

HARMONI: Home Automation Module Berbasis Internet of Things dan Deep Learning

Muhammad Ma'sum Juniyanto¹, Bernadus Anggo Seno Aji, S. Kom., M. Kom.², Muhammad Adib Kamali, S. T., M. Eng.³

^{1,2,3}Program Studi Teknologi Informasi, Fakultas Informatika, Telkom University Kampus Surabaya, Jln. Ketintang No. 156, Kota Surabaya, 60231, Indonesia

Info Artikel

Riwayat Artikel:

Received 2024-07-08

Revised 2025-05-04

Accepted 2025-05-25

Corresponding Author:

Bernadus Anggo Seno Aji

Email:

bernadusanggosenoaji@telkomuniversity.ac.id



This is an open access article under the [CC BY 4.0](https://creativecommons.org/licenses/by/4.0/) license.

Abstract – Electrical appliances that are not turned off when not in use often cause electrical short circuits that result in catastrophic fires. In addition, this also has the potential to waste the use of electrical energy. People connect electrical appliances directly to the power source through a socket or through an electrical line and then connected to a switch, in its operation. It is often experienced as negligence in turning it off or unplugging it, making it potentially dangerous. An IoT and Deep Learning-based home automation module was created by the author to digitize and automate switches. This system consists of ESP32-S3 and ESP32 microcontrollers that function as the main controllers, relay modules used as automatic switches, camera modules to detect the presence of people, Google Home integration through the Sinric.Pro platform, and the Mowny website that has been integrated with the HTTPS protocol. Microcontrollers, modules, relays are arranged on printed-circuit boards. Mowny website to control the switch and monitor the room. People detection using YOLO as a trigger for switch automation. The detection model is loaded via API to be accessed on the website. System testing included four scenarios for digital and automatic switch on and off, resulting in the following response times (in seconds): Google Home ($\pm 3,468$), Google Assistant ($\pm 4,348$), Mowny website ($\pm 1,042$), and object detection automation ($\pm 19,375$). This automation module can control electrical appliances digitally and automatically, which has an impact on the ease of operating the switch when experiencing negligence in turning off electrical appliances.

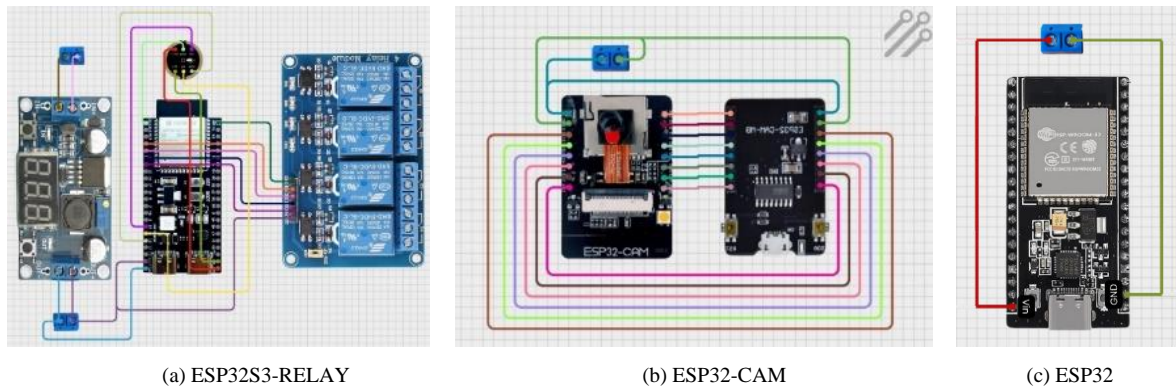
Keywords: Automation; Detection; Efficiency; Module; Switch.

Abstrak – Alat listrik yang tidak dimatikan saat tidak digunakan seringkali menyebabkan terjadinya korsleting listrik yang berakibat bencana kebakaran. Selain itu, hal ini juga berpotensi dalam pemborosan penggunaan energi listrik. Orang-orang menyambungkan alat listrik langsung pada sumber listrik melalui stop kontak atau melalui jalur listrik kemudian dihubungkan dengan sakelar, dalam pengoperasiannya. Seringkali dialami kelalaian dalam mematikan atau mencabutnya, sehingga berpotensi membahayakan. Modul otomasi rumah berbasis IoT dan Deep Learning dibuat oleh penulis untuk melakukan digitalisasi dan otomasi sakelar. Sistem ini terdiri dari mikrokontroler ESP32-S3 dan ESP32 yang berfungsi sebagai pengendali utama, modul relay yang digunakan sebagai sakelar otomatis, modul kamera untuk mendeteksi keberadaan orang, integrasi Google Home melalui platform Sinric.Pro, serta website Mowny yang telah diintegrasikan dengan protokol HTTPS. Mikrokontroler, modul, relay disusun pada papan-sirkuit-cetak. Website Mowny untuk mengontrol saklar dan monitoring ruangan. Pendeteksian keberadaan orang menggunakan YOLO sebagai pemicu otomasi sakelar. Model deteksi dimuat melalui API untuk diakses pada website. Pengujian sistem meliputi empat skenario untuk menyalakan-matikan sakelar secara digital dan otomatis, menghasilkan waktu respon sebagai berikut (dalam satuan detik): Google Home ($\pm 3,468$), Google Assistant ($\pm 4,348$), website Mowny ($\pm 1,042$), dan otomasi deteksi objek ($\pm 19,375$). Modul otomasi ini dapat mengontrol alat listrik dari secara digital dan otomatis, yang berdampak pada kemudahan pengoperasian sakelar ketika mengalami kelalaian mematikan alat listrik.

Kata Kunci: Deteksi, Efisiensi, Modul, Otomasi, Saklar

I. PENDAHULUAN

Pada era modern saat ini, listrik telah menjadi suatu keperluan esensial bagi kehidupan manusia. Pada siang hari, manusia sering mengandalkan perangkat pendingin udara seperti AC untuk menjaga suhu ruangan agar tetap nyaman, sementara pada malam hari, penerangan dari lampu menjadi suatu keharusan untuk mendukung berbagai aktivitas. Banyaknya aktivitas dan mobilitas individu baik di siang maupun malam hari seringkali mengakibatkan kurangnya kesadaran terhadap keadaan perangkat listrik yang tengah beroperasi, termasuk lampu, AC, kipas angin, maupun kompor listrik. Seperti yang diberitakan dalam berbagai insiden kebakaran rumah, sebagian di antaranya disebabkan oleh penghuni yang lupa mematikan peralatan listrik seperti kipas angin atau pompa air saat meninggalkan



Gambar 1. Rangkaian elektronik pada PCB (Rangkaian elektronik yang menghubungkan relay sebagai saklar, kamera sebagai pemantau keberadaan orang, dan pengirim *trigger* di aplikasi)

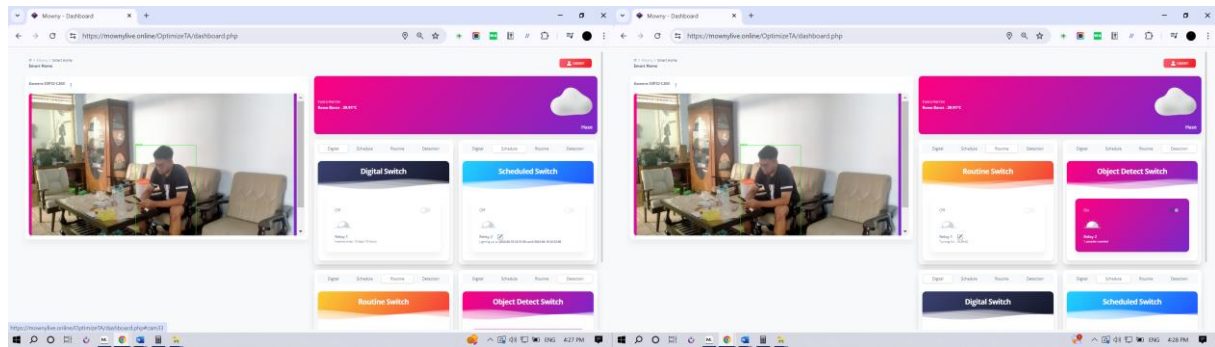
rumah yang akhirnya mengakibatkan arus pendek listrik [1][2]. Banyak orang yang menggampangkan kelalaian ini karena menganggap alat listrik sudah aman, namun di sisi lain potensi bahaya akan selalu ada. Hal lain adalah karena perlu pergerakan untuk mematikan alat listrik misalnya melepas steker atau mematikan saklar. Jika kelalaian mematikan masih di dalam rumah, tentu tidak membutuhkan waktu lama, namun jika orang terlanjur pergi ke luar rumah untuk beraktivitas lain, tentu memerlukan waktu lebih. Oleh karenanya, perlu dikembangkan sebuah modul saklar otomatis yang dapat memantau kondisi ruangan dan mampu untuk mematikan serta menyalakan alat listrik secara otomatis dari tempat pengguna atau orang-orang berada.

Salah satu cara yang dapat dimanfaatkan untuk pengontrolan ini adalah menggunakan *Internet of Things (IoT)*. Tentu saat ini sudah banyak orang yang mudah dalam mendapatkan akses internet, sehingga ini dapat dimanfaatkan. Iot mampu menyediakan sebuah layanan untuk menghubungkan kontrol yang diberikan pengguna dengan alat-alat listrik yang terhubung pada sebuah modul. Upaya seperti ini akan menghemat waktu untuk mematikan alat listrik. Pendekatan seperti *deep learning* juga digunakan, untuk meniru penilaian manusia. Dalam hal ini penilaian terkait penentuan keberadaan orang dalam sebuah ruangan yang memberikan pemicu bagi alat listrik apakah harus dalam kondisi mati atau menyala. Model *deep learning* yang digunakan adalah YOLO yang mampu untuk melakukan pendeteksian terhadap suatu objek (orang misalnya) untuk nantinya hasil dari pendeteksian dihubungkan dalam sistem kontrol IoT di dalam modul.

Dalam konteks integrasi aplikasi dan sistem, Google Home digunakan dengan dukungan *Internet of Things (IoT)* melalui mikrokontroler ESP32 dengan memanfaatkan platform Sinric.pro untuk melakukan integrasi, ESP32-S3 untuk melakukan kontrol pada sakelar, dan ESP32-CAM untuk mendeteksi keberadaan manusia. Data gambar yang diambil oleh ESP32-CAM akan diproses menggunakan algoritma YOLO-V8 untuk mendeteksi kehadiran manusia. Dibuat sebuah *website* (mownyonline.live) untuk dapat melakukan *monitoring* ruangan dan kontrol terhadap masing-masing saklar yang terhubung pada ESP32-S3 yang diintegrasikan dengan protokol HTTP. Dalam sistem ini, *input* terdiri dari dua macam, input pertama berupa *trigger* yang dioperasikan melalui menu perangkat pada aplikasi Google Home, perintah suara pada Google Assistant, dan elemen *button* yang ada pada *website* mowny. *Input* kedua berupa hasil deteksi jumlah orang yang dibaca pada model YOLO yang dibuat. Kedua input ini melakukan *update* pada *database* yang kemudian dapat menjadi acuan untuk penyalakan saklar secara otomatis dan *real time*.



Gambar 2. Rangkaian elektronik yang telah terpasang



Gambar 3. Dashboard website kontrol saklar dan monitor kamera

Tujuan yang hendak dicapai terdapat dua tujuan, untuk tujuan pertama adalah merancang dan mengimplementasikan modul otomasi pada barang elektronik rumah dengan basis IoT serta mengintegrasikan dengan aplikasi *Google Home* untuk dapat dikontrol dari jarak jauh. Di sini modul berperan sebagai kontrol dan monitor melalui internet. Untuk tujuan kedua adalah membuat sistem otomasi barang elektronik berbasis keberadaan objek orang. Di sini modul berperan untuk dapat melakukan otomasi penyalamatian saklar dengan berdasar pada keberadaan orang pada ruangan.

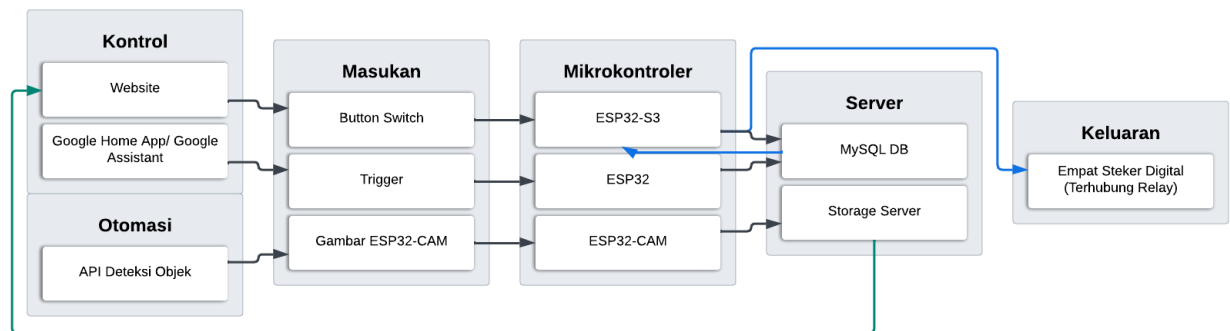
Penelitian terkait sebagai berikut, penelitian pertama membahas pengendalian alat listrik rumah tangga melalui internet menggunakan RaspberryPi, relay, dan platform Cayenne dengan MQTT [3]. Meskipun mampu mengendalikan lampu melalui Cayenne, belum ada otomasi yang ditunjukkan. Penelitian kedua menggunakan ESP 32, modul arus, modul relay, sensor suhu DHT11, dan buzzer untuk monitoring daya, kontrol lampu, pemantauan suhu, dan perkiraan tarif listrik pada platform Blynk [4]. Meskipun ada notifikasi suhu tinggi, belum ada otomasi saklar yang terlihat. Penelitian ketiga mengenai lampu otomatis dengan pengontrolan suara menggunakan IFTTT dan Blynk IoT [5]. Meskipun ada pengendalian suara, belum ada otomasi yang terlihat. Penelitian keempat menggunakan NodeMCU ESP8266, IFTTT, dan MQTT untuk kontrol lampu dengan Google Assistant [6]. Meskipun pengontrolan digital berjalan baik, belum ada otomasi. Penelitian kelima membahas sistem pendeteksi jumlah orang menggunakan YOLOv3 dan ESP32-CAM dengan Raspberry Pi sebagai server database [7]. Dataset dari Kaggle ditambahkan untuk pembentukan model dengan tujuan membantu proses klasifikasi objek sesuai kondisi yang dibutuhkan. Fokus penelitian ini untuk mendeteksi jumlah orang tanpa ada aksi setelahnya dari hasil deteksi yang diperoleh. Penelitian keenam tentang pendeteksi jarak objek manusia untuk mencegah penyebaran virus, menggunakan Raspberry Pi, webcam, dan algoritma YOLO dengan akurasi 90% [8]. Pada penelitian ini hanya berfokus pada pendeteksian jumlah dan jarak antar objek orang. Penelitian ketujuh menggunakan *Google Speech to Text*, ESP 32, dan sensor suhu DHT11 untuk kendali suara lampu dan pemantauan cuaca [9]. Pada penelitian ini *input* berupa suara dan sensor suhu yang masih menyesuaikan dengan lingkungan, bukan pada kebutuhan adanya orang pada lingkungan/ ruangan. Penelitian kedelapan membahas sistem otomasi rumah pintar berbasis suara dengan NLP (*Natural Language Processing*), chatBot, HTTP, dan MQTT untuk menerjemahkan perintah suara pengguna menjadi aksi pada perangkat rumah [10]. Fokus penelitian ini berupa *input* suara melalui chatBot yang memerlukan pemrosesan lebih lama dari pendeteksian adanya objek orang.

Penelitian ini bertujuan untuk mengembangkan sistem IoT cerdas yang mampu mendeteksi keberadaan objek, khususnya manusia, secara real-time menggunakan algoritma YOLO, dan secara otomatis mengaktifkan aktuator melalui mikrokontroler seperti ESP32. Sistem ini dirancang agar dapat mengambil keputusan berbasis visual secara langsung di perangkat edge, tanpa perlu mengandalkan server eksternal, sehingga tetap dapat berfungsi meskipun dalam kondisi terbatas konektivitas atau daya komputasi.

Kontribusi utama dari penelitian ini adalah integrasi algoritma deteksi objek ke dalam sistem IoT berbasis mikrokontroler yang ringan, yang memungkinkan sistem merespons kondisi lingkungan secara adaptif dan kontekstual. Pekerjaan ini penting karena menawarkan solusi yang lebih cerdas dibandingkan sistem IoT konvensional yang hanya mengandalkan sensor sederhana. Dengan pendekatan ini, sistem dapat meningkatkan efisiensi energi, kenyamanan pengguna, dan fleksibilitas dalam pengembangan smart environment di berbagai bidang seperti rumah pintar, industri, atau pertanian.

II. METODE

A. Arsitektur Jaringan



Gambar 4. Diagram Sistem

Diagram sistem pada gambar 4, menggambarkan arsitektur dari sistem otomasi berbasis *Internet of Things* (IoT) yang terintegrasi antara kontrol manual, otomatis, dan sistem pemantauan berbasis kamera. Sistem ini dibagi ke dalam lima komponen utama, yaitu Kontrol, Masukan, Mikrokontroler, Server, dan Keluaran. Pada bagian kontrol, pengguna dapat mengendalikan sistem secara manual melalui antarmuka web atau menggunakan perintah suara dengan bantuan Google Home App atau Google Assistant. Di sisi lain, sistem ini juga dilengkapi fitur otomasi yang menggunakan API deteksi objek berbasis kamera (dari ESP32-CAM) untuk mengenali adanya objek orang dan secara otomatis mengaktifkan perintah nyala/ mati saklar. Kontrol ini memberikan fleksibilitas dalam mengatur perangkat, baik dari jarak jauh maupun secara otomatis berdasarkan pengenalan visual.

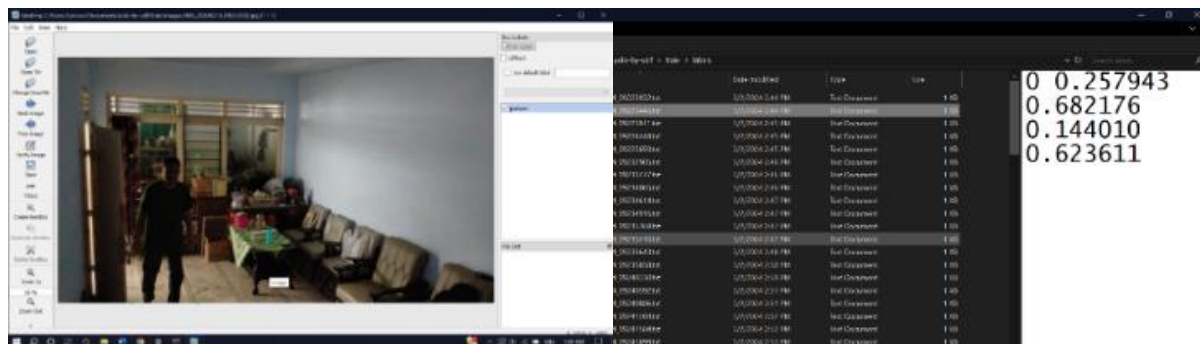
Komponen Masukan terdiri dari tiga jenis input utama: *Button Switch* sebagai pemicu manual dari *website*, *Trigger* sebagai masukan otomatis dari perintah suara/ *Google Home App*, dan Gambar dari ESP32-CAM sebagai input visual yang dikirim ke sistem untuk dianalisis oleh API deteksi objek. Semua input ini kemudian diproses oleh *Mikrokontroler*, yang terdiri dari ESP32-S3 (untuk menangani input manual dari *website*), ESP32 (untuk mengelola input dari perintah suara/ *Google Home App*), dan ESP32-CAM (yang bertugas mengambil gambar dan mengirimkan ke server atau API). Hasil pemrosesan dan pengambilan keputusan dikirim ke Server, yang menyimpan data ke dalam MySQL Database untuk logika kontrol dan Storage Server untuk meng-*update* gambar. Akhirnya, berdasarkan hasil kontrol dari mikrokontroler dan server, sistem akan mengaktifkan salah satu dari Empat Steker Digital yang terhubung ke relay untuk mengendalikan perangkat listrik seperti lampu, kipas, atau aktuator lainnya.

B. Perangkaian Model Elektronik

Rangkaian elektronik dibuat pada *printed circuit board* (PCB) dengan jalur rangkaian seperti ditampilkan pada Gambar 1. Rangkaian modul PCB untuk menghubungkan ESP32-S3 dengan Relay. ESP32-S3 membaca data dari database untuk mengecek kondisi setiap relay. Kondisi dari relay ini kemudian menjadi trigger untuk menyalakan atau mematikan relay. ESP32-CAM digunakan untuk mengambil gambar dari kondisi ruangan. Pengambilan dan pengiriman gambar ke server diulangi setiap 0,1 detik. Rentang tersebut dipilih untuk membuat gambar yang didapat adalah kondisi terkini dari lingkungan ruang. ESP32 dihubungkan dengan hanya memanfaatkan power atau tegangan dari modul untuk dapat bekerja. Fungsi dari ESP32 ini adalah untuk melakukan pengiriman trigger dari aplikasi Google Home dan melalui perintah suara Google Assistant. Rangkaian pada alat yang telah dihubungkan sebagai Gambar 2.

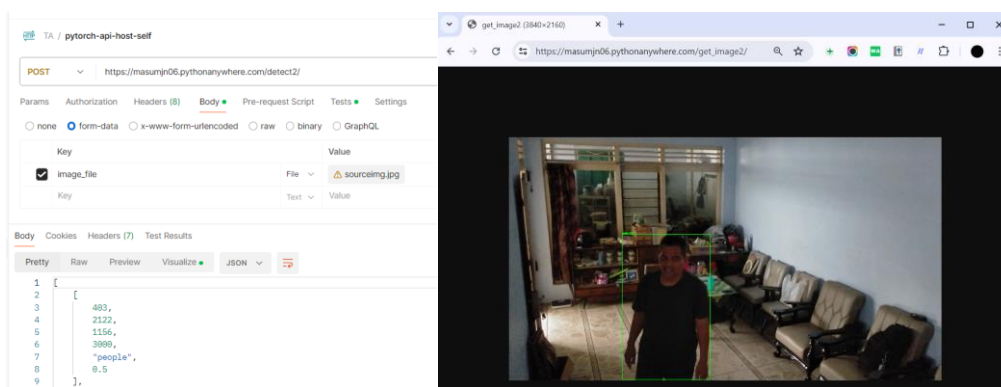
C. Pembuatan Dashboard Website

Pembuatan dashboard website dimaksudkan untuk melakukan monitor kondisi ruangan yang didapat dari pengiriman gambar oleh ESP32-CAM dan kontrol saklar otomatis pada relay. Server pada website ini juga dimanfaatkan untuk penyimpanan gambar dari ESP32-CAM, sehingga lebih memudahkan untuk penampilan gambar dan deteksi orang pada gambar. Dashboard berisi tombol-tombol untuk menyalakan/ mematikan relay, tombol ini terhubung dengan fungsi untuk melakukan update data pada database, sehingga ketika kondisi berubah, secara langsung dapat men-trigger relay. Terdapat empat mode untuk tiap relay yang terhubung. Mode pertama, switch digital yang dapat dioperasikan langsung melalui website, Google Home, atau Google Assistant. Mode kedua, mode jadwal khusus yang akan menyalakan dan mematikan relay pada waktu dan tanggal tertentu. Mode ketiga, mode jadwal rutin yang akan menyalakan dan mematikan relay di jam yang sama setiap harinya. Mode keempat, mode otomasi deteksi objek yang akan menyalakan relay jika terdeteksi ada objek orang. Tampilan dashboard secara utuh ditampilkan pada Gambar 3.



Gambar 5. Proses pelabelan dan parameter yang dihasilkan

D. Pembuatan Model Deteksi YOLO



Gambar 6. Hasil pembacaan gambar dalam bentuk koordinat dan bounding box

Model deteksi untuk proses learning dibuat menggunakan algoritma YOLO dari *library* ultralytics. Menggunakan dataset yang diambil secara mandiri yang mana sesuai dengan tempat penerapan modul kamera. Untuk *dataset* yang digunakan sebanyak 153 foto yang terbagi sebanyak 108 foto sebagai *train* dengan 45 foto sebagai *value* atau *test* atau dengan kata lain menggunakan rasio 70:30 untuk *learning model*. Berikut untuk contoh dataset yang digunakan dapat dilihat pada Gambar 5.

Dataset tersebut diambil di titik di mana kamera nantinya akan ditempatkan. Hal ini ditujukan agar lingkungan *training* dan *testing* sama. Proses *learning* ini diawali dengan penentuan area objek setiap gambar menggunakan modul *labelImg* untuk memberikan *boundaries box* pada setiap objek orang yang dideteksi. Setiap data gambar akan dilabeli dengan label “people”, hasil label dihasilkan dalam bentuk *.txt*, untuk nantinya menjadi bahan dalam proses *learning*. Label tersebut berisi lima parameter yang diperlukan untuk deteksi objek pada YOLO. Parameter secara berurutan yaitu *x*, *y*, *width*, *height*. Proses pelabelan dan hasil parameter seperti terlihat pada Gambar 5.

Model yang telah dibuat memiliki ekstensi *.pt* (*pytorch*) di mana merupakan hasil proses *learning* dari algoritma YOLO. Model ini dapat dimuat dan dijalankan pada *website* dengan memanfaatkan *library* Flask pada Python. Hal ini dimaksudkan untuk dapat membuat sebuah API integrasi yang dapat memproses gambar yang telah ditangkap ESP32-CAM (yang selanjutnya dikirim pada *server*) untuk dilakukan proses deteksi objek orang. Hasil dari API adalah data JSON untuk dapat dihitung berapa jumlah orang yang dihitung dalam gambar, yang kemudian dihitung pada web untuk kemudian dikirimkan jumlahnya pada *database*. Hasil deteksi dan respon API dengan format json terlihat pada Gambar 6.

III. HASIL DAN PEMBAHASAN

A. Hasil Skenario Deteksi

TABEL 1
HASIL SKENARIO DETEKSI

Skenario Deteksi	Aktivitas	Rata-Rata Waktu Respon
Pengujian <i>relay</i> melalui aplikasi Google Home	Pengguna menekan <i>button</i> pada aplikasi Google Home untuk mematikan/ menyalakan <i>relay</i>	3,468 detik

Pengujian <i>relay</i> menggunakan perintah suara melalui <i>Google Assistant</i>	Pengguna memberikan perintah "Ok Google", kemudian diikuti perintah "Nyalakan/ Matikan <i>Relay</i> "	4,348 sekon
Pengujian <i>relay</i> melalui <i>dashboard website</i> <i>mowny</i>	Pengguna menekan <i>switch</i> pada <i>dashboard website</i>	1,042 sekon
Pengujian deteksi objek untuk mengontrol otomasi saklar (model pretrained YOLOv8)	Alat listrik menyala kemudian orang meninggalkan ruangan dan alat listrik mati. Alat listrik kondisi	20,306 sekon
Pengujian deteksi objek untuk mengontrol otomasi saklar (model sendiri dari dataset ruangan sendiri)	mati kemudian orang memasuki ruangan dan alat listrik menyala	18,444 sekon

Secara berurutan uji rata-rata waktu respon dari tombol/ perintah diberikan sampai *relay* menyala pada *Google Home*, *Google Assistant*, *website mowny*, dan deteksi objek dengan model *pretrained* YOLOv8 serta deteksi objek dengan model sendiri dari *dataset* ruangan adalah 3,468 sekon, 4,348 sekon, 1,042 sekon, dan 20,306 sekon serta 18,444 sekon. Untuk *confussion matrix* dari masing-masing model ditampilkan pada Gambar 7.



Gambar 7. *Confussion Matrix* model sendiri (kiri), *Confussion Matrix* model *pretrained* YOLOv8 (kanan)

Confusion matrix menghitung nilai perbandingan antara nilai sebenarnya dengan nilai prediksi, nilai perbandingan ini kemudian dihitung sesuai rumus untuk mengukur performansi dari model *machine learning* yang dibuat [16]. Untuk nilai sebenarnya ada orang (*actual positive*) dan diprediksi ada orang (*predicted positive*), maka masuk dalam *True Positive*. Untuk nilai sebenarnya ada orang (*actual positive*) dan diprediksi tidak ada orang (*predicted negative*), maka masuk dalam *False Negative*. Untuk nilai sebenarnya tidak ada orang (*actual negative*) dan diprediksi ada orang (*predicted positive*), maka masuk dalam *False Positive*. Untuk nilai sebenarnya tidak ada orang (*actual negative*) dan diprediksi tidak ada orang (*predicted negative*), maka masuk dalam *True Negative*.

B. Hasil Uji Model

TABEL 2
HASIL UJI MODEL

Parameter	Rumus Hitung	Model Sendiri	Model YOLOv8	Keterangan
Akurasi	$(TP + TN)/(TP + FN + FP + TN)$	0,7431	1,00	Perbandingan jumlah prediksi benar dengan jumlah seluruh prediksi
Presisi	$TP/(TP + FP)$	0,7455	1,00	Perbandingan jumlah prediksi benar dengan jumlah prediksi positif
Recall	$TP/(TP + FN)$	0,7455	1,00	Perbandingan jumlah prediksi benar dengan jumlah aktual positif
F1 Score	$2 * presisi * recall/(presisi + recall)$	0,7455	1,00	Perbandingan parameter presisi rendah dan recall tinggi serta sebaliknya

Pengujian untuk model sendiri dilakukan sebanyak 109 kali percobaan dan untuk model YOLOv8 sebanyak 152 kali percobaan. Nilai akurasi, presisi, recall, dan F1 score dari pretrained YOLOv8 dengan sempurna mampu memprediksi objek orang dengan tepat. Sedangkan nilai akurasi, presisi, recall, dan F1 score dari model yang dibuat sendiri memprediksi dengan nilai kesesuaian ~0,7. Hasil pengujian menunjukkan hasil performa pada tiap parameter model YOLOv8, 1,33 kali lebih tinggi jika dibandingkan parameter pada hasil performa model sendiri. Secara umum, jika terdapat sepuluh kali proses deteksi, model yang dibuat sendiri dapat memprediksi benar sebanyak tujuh kali, sedangkan model pretrained dari YOLOv8 memprediksi benar secara keseluruhan.

C. Hasil Uji

Pada tiga skenario pengujian kontrol digital, didapat waktu rata-rata respon paling cepat adalah skenario tiga, yaitu pada *website mowny*. Proses yang terjadi di sini adalah ketika tombol pada *website* ditekan, data langsung di-

update pada *database*. Waktu rata-rata tercepat kedua adalah skenario satu, yaitu pada Google *Home*. Proses yang terjadi adalah ketika tombol ditekan, aplikasi *Sinric.pro* sebagai perantara akan memberikan trigger pada ESP32 untuk melakukan update data pada *database*. Waktu rata-rata tercepat ketiga adalah skenario ketiga, yaitu pada Google *Assistant*, proses yang terjadi adalah ketika perintah suara diberikan, akan mengecek perangkat pada Google *Home*, kemudian ditambah proses yang sama seperti pada skenario satu. Proses tercepat ini (skenario tiga) membutuhkan waktu 1,042 sekon untuk memproses trigger dari *website* sampai *relay* menyalakan alat listrik.

Catatan: Mengapa harus diukur kecepatannya dan dibandingkan? Jika deteksi gerak memberikan waktu respon yang lebih lama, lalu kenapa (so what)? Apakah tidak digunakan? Jika pada akhirnya semua tetap digunakan, mengapa harus diukur kecepatannya dan dibandingkan?

IV. SIMPULAN

Hasil pembuatan modul otomasi alat rumah berbasis *internet of things* dan *deep learning* dapat berjalan dengan baik. Untuk fungsi modul monitoring dapat berjalan dengan baik melalui website *mowny* yang telah dibuat. Untuk fungsi kontrol sendiri yang mana terdiri dari tiga cara (kontrol melalui *website*, *google home*, dan *google assistant*) dapat dilakukan dengan rata-rata waktu dari ketiganya 2,952 sekon. Untuk hasil deteksi objek orang, dari model yang dibuat memiliki nilai akurasi, presisi, *recall*, dan F1 score berturut-turut sebesar 0,7431 0,7455 0,7455 dan 0,7455. Kedua mode ini, kontrol dan otomasi dapat terhubung dengan baik melalui integrasi yang dilakukan, baik melalui *website*, platform *google*, dan dengan deteksi objek orang. Saran untuk penambahan modul ini, apabila ada pengembangan lebih lanjut, ditambahkan terkait integrasi kamera dengan aplikasi Google *Home*, karena saat ini memanfaatkan *platform Sinric Pro* yang ternyata belum bisa memenuhi kebutuhan tersebut, sehingga untuk hasil deteksi dari kamera, ditampilkan melalui website *mowny*.

UCAPAN TERIMAKASIH

Ucapan terima kasih penulis kepada Tuhan Yang Maha Esa atas limpahan rahmat-Nya, kepada orang tua yang selalu memberikan dukungan, serta dosen pembimbing yang telah membantu mengarahkan penulisan artikel ini.

DAFTAR PUSTAKA

- [1] Pemilik Lupa Matikan Listrik Pompa Air, Rumah di Pasuruan Ludes Terbakar. [Online] Available: <https://tugujatim.id/pemilik-lupa-matikan-listrik-pompa-air-rumah-di-pasuruan-ludes-terbakar/> [Accessed: 04-May-2024].
- [2] Lupa Matikan Kipas Angin lalu Korsleting, Rumah Warga di Probolinggo Terbakar. [Online] Available: <https://probolinggo.inews.id/read/317678/lupa-matikan-kipas-angin-lalu-korsleting-rumah-warga-di-probolinggo-terbakar/> [Accessed: 04-May-2024].
- [3] K. U. Ariawan. "Penerapan IoT untuk Sistem Kendali Jarak Jauh Peralatan Listrik Rumah Tangga Berbasis Raspberry Pi," *Jurnal Nasional Pendidikan Teknik Informatika (JANAPATI)*, vol. 9, no. 3, pp. 292, 2020.
- [4] A. B. Lasera and I. H. Wahyudi. "Pengembangan Prototipe Sistem Pengontrolan Daya Listrik berbasis IoT ESP32 pada Smart Home System," *Elinvo (Electronics, Informatics, and Vocational Education)*, vol. 5, no. 2, pp. 112–120, 2020.
- [5] S. Hadi, P. Dewi, R. P. M. D. Labib, and P. D. Widayaka. "Sistem Rumah Pintar Menggunakan Google Assistant dan Blynk Berbasis Internet of Things," *MATRIK: Jurnal Manajemen, Teknik Informatika Dan Rekayasa Komputer*, vol. 21, no. 3, pp. 667–676, 2022.
- [6] B. Rahman and Imelda. "Prototipe Sistem Kontrol Smart Home Berbasis IoT Dengan Metode MQTT Menggunakan Google Assistant," *JURNAL RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, vol. 4, no. 3, pp. 303–310, 2021.
- [7] I. P. S. Yoga, G. Sukadarmika, R. S. Hartati, and Y. Divayana. "Pendeteksi Jumlah Orang pada Sistem Bangunan Pintar Menggunakan Algoritma You Only Look Once," *Majalah Ilmiah Teknologi Elektro*, vol. 22, no. 1, pp. 11, 2023.
- [8] M. D. Furqan and A. Achmad. "Aplikasi Jaga Jarak Dan Penelusuran Kontak Fisik Berbasis IoT Untuk Mencegah Penularan Covid19," *Jurnal Eksitasi Departemen*, vol. 1, no. 2, pp. 12–21, 2022.
- [9] A. M. Avi, M. S. Rana, M. B. Bedar, and M. A. Talukder. "An Android Application and Speech Recognition-Based IoT-Enabled Deployment Using NodeMCU for Elderly Individuals," *Bulletin of Electrical Engineering and Informatics*, vol. 12, no. 5, 2023.
- [10] M. A. Torad, B. Bouallegue, and M. A. Ahmed. "A Voice Controlled Smart Home Automation System Using Artificial Intelligence and Internet of Things," *Telkomnika (Telecommunication Computing Electronics and Control)*, vol. 2, no. 4, pp. 808–816, 2022.
- [11] Z. Wu, K. Qiu, and J. Zhang. "A Smart Microcontroller Architecture for The Internet of Things," *Sensors (Switzerland)*, vol. 20, no. 7, pp. 1–17, 2020.
- [12] Object Detection – Wikipedia. [Online] Available: https://en.wikipedia.org/wiki/Object_detection/ [Accessed: 04-May-2024].
- [13] YOLO - Object Detection - OpenCV Tutorial Documentation 2019. [Online] Available: <https://opencv-tutorial.readthedocs.io/en/latest/yolo/yolo.html/> [Accessed 04-May-2024]
- [14] R-CNN, Fast R-CNN, Faster R-CNN, YOLO — Object Detection Algorithms. [Online] Available: <https://towardsdatascience.com/r-cnn-fast-r-cnn-faster-r-cnn-yolo-object-detection-algorithms-36d53571365e> [Accessed: 04-May-2024]
- [15] YOLO Explained. [Online] Available: <https://medium.com/analytics-vidhya/yolo-explained-5b6f4564f31/> [Accessed: 04-May-2023]
- [16] Understanding Confusion Matrix. [Online] Available: <https://towardsdatascience.com/understanding-confusion-matrix-a9ad42dcfd62> [Accessed 06-May-2024]