian menggunakan metode *Research and Development* (R&D). Pada sisi *edge*, perangkat ESP32-CAM diintegrasikan dengan sensor PIR HC-SR501 untuk mendeteksi pergerakan dari perubahan radiasi termal dari tubuh satwa dan membangkitkan mikrokontroler. Transmisi data gambar dikirimkan menggunakan protokol MQTT menuju server backend berbasis Node.js/Express melalui jaringan WiFi ke server untuk disimpan dalam database lokal serta divisualisasikan pada dashboard monitoring berbasis web secara *real-time*. Proses klasifikasi citra dilakukan secara dinamis menggunakan penggabungan model umum COCO-SSD mendeteksi segala jenis Satwa Liar dengan manajemen memori menggunakan *Worker Thread Pool*. Hasil pengujian fungsional menunjukkan seluruh komponen bekerja dengan baik. Pengujian akurasi *machine learning* mencapai rata-rata di atas 85% untuk model umum dan 88% - 97% untuk model kustom. Latensi pengiriman dan inferensi data dari lapangan hingga tampil di dashboard web *React* (TanStack Start) rata-rata berkisar antara 2,6 hingga 4,4 detik. Analisis konsumsi daya menunjukkan mode *deep sleep* (~6 μ A) mampu menjaga daya tahan baterai 3500 mAh hingga lebih dari 60 hari. Sistem ini terbukti efektif, hemat energi, dan responsif sebagai solusi pemantauan perimeter pariwisata alam.

Kata kunci: *Internet of Things*, Deteksi Satwa Liar, ESP-32 CAM, PIR HC-SR501, Machine Learning, TensorFlow JS, MQTT.

ABSTRACT

*The increasing activities of ecotourism and camping grounds in conservation areas of North Sumatra have led to a higher incidence of Human–Wildlife Conflict (HWC), particularly involving long-tailed macaques (*Macaca fascicularis*) and wild boars (*Sus scrofa*). Conventional security measures, such as ranger patrols, are reactive and pose significant risks, while commercial camera traps are unable to transmit data in real time. This study aims to design and develop an *Internet of Things* (IoT)-based wildlife intrusion detection and classification system using an ESP32-CAM microcontroller and TensorFlow.js machine learning as an early warning system. The research employed the *Research and Development* (R&D) method. On the edge device, the ESP32-CAM was integrated with a PIR HC-SR501 sensor to detect movement based on changes*

in thermal radiation emitted by wildlife and to activate the microcontroller. Captured image data were transmitted using the MQTT protocol to a Node.js/Express backend server over a Wi-Fi network for storage in a local database and real-time visualization through a web-based monitoring dashboard. Image classification was performed dynamically by combining the general COCO-SSD model for detecting various types of wildlife with memory management using a Worker Thread Pool. Functional testing demonstrated that all system components operated successfully. The machine learning evaluation achieved an average accuracy of over 85% for the general model and between 88% and 97% for the custom model. The average latency for data transmission and inference, from field acquisition to display on the React (TanStack Start) web dashboard, ranged from 2.6 to 4.4 seconds. Power consumption analysis showed that the deep sleep mode (approximately 6 μ A) enabled a 3500 mAh battery to operate for more than 60 days. These results indicate that the proposed system is effective, energy-efficient, and responsive as a perimeter monitoring solution for nature tourism areas.

Keywords: Internet of Things (IoT), Wildlife Intrusion Detection, ESP32-CAM, TensorFlow.js, Machine Learning, PIR HC-SR501, MQTT.

PENDAHULUAN

Indonesia, sebagai negara kepulauan tropis terbesar di dunia, memiliki kekayaan biodiversitas yang luar biasa, menjadikannya salah satu dari 17 negara megadiversitas di dunia. Hutan hujan tropis Indonesia bukan hanya paru-paru dunia, tetapi juga habitat bagi ribuan spesies satwa liar endemik. Seiring dengan pertumbuhan ekonomi dan pergeseran tren gaya hidup pascapandemi, sektor pariwisata berbasis alam (*nature-based tourism*) atau ekowisata mengalami lonjakan popularitas yang signifikan. Fenomena ini ditandai dengan menjamurnya area perkemahan (*camping ground*), fasilitas glamorous camping (*glamping*), dan jalur pendakian baru yang sering kali berbatasan langsung dengan, atau bahkan berada di dalam, kawasan konservasi seperti Taman Nasional, Cagar Alam, dan Hutan Lindung. Meskipun memberikan dampak ekonomi positif bagi masyarakat lokal dan meningkatkan kesadaran lingkungan, ekspansi aktivitas manusia ke dalam habitat alami satwa liar membawa konsekuensi ekologis yang serius, yakni meningkatnya intensitas Konflik Manusia dan Satwa Liar atau *Human-Wildlife Conflict* (HWC). (Rifaie & Arida, 2022)

Interaksi negatif antara manusia dan satwa liar di area rekreasi alam telah menjadi isu manajemen konservasi yang mendesak. Di Pulau Jawa dan Sumatera, dua pulau dengan kepadatan penduduk tertinggi sekaligus kantong habitat satwa yang terfragmentasi, konflik ini sering kali melibatkan spesies primata seperti Monyet Ekor Panjang (*Macaca fascicularis*) dan Beruk (*Macaca nemestrina*), serta mamalia darat seperti Babi Hutan (*Sus scrofa*). *Macaca fascicularis*, khususnya, telah menunjukkan perubahan perilaku yang adaptif namun merugikan akibat terbiasa dengan kehadiran manusia (*human habituation*). Ketersediaan sisa makanan logistik pendaki atau sampah organik di area perkemahan menjadi atraktan utama yang memicu perilaku raiding atau penjarahan. Laporan dari berbagai daerah, termasuk Sumatera Utara, mencatat insiden di mana kelompok monyet tidak hanya mencuri makanan tetapi juga menunjukkan perilaku agresif, seperti menyeringai, mengejar, hingga menggigit pengunjung yang mencoba mempertahankan barang bawaan mereka. (Wulandari et al., 2023)

Kasus serangan satwa di area wisata tidak hanya menimbulkan kerugian materil berupa kerusakan tenda dan hilangnya logistik, tetapi juga

risiko kesehatan dan keselamatan jiwa. Gigitan primata liar berpotensi menularkan penyakit zoonosis seperti Rabies, Herpes B Virus, dan Tetanus. Selain itu, intrusi hewan nokturnal seperti babi hutan menimbulkan ancaman fisik yang lebih serius bagi para pekemah yang sedang tidur, mengingat ukuran tubuh dan kekuatan fisik spesies ini yang mampu merobohkan tenda. Data dari Balai Besar Konservasi Sumber Daya Alam (BBKSDA) sering kali menunjukkan lonjakan laporan konflik pada musim-musim tertentu dimana sumber pakan alami di hutan berkurang, memaksa satwa untuk memperluas daerah jelajah (*home range*) mereka hingga ke zona aktivitas manusia. (Kudus et al., 2023)

Dalam menghadapi tantangan ini, metode pengamanan yang diterapkan di sebagian besar area perkemahan di Indonesia masih sangat konvensional dan reaktif. Pengelola umumnya mengandalkan patroli fisik oleh jagawana atau petugas keamanan yang dilakukan secara berkala. Namun, pendekatan ini memiliki keterbatasan fundamental. Pertama, keterbatasan sumber daya manusia membuat pengawasan 24 jam di seluruh perimeter perkemahan yang luas menjadi mustahil. Kedua, faktor kelelahan manusia (*human fatigue*) dan kondisi lingkungan (gelap, hujan, kabut) menurunkan efektivitas deteksi dini. Ketiga, risiko keselamatan bagi petugas itu sendiri sangat tinggi ketika harus berhadapan langsung dengan satwa liar yang agresif tanpa adanya sistem peringatan dini (*early warning system*).

Teknologi pemantauan satwa liar yang ada saat ini, seperti kamera jebak (*camera trap*), telah lama digunakan dalam penelitian ekologi. Namun, mayoritas kamera trap yang beredar di pasaran didesain untuk pengambilan data pasif. Data gambar atau video disimpan dalam kartu memori lokal (SD Card) dan hanya dapat diakses setelah petugas mengambil kamera tersebut secara fisik, yang mungkin dilakukan mingguan atau bulanan. Karakteristik offline ini menjadikan kamera trap tidak relevan untuk tujuan mitigasi konflik keamanan yang membutuhkan respons waktu nyata (*real-time*). Di sisi lain, sistem pengawasan berbasis kamera CCTV (*Closed-Circuit Television*) konvensional menawarkan pemantauan *real-time*, tetapi implementasinya di area perkemahan terhambat oleh kendala infrastruktur. CCTV membutuhkan suplai daya listrik yang stabil dan besar (biasanya AC 220V), serta jaringan kabel yang rumit, yang sering kali tidak tersedia di lokasi perkemahan yang terpencil atau off-grid. (Xin & Rosely, 2025)

Evolusi pesat dalam bidang *Internet of Things* (IoT) dan *Embedded Systems* menawarkan peluang untuk menjembatani kesenjangan teknologi ini. Paradigma IoT memungkinkan pengembangan perangkat cerdas yang hemat energi, berukuran kecil, dan mampu berkomunikasi secara nirkabel. Salah satu komponen yang menjadi primadona dalam pengembangan visi komputer pada perangkat tepi (*edge computer vision*) adalah ESP32-CAM. Modul ini merupakan mikrokontroler berbasis ESP32 yang mengintegrasikan prosesor dual-core 32-bit yang bertenaga, konektivitas *Wi-Fi* dan *Bluetooth*, serta antarmuka kamera (mendukung sensor OV2640) dengan harga yang sangat terjangkau. Kemampuan ESP32-CAM untuk menangkap citra visual, melakukan pemrosesan awal (seperti kompresi), dan mengirimkannya melalui protokol internet menjadikannya kandidat ideal untuk sistem keamanan nirkabel di area terpencil yang hanya mengandalkan sumber daya baterai atau panel surya. (Aryunita et al., 2024)

Sistem deteksi intrusi satwa liar ini bekerja secara hibrida dengan mengintegrasikan sensor PIR HC-SR501 sebagai pemicu utama yang mendeteksi perubahan radiasi inframerah dari panas tubuh satwa, sehingga membangunkan ESP32-CAM dari mode deep sleep untuk menangkap gambar/video secara real-time. Sensor PIR memindai perimeter perkemahan; saat mendeteksi gerakan biologis (monyet/babi hutan), kirim sinyal *HIGH* ke GPIO ESP32-CAM dalam <1 detik. Kamera OV2640 aktif, kompresi JPEG, dan upload gambar ke Arduino IoT Cloud via *WiFi* dalam 3-5 detik hemat baterai 95% karena hanya aktif saat ada ancaman. *Image Widget* di dashboard Arduino Cloud update otomatis + push notifikasi ke HP pengelola; data historis tersimpan untuk analisis pola intrusi malam hari. (Bazir et al., 2022)

Pada tahun 2024, Arduino, sebagai salah satu pelopor ekosistem perangkat keras terbuka, memperbarui platform cloud mereka, Arduino IoT Cloud, dengan fitur-fitur visualisasi yang lebih canggih, termasuk *Image Widget*. Fitur ini memungkinkan pengguna untuk menampilkan gambar yang diperbarui secara dinamis dari mikrokontroler langsung di dasbor web atau aplikasi seluler tanpa memerlukan konfigurasi server web yang rumit atau port forwarding. Integrasi native antara perangkat keras ESP32 (yang didukung penuh oleh ekosistem Arduino) dan Arduino Cloud menawarkan kestabilan koneksi, keamanan data melalui enkripsi SSL/TLS, dan kemudahan penggunaan yang belum banyak dieksplorasi dalam konteks mitigasi konflik satwa liar. Pemanfaatan *Image Widget* ini berpotensi memberikan solusi visualisasi yang bersih dan profesional bagi pengelola perkemahan: sebuah layar dasbor tunggal yang menampilkan status keamanan perimeter dan foto real-time saat intrusi terdeteksi. (Arduino, 2024).

METODE PENELITIAN

Penelitian ini menggunakan metode *Research and Development* (R&D) yang bertujuan untuk merancang dan mengembangkan sistem deteksi intrusi satwa liar berbasis *Internet of Things* (IoT) dan kecerdasan buatan. Produk yang dikembangkan berupa prototipe *Beastly Vision* yang mengintegrasikan ESP32-CAM, sensor PIR HC-SR501, komunikasi MQTT, server Node.js, TensorFlow.js dengan model COCO-SSD, PostgreSQL, dan dashboard web. Sistem dirancang untuk mendeteksi pergerakan, mengambil gambar, mengenali objek, menentukan tingkat ancaman, serta menampilkan hasil deteksi secara real-time.

Tahapan Penelitian

Penelitian dilaksanakan melalui beberapa tahapan, yaitu:

1. Identifikasi Masalah

Tahap ini dilakukan untuk mengidentifikasi kebutuhan sistem deteksi satwa liar pada area perkemahan serta permasalahan yang muncul dalam proses pemantauan.

2. Studi Literatur

Peneliti mempelajari teori dan penelitian terdahulu yang berkaitan dengan IoT, ESP32-CAM, sensor PIR, MQTT, pengolahan citra, kecerdasan buatan, TensorFlow.js, COCO-SSD, dan sistem monitoring berbasis web.

3. Perancangan Sistem

Perancangan meliputi perangkat keras, perangkat lunak, komunikasi data, dan dashboard monitoring. ESP32-CAM digunakan untuk mengambil gambar ketika sensor PIR mendeteksi pergerakan. Data gambar kemudian dikirim melalui MQTT ke server untuk diproses oleh model AI.

4. Pengembangan dan Perakitan Sistem

Pada tahap ini dilakukan perakitan ESP32-CAM dan sensor PIR, pemrograman perangkat, pembangunan server, integrasi model AI, basis data, serta pengembangan dashboard Beastly Vision.

5. Integrasi Sistem

Seluruh komponen dihubungkan agar dapat bekerja sebagai satu sistem. ESP32-CAM menangkap gambar, MQTT mengirimkan data ke server, TensorFlow.js melakukan deteksi objek, hasil disimpan ke PostgreSQL, dan informasi ditampilkan pada dashboard.

6. Pengujian Sistem

Pengujian dilakukan untuk mengetahui apakah sistem bekerja sesuai dengan rancangan. Pengujian meliputi:

- a) sensitivitas sensor PIR;
- b) fungsi kamera dan flash;
- c) koneksi Wi-Fi dan MQTT;
- d) deteksi objek menggunakan AI;
- e) penyimpanan dan tampilan hasil pada dashboard; dan
- f) kendali perangkat dari dashboard.

7. Evaluasi dan Perbaikan

Hasil pengujian dianalisis untuk menemukan kesalahan atau kekurangan sistem. Perbaikan dilakukan pada bagian yang belum sesuai, seperti pengaturan *confidence threshold*, klasifikasi tingkat ancaman, dan penambahan fitur *push notification*.

8. Pengujian Akhir

Setelah perbaikan dilakukan, sistem diuji kembali untuk memastikan bahwa fungsi deteksi objek, klasifikasi ancaman, notifikasi, komunikasi MQTT, dan perangkat ESP32-CAM dapat berjalan dengan baik dan stabil.

Perancangan Sistem

Sistem Beastly Vision menggunakan arsitektur IoT yang terdiri atas perangkat ESP32-CAM, sensor PIR, jaringan Wi-Fi, MQTT Broker, server backend, model AI, basis data, dan dashboard web.

Sensor PIR berfungsi mendeteksi pergerakan. Ketika pergerakan terdeteksi, ESP32-CAM mengambil gambar menggunakan kamera OV2640. Gambar kemudian dikirim melalui jaringan Wi-Fi menggunakan protokol MQTT menuju server. Server memproses gambar menggunakan TensorFlow.js dan model COCO-SSD untuk mengenali objek serta menentukan tingkat ancaman. Hasil deteksi disimpan dalam PostgreSQL dan ditampilkan pada dashboard Beastly Vision.

Dashboard juga menyediakan fitur pemantauan dan kendali jarak jauh, seperti pengambilan gambar secara manual, pengaturan sensor PIR, pengendalian flash, dan *reboot* perangkat.

Perangkat Penelitian

Perangkat yang digunakan terdiri atas perangkat keras dan perangkat lunak. Perangkat keras utama meliputi ESP32-CAM dan kamera OV2640, sensor PIR HC-SR501, baterai Li-ion, buck converter, project box, dan perangkat pendukung perakitan.

Perangkat lunak yang digunakan meliputi Arduino IDE, ESP32-CAM Board Package, Visual Studio Code, TensorFlow.js, COCO-SSD, Node.js/TypeScript, PostgreSQL, MQTT Broker EMQX, serta React.js, Vite, dan Tailwind CSS untuk pengembangan dashboard.

Teknik Pengujian

Pengujian dilakukan menggunakan *Black Box Testing* dan *Performance Testing*. Black Box Testing digunakan untuk mengetahui apakah setiap fungsi sistem berjalan sesuai dengan yang dirancang, sedangkan Performance Testing digunakan untuk mengetahui respons dan kestabilan sistem.

Pengujian dilakukan melalui beberapa skenario, yaitu pengujian sensor PIR pada berbagai jarak, pengujian kamera dan flash, pengujian koneksi Wi-Fi dan MQTT, pengujian deteksi objek AI, pengujian kendali jarak jauh, serta pengujian notifikasi ancaman.

Sensor PIR juga dikalibrasi saat perangkat pertama kali dinyalakan agar pembacaan sensor lebih stabil. Setelah sistem bekerja, hasil pengujian dianalisis untuk mengetahui kesalahan deteksi atau gangguan sistem.

Evaluasi dan Pengujian Akhir

Apabila ditemukan kesalahan selama pengujian, dilakukan perbaikan terhadap sistem. Perbaikan yang dilakukan meliputi penyesuaian *confidence threshold*, perbaikan klasifikasi tingkat ancaman, serta penambahan fitur *push notification* untuk ancaman tingkat tinggi.

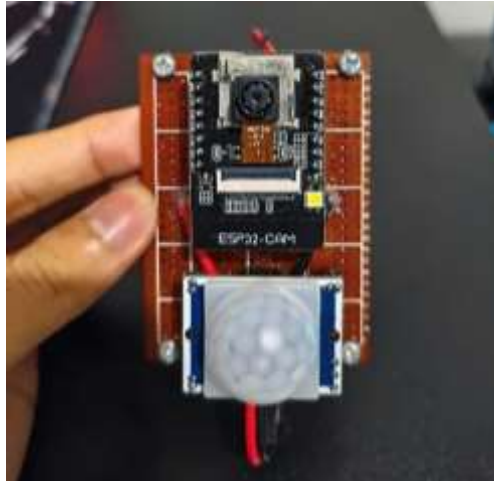
Setelah perbaikan selesai, dilakukan pengujian ulang untuk memastikan sistem mampu mendeteksi objek dengan benar, memberikan tingkat ancaman sesuai ketentuan, mengirimkan notifikasi secara real-time, serta mempertahankan koneksi antara ESP32-CAM, MQTT Broker, server, dan dashboard.

Dengan tahapan tersebut, penelitian menghasilkan prototipe Beastly Vision yang dapat digunakan untuk mendeteksi pergerakan, mengambil dan mengirimkan gambar, mengenali objek menggunakan AI, menentukan tingkat ancaman, serta menampilkan hasil pemantauan secara real-time melalui dashboard web.

HASIL DAN PEMBAHASAN

Implementasi Sistem

Implementasi sistem dilakukan dengan mengintegrasikan seluruh perangkat keras dan perangkat lunak yang telah dirancang menjadi sebuah sistem deteksi intrusi satwa liar yang dapat bekerja secara otomatis. Sistem terdiri dari perangkat utama berupa ESP32-CAM sebagai mikrokontroler sekaligus kamera, sensor PIR HC-SR501 sebagai pendeteksi gerakan, baterai Li-ion sebagai sumber daya, serta dashboard web sebagai media pemantauan.



Gambar 1 Tampilan Depan Alat

Pada bagian atas terdapat modul ESP32-CAM yang dilengkapi kamera OV2640 sebagai perangkat utama untuk mengambil gambar ketika sistem mendeteksi adanya pergerakan. Tepat di bawahnya terpasang sensor PIR HC-SR501 yang berfungsi mendeteksi perubahan radiasi inframerah akibat pergerakan objek di area pemantauan. Kedua komponen dipasang pada PCB berlubang menggunakan baut sehingga posisinya kokoh dan memudahkan proses perawatan maupun penggantian komponen apabila diperlukan.

ESP32-CAM berfungsi sebagai pusat kendali sistem yang menerima sinyal dari sensor PIR, mengambil gambar menggunakan kamera OV2640, kemudian mengirimkan gambar tersebut ke server melalui jaringan Wi-Fi menggunakan protokol MQTT. Sensor PIR HC-SR501 berperan sebagai pendeteksi awal adanya pergerakan di area pemantauan sehingga kamera hanya aktif ketika diperlukan. Dengan mekanisme tersebut, penggunaan daya menjadi lebih efisien dibandingkan apabila kamera melakukan pengambilan gambar secara terus-menerus.



Gambar 2 Tampilan Samping Alat

Susunan komponen dibuat bertingkat menggunakan spacer sehingga memberikan ruang yang cukup untuk jalur kabel dan sambungan antarperangkat. Pada bagian tengah terlihat modul step-up DC converter 5V yang digunakan untuk menstabilkan tegangan keluaran baterai sebelum

dialirkan ke ESP32-CAM. Selain itu, sambungan kabel antar komponen dibuat secara permanen melalui proses penyolderan untuk meningkatkan kestabilan koneksi selama perangkat beroperasi.

Secara umum, alur kerja sistem dimulai ketika sensor PIR mendeteksi adanya pergerakan objek. ESP32-CAM kemudian mengambil gambar dan mengirimkannya ke MQTT Broker. Selanjutnya server backend menerima gambar tersebut untuk diproses menggunakan TensorFlow.js sehingga objek dapat dikenali dan hasilnya ditampilkan pada dashboard Beastly Vision secara *real-time*. Seluruh komponen dihubungkan menggunakan sambungan solder yang meningkatkan kerapihan dan bebas dari perkabelan yang mengganggu. Sebagai sumber daya, sistem menggunakan dua buah baterai Li-ion 18650 yang dipasang pada battery shield sehingga perangkat dapat beroperasi secara mandiri (*off-grid*).

Berdasarkan hasil implementasi, seluruh komponen berhasil dirakit dan diintegrasikan sesuai dengan rancangan sistem. ESP32-CAM berfungsi sebagai pusat pengendali yang menerima sinyal dari sensor PIR, mengendalikan proses pengambilan gambar, serta mengirimkan data ke server menggunakan protokol MQTT melalui jaringan Wi-Fi. Sementara itu, sensor PIR bekerja sebagai pemicu (*trigger*) sehingga kamera hanya aktif ketika mendeteksi adanya pergerakan. Mekanisme ini membuat penggunaan daya menjadi lebih efisien karena proses pengambilan gambar tidak dilakukan secara terus-menerus.

Program ESP32-CAM

Perangkat lunak pada ESP32-CAM dikembangkan menggunakan bahasa pemrograman C++ pada lingkungan Arduino IDE dengan memanfaatkan pustaka utama seperti `esp_camera.h` untuk kontrol kamera dan `PubSubClient.h` untuk komunikasi data. Program ini bertugas mengendalikan seluruh siklus kerja sistem secara terintegrasi, mulai dari inisialisasi pin sensor PIR dan Flash LED, konfigurasi sensor kamera OV2640 pada resolusi VGA, hingga manajemen koneksi Wi-Fi. Selain itu, program menangani komunikasi data dua arah menggunakan protokol MQTT untuk mempublikasikan status perangkat (*heartbeat*), mengirimkan hasil tangkapan gambar secara *real-time*, serta menerima dan mengeksekusi instruksi kendali jarak jauh yang dikirimkan dari dashboard.



```

1 // ESP32-CAM
2 #include "esp_camera.h"
3 #include "Arduino.h"
4 #include "PubSubClient.h"
5 #include "mqtt.h"
6
7 // --- MQTT Broker ---
8 const char *MQTT_HOST = "broker.emqx.io";
9 const int  MQTT_PORT = 1883;
10
11 // --- MQTT Client ---
12 const char *MQTT_USER = "user";
13 const char *MQTT_PASS = "password";
14
15 // --- MQTT Topic ---
16 const char *MQTT_TOPIC = "beastlyvision";
17
18 // --- Device ID ---
19 const char *DEVICE_ID = "esp32-cam-2";
20
21 // --- Interval Heartbeat (ms) ---
22 const int HEARTBEAT_INTERVAL_MS = 10000; // 10 detik
23
24 // --- Buffer MQTT ---
25 const int MQTT_BUFFER_SIZE = 51200; // 50 KB
26
27 // --- Pin Definitions ---
28 #define PIR_PIN 15 // GPIO 15 -> PIR sensor VCC
29 #define FLASH_PIN 4 // GPIO 4 -> LED Flash (built-in AT-Transceiver)
30
31 // --- Camera Module (AT-Transceiver) ---
32 #define CAMERA_MODULE_AT_TRANSCEIVER
33 #include "camera_pins.h"
34
35 // --- MQTT Client ---
36 PubSubClient mqtt(MQTT_HOST);
37
38 // --- Topic Strings (Global) ---
39 String topicFrame;
40 String topicStatus;

```

Gambar 3 Code ESP32-CAM Pada Arduino IDE

Gambar di atas menunjukkan interface software Arduino IDE dan tampilan dari script yang sudah dituliskan. Program ini telah diketikkan ke software Arduino IDE untuk memungkinkan pembentukan sistem tersebut.

Program diawali dengan proses inisialisasi seluruh pustaka yang dibutuhkan, seperti *esp_camera*, *WiFi*, *PubSubClient*, dan beberapa pustaka pendukung lainnya. Setelah itu sistem melakukan konfigurasi pin GPIO yang digunakan oleh sensor PIR dan lampu *flash*. Selanjutnya ESP32-CAM melakukan koneksi ke jaringan Wi-Fi menggunakan SSID dan kata sandi yang telah ditentukan. Setelah koneksi berhasil, perangkat akan terhubung ke EMQX MQTT Broker sehingga dapat melakukan proses *publish* dan *subscribe*.

Setelah koneksi MQTT berhasil dilakukan, sensor PIR menjalani proses kalibrasi selama kurang lebih lima detik agar pembacaan sensor menjadi stabil. Tahap ini penting untuk mengurangi kemungkinan terjadinya deteksi palsu ketika perangkat pertama kali dinyalakan.

Program kemudian memasuki mode *always-on*, yaitu kondisi di mana ESP32-CAM terus memantau status sensor PIR sekaligus mendengarkan pesan dari MQTT Broker. Ketika sensor mendeteksi adanya pergerakan, ESP32-CAM akan menyalakan lampu *flash*, mengambil gambar menggunakan kamera OV2640, kemudian mematikan kembali lampu *flash* setelah proses pengambilan gambar selesai. Gambar yang diperoleh selanjutnya dikirim ke MQTT Broker dalam bentuk raw binary berformat JPEG melalui topik *beastlyvision/device_id/frame/jpeg*. Selain mengirim

gambar, perangkat juga mengirimkan *heartbeat* setiap sepuluh detik sebagai indikator bahwa perangkat masih aktif dan terhubung dengan server. ESP32-CAM juga menerima berbagai perintah dari dashboard melalui topik `beastlyvision/device_id/cmd`. Perintah tersebut meliputi pengambilan gambar secara manual (*manual capture*), mengaktifkan atau menonaktifkan sensor PIR, mengendalikan lampu *flash*, serta melakukan *reboot* perangkat apabila diperlukan. Dengan adanya komunikasi dua arah melalui MQTT, pengelola dapat mengoperasikan perangkat dari jarak jauh tanpa harus mendatangi lokasi pemasangan.

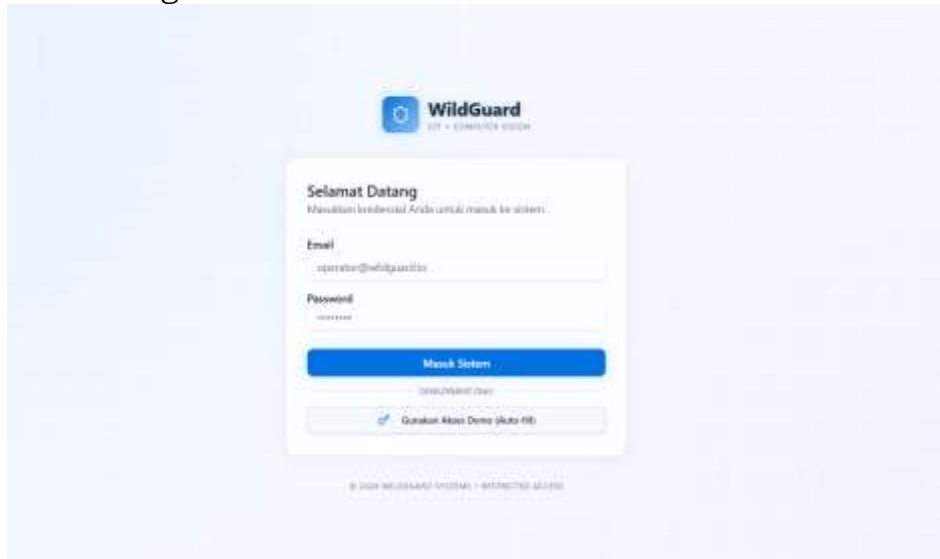
Table 1 Penjelasan Code Program ESP32-CAM

NO	POTONGAN CODE	FUNGSI PROGRAM
1	<code>#include "esp_camera.h"</code>	Library untuk mengendalikan kamera OV2640 pada modul ESP32-CAM.
2	<code>#include <Arduino.h></code>	Library utama Arduino yang menyediakan fungsi dasar seperti <code>setup()</code> , <code>loop()</code> , <code>pinMode()</code> , dan <code>digitalWrite()</code> .
3	<code>#include <PubSubClient.h></code>	Library untuk komunikasi menggunakan protokol MQTT antara ESP32-CAM dan EMQX Broker.
4	<code>#include <WiFi.h></code>	Digunakan untuk menghubungkan ESP32-CAM ke jaringan Wi-Fi.
5	<code>const char *WIFI_SSID</code> dan <code>WIFI_PASSWORD</code>	Menyimpan informasi SSID dan password jaringan Wi-Fi yang digunakan perangkat.
7	<code>DEVICE_ID</code>	Menentukan identitas perangkat
9	<code>MQTT_BUFFER_SIZE</code>	Menentukan ukuran buffer MQTT untuk mengirim citra JPEG dalam bentuk <i>raw binary</i> .
10	<code>#define PIR_PIN</code>	Mendefinisikan GPIO 13 sebagai pin masukan sensor PIR.
11	<code>#define FLASH_PIN</code>	Mendefinisikan GPIO 4 sebagai pin keluaran LED flash bawaan ESP32-CAM.
12	<code>#define CAMERA_MODEL_AI_THINKER</code>	Menentukan tipe modul kamera yang digunakan, yaitu AI Thinker ESP32-CAM.
13	<code>WiFiClient wifiClient</code>	Membuat objek koneksi jaringan Wi-Fi.
14	<code>void setup()</code>	Melakukan inisialisasi perangkat, konfigurasi pin, koneksi Wi-Fi, kamera, sensor PIR, dan MQTT Broker saat ESP32-CAM dinyalakan.
15	<code>void loop()</code>	Menjalankan proses utama sistem secara terus-menerus, meliputi pemantauan sensor PIR, komunikasi MQTT, pengiriman <i>heartbeat</i> , dan pengambilan gambar ketika terjadi deteksi gerakan.

Implementasi Dashboard

Pengujian dashboard dilakukan untuk memastikan bahwa seluruh fitur pada Beastly Vision dapat berfungsi sesuai dengan kebutuhan sistem yang telah dirancang. Pengujian mencakup tampilan antarmuka (user interface), proses komunikasi data antara dashboard dengan server, penerimaan hasil deteksi objek dari ESP32-CAM, serta pengujian fungsi kendali perangkat melalui protokol MQTT. Selain itu, dilakukan pula pengujian terhadap penyimpanan data deteksi ke dalam database PostgreSQL dan pembaruan informasi pada dashboard secara otomatis. Dashboard dikembangkan menggunakan framework React.js sehingga mampu menampilkan perubahan data secara real-time tanpa memerlukan proses reload halaman. Dengan demikian, pengguna dapat memantau kondisi perangkat, melihat hasil deteksi, serta mengendalikan ESP32-CAM secara langsung melalui antarmuka web dengan respons yang cepat dan stabil.

1. Halaman Login



Gambar 4 Tampilan Login

Halaman login merupakan gerbang utama sebelum pengguna dapat mengakses sistem. Pengguna harus memasukkan username dan password yang telah terdaftar. Setelah proses autentikasi berhasil, pengguna akan diarahkan menuju halaman utama dashboard.

2. Menu Live Monitoring



Gambar 5 Tampilan Live Monitoring

Halaman Live Monitoring digunakan untuk menampilkan gambar terbaru yang dikirim oleh ESP32-CAM beserta hasil deteksi objek secara *real-time*. Selain itu, halaman ini menyediakan tombol Manual Capture, Flash, PIR ON/OFF, dan Reboot Device yang memungkinkan pengguna mengendalikan perangkat dari jarak jauh melalui MQTT.

3. Menu Riwayat Deteksi



Gambar 6 Tampilan Riwayat Deteksi

Halaman ini menampilkan seluruh riwayat deteksi dari database PostgreSQL, termasuk waktu, gambar hasil deteksi, nama objek, persentase confidence, dan tingkat ancaman. Pengguna juga dapat mencari dan menyaring data berdasarkan kriteria tertentu, serta menghapus riwayat deteksi baik secara satuan maupun massal.

Kekurangan dan Kelebihan Sistem

Berdasarkan hasil pengujian sistem Beastly Vision, berikut adalah poin-poin kelebihan dan kekurangan sistem yang berhasil diidentifikasi:

a. Kelebihan Sistem

- 1) Efisiensi Konsumsi Daya: Menggunakan konsep event-driven berbasis sensor PIR, sehingga kamera ESP32-CAM dan lampu flash hanya aktif saat mendeteksi
- 2) Pengiriman Data Cepat (Real-Time): Protokol MQTT melalui EMQX Broker memiliki latensi sangat rendah (di bawah 5 detik) dan hemat bandwidth, memastikan gambar dan status perangkat terkirim dengan cepat ke dashboard.
- 3) Filter Ancaman dan Notifikasi Pintar: Model AI TensorFlow.js mengklasifikasikan objek secara otomatis. Level ancaman manusia diset rendah (Low) untuk menghindari spam, sedangkan ancaman tinggi (High seperti satwa liar) langsung memicu push notification
- 4) Kendali Dua Arah: Dashboard tidak hanya menerima data, tetapi juga dapat mengirimkan perintah kontrol balik seperti capture manual, kontrol flash, sensor PIR on/off, hingga reboot perangkat secara jarak jauh.

b. Kekurangan Sistem

- 1) Ketergantungan pada Jaringan Internet: Sistem membutuhkan koneksi Wi-Fi dan internet yang stabil. Jika sinyal terputus di area outdoor, monitoring real-time dan notifikasi tidak dapat berjalan.
- 2) Beban Komputasi di Sisi Klien (*Client-Side*): Pemrosesan AI TensorFlow.js berjalan pada browser pengguna. Perangkat smartphone atau PC dengan spesifikasi rendah dapat mengalami kelambatan saat menganalisis gambar.
- 3) Kualitas Gambar Malam Hari Terbatas: Resolusi kamera OV2640 dibatasi pada tingkat VGA demi kecepatan kirim. Di malam hari, kualitas gambar menurun (noise tinggi) yang berpotensi mengurangi akurasi deteksi AI.
- 4) Keterbatasan Fisik Sensor PIR: Deteksi berbasis suhu infra merah membuat sensor PIR kurang sensitif jika suhu lingkungan di siang hari sangat panas atau mendekati suhu tubuh objek.
- 5) Dataset tidak mencakup seluruh jenis satwa liar: Keterbatasan ini disebabkan oleh penggunaan model AI TensorFlow.js (COCO-SSD) yang memiliki keterbatasan bawaan, yaitu hanya mampu mendeteksi 80 kategori objek yang telah dilatih sebelumnya (pre-trained). Hal ini bertujuan untuk menghemat ruang penyimpanan dan meminimalkan beban komputasi pada CPU. Oleh karena itu, pada proyek saat ini, pemrosesan AI dijalankan sepenuhnya menggunakan CPU. Jika di kemudian hari sistem ingin ditingkatkan agar dapat mengenali lebih banyak jenis satwa liar, maka akan dibutuhkan perangkat keras server dengan spesifikasi komputasi yang lebih tinggi (seperti dukungan GPU dedicated) agar AI dapat bekerja secara optimal."

KESIMPULAN

Berdasarkan hasil perancangan, implementasi, dan pengujian sistem monitoring struktur ringan berbasis *Internet of Things* (IoT), dapat

disimpulkan bahwa sistem deteksi intrusi satwa liar berhasil dirancang dan diimplementasikan dengan mengintegrasikan ESP32-CAM sebagai pengendali utama dan sensor PIR HC-SR501 sebagai pendeteksi pergerakan pada area pemantauan. Integrasi seluruh komponen memungkinkan sistem bekerja secara terpadu dalam mendeteksi keberadaan objek dan melakukan proses pemantauan secara otomatis. Sensor PIR HC-SR501 berfungsi sebagai pemicu pengambilan gambar ketika mendeteksi adanya pergerakan, kemudian ESP32-CAM secara otomatis menangkap gambar dan mengirimkannya ke server menggunakan protokol MQTT melalui EMQX Broker. Data hasil deteksi selanjutnya dapat dipantau melalui dashboard Beastly Vision yang mampu menampilkan informasi secara real-time, menyimpan riwayat deteksi, serta menyediakan fitur kendali perangkat dari jarak jauh. Selain itu, penerapan TensorFlow.js dengan model COCO-SSD memungkinkan sistem melakukan deteksi objek pada gambar yang dikirim oleh ESP32-CAM dengan menghasilkan informasi berupa nama objek, nilai confidence, dan bounding box. Hasil deteksi tersebut kemudian disimpan dan ditampilkan pada dashboard sehingga dapat memberikan informasi yang lebih jelas kepada pengguna mengenai objek yang teridentifikasi pada area pemantauan. Dengan demikian, sistem yang dikembangkan telah mampu mengintegrasikan proses deteksi gerakan, pengambilan dan pengiriman gambar, identifikasi objek, penyimpanan data, serta pemantauan dan pengendalian perangkat secara terpadu dalam satu sistem berbasis IoT.

DAFTAR PUSTAKA

- Alui, N., & Mohamed, M. (2023). The Prototype of Animal Intrusion Detection and Monitoring System using IoT. *Evolution in Electrical and Electronic Engineering*, 4(2), 707–715.
<http://publisher.uthm.edu.my/periodicals/index.php/eeee>
- Arrahma, S. A., & Mukhaiyar, R. (2023). *Pengujian Esp32-Cam Berbasis Mikrokontroler*. 4(1), 60–66.
- Aryunita, F., Rasyid, N., & Barat, U. S. (2024). *Rancangbangun Sistem Monitoring Keamanan Kandang Ayam Bloiler Menggunakan Esp32-*. 12(1).
- Barelvi, A., & Basri, M. (2025). Smart Warning System Prototype DAM. *Tsabit Journal of Computer Science*, 2(1), 9–16.
<https://doi.org/10.56211/tsabit35>
- Bayu Unggul Sejati, & Hengky Triyo. (2025). Sistem Pemantauan Gudang Berbasis ESP32-Cam untuk Deteksi Gerakan dan Keberadaan Objek. *Just IT: Jurnal Sistem Informasi, Teknologi Informasi Dan Komputer*, 15(2), 356–361. <https://jurnal.umj.ac.id/index.php/just-it/index>
- Bazir, D., Haz, H., Saputra, J., Choirina, P., Jurusan,), Elektronika, T., Senjata, S., Darat, A., & Anggrek, J. R. (2022). Sistem Pendeteksi Insurjen Pada Bod (Basis Operasi Depan) Di Daerah Pos Dengar Menggunakan Sensor Passive Infrared Hc-Sr501 Insurgent Detection System on Bod (Front Operating Base) in Hearing Posting Area Using Passive Infrared Sensor Hc-Sr501. *Jurnal Elkasista*, 3, 2723–1291.
- Cakrayuda, L. (2025). Sicemot: Sistem Keamanan Cerdas Berbasis Esp32-Cam, Sensor Gerak, Dan Notifikasi Telegram. *Jurnal Informatika Dan Teknik Elektro Terapan*, 13(2).

- <https://doi.org/10.23960/jitet.v13i2.6420>
- Fadhlurrohman, D., & Basri, M. (2025). Home Anti Theft System Uses Based Telegram Bot *Internet of Things*. *Hanif Journal of Information ...*, 1–8. <https://journal.ilmubersama.com/index.php/hanif/article/view/36%0Ahttps://journal.ilmubersama.com/index.php/hanif/article/download/36/36>
- Hasanah, U., Syaputra, M., & Sari, D. P. (2022). *Pada Tipe Gangguan Habitat Berbeda Di Desa Ntori Types of Habitat Disturbance At Ntori Village , Wawo Sub*. 1–10.
- Ikhsan, M. K., & Sari, I. P. (2025). Perancangan Sistem Keamanan Kamar Kos Berbasis *Internet of Things* (IoT) dengan Menggunakan Wemos D1 Mini dan Sensor Pintu, serta Integrasi Pemberitahuan Melalui Telegram. *Blend Sains Jurnal Teknik*, 3(2), 229–236. <https://doi.org/10.56211/blendsains.v3i2.639>
- Kudus, M., Budhi Santoso, D., & Nurpulaela, L. (2023). Purwarupa Alat Monitoring Satwa Langka Berbasis ESP32-CAM. *JIM: Jurnal Ilmiah Mahasiswa Pendidikan Sejarah*, 8, 719–724. <https://doi.org/10.24815/jimps.v8i2.24792>
- Lubis, D. F., & Basri, M. (2025). *Prototype of Salt Content Monitoring System in Salted Fish Production Process Using Internet of Things*. 2(1), 1–6.
- Lubis, N. H., & Basri, M. (2024). *Pengembangan Sistem Smart Rubbish Organik Dan Non Organik Otomasi Berbasis IoT*.
- Muhammad Riesky Pratama, D. L. (2024). *Prototipe Sistem Deteksi Burung Menggunakan*. 27(November), 78–87.
- Purnama Sari, I., Azhari, M., & Riady Hasibuan, A. (2024). Design and Construction of Room Door Security Using ESP 32 Cam and Blynk Based on *Internet of Things*. *Altafani: Jurnal Pengabdian Masyarakat*, 1(2), 50–57.
- Rifaie, F., & Arida, E. A. (2022). Pemetaan konflik manusia dengan biawak (*Varanus salvator*) berbasis web scraping berita online. *Prosiding Seminar Nasional MIPA UNIBA*, 201–209.
- Sari, I. P., Apdilah, D., & Guntur, S. (2025). Sistem Smart Class Berbasis *Internet of Things* (IoT). *Sudo Jurnal Teknik Informatika*, 4(1), 33–39. <https://doi.org/10.56211/sudo.v4i1.778>
- Sari, I. P., Novita, A., Al-Khowarizmi, A.-K., Ramadhani, F., & Satria, A. (2024). Pemanfaatan *Internet of Things* (IoT) pada Bidang Pertanian Menggunakan Arduino UnoR3. *Blend Sains Jurnal Teknik*, 2(4), 337–343. <https://doi.org/10.56211/blendsains.v2i4.505>
- Sitopu, J. W., Safarudin, M. S., Wahyu, M., Adam, S., Safar, M., Ichsan, U., Rappang, S., Simalungun, U., & Bone, U. M. (2024). *Mengenal Internet of Things (IoT): Penerapan Konsep dan Manfaatnya dalam Kehidupan Sehari-hari*. 4(4), 827–835.
- Syahri, A., & Basri, M. (2025). Perancangan Sistem Keamanan untuk Deteksi Pencurian Menggunakan Sensor Magnetic dan RFID Berbasis *Internet of Things*. *Hello World Jurnal Ilmu Komputer*, 4(3), 163–170. https://jurnal.ilmubersama.com/index.php/hello_world/article/view/1166
- Wulandari, Erniawati, & Siswahyono. (2023). Konflik Manusia Dengan Harimau Sumatera (*Panthera Tigris Sumatrae*) Di Hutan Produksi Yang

Dapat Di Konversi (Hpk) Dan Hutan Produksi Terbatas (Hpt) Air Ketahun Dusun Limas Jaya Bengkulu Utara (Human Conflict With Sumatran Tiger (*Panthera tigris sumatrae*) . *Journal of Global Forest and Environmental Science*, 3(1), 19–34.

Xin, L. M., & Rosely, N. F. N. (2025). Development of Machine Learning Wildlife Camera Using ESP32 CAM for Small Mammals. *Pertanika Journal of Science and Technology*, 33(4), 1783–1807. <https://doi.org/10.47836/pjst.33.4.05>.