

Pengenalan Gesture Tangan Berbasis MediaPipe Hands Untuk Kendali Perangkat Internet of Things

Syahputra Anugrah Rijky^{1*}, Mhd. Basri²

^{1,2} Program Studi Teknologi Informasi, Universitas Muhammadiyah Sumatera Utara

E-mail: putrariijky@gmail.com

ABSTRACT

This study presents a hand gesture recognition system based on MediaPipe Hands as a real-time interface for controlling Internet of Things (IoT) devices. The system uses a camera as the input medium, MediaPipe Hands to extract 21 hand landmarks, and a rule-based classifier to recognize four static gestures: Open Palm, Fist, One Finger, and Two Finger. A gesture stabilization mechanism validates each result before it is converted into a command and published through the Message Queuing Telemetry Transport (MQTT) protocol to an ESP32 microcontroller that drives a relay-controlled lamp and a servo motor representing a door. An experimental, system-engineering approach was applied, and the system was evaluated in terms of recognition success rate, confidence value, frame rate, processing delay, and the success of MQTT communication and device actuation. Open Palm and Fist were recognized in 100% of the trials, while One Finger and Two Finger reached 95.24%, giving an overall success rate of 97.59% over 83 trials. Confidence values ranged from 90.67% to 99.02%, the frame rate was approximately 27.8 FPS, and the recorded delay was 9.12-9.31 ms. All MQTT payloads were received by the ESP32 and executed correctly by the relay and the servo motor. These results indicate that a lightweight rule-based approach can provide a practical contactless control interface for simple IoT devices under the tested conditions.

Keywords: *MediaPipe Hands, Hand Gesture Recognition, Internet of Things, MQTT, ESP32, Rule-Based*

ABSTRAK

Penelitian ini menyajikan sistem pengenalan gesture tangan berbasis MediaPipe Hands sebagai antarmuka real-time untuk mengendalikan perangkat Internet of Things (IoT). Sistem menggunakan kamera sebagai media masukan, MediaPipe Hands untuk mengekstraksi 21 titik landmark tangan, dan klasifikasi berbasis aturan (rule-based) untuk mengenali empat gesture statis, yaitu Open Palm, Fist, One Finger, dan Two Finger. Setiap hasil

Article History

Received: September 2026

Reviewed: Oktober 2026

Published: Oktober 2026

Plagiarism Checker No
237.u873247.fh8374

Copyright : Author
Publish by : Kohesi



This work is licensed
under a [Creative Commons Attribution-NonCommercial 4.0 International License](https://creativecommons.org/licenses/by-nc/4.0/)



klasifikasi divalidasi dengan mekanisme stabilisasi gesture, kemudian dikonversi menjadi perintah dan dikirim melalui protokol Message Queuing Telemetry Transport (MQTT) kepada mikrokontroler ESP32 yang menggerakkan lampu melalui relay dan motor servo sebagai representasi pintu. Penelitian menggunakan pendekatan eksperimental berbasis rekayasa sistem dengan evaluasi pada tingkat keberhasilan pengenalan, nilai confidence, frame rate, waktu respons, serta keberhasilan komunikasi MQTT dan pengendalian perangkat. Hasil pengujian menunjukkan gesture Open Palm dan Fist dikenali dengan tingkat keberhasilan 100%, sedangkan One Finger dan Two Finger sebesar 95,24%, sehingga tingkat keberhasilan keseluruhan mencapai 97,59% dari 83 percobaan. Nilai confidence berada pada rentang 90,67% hingga 99,02%, frame rate sekitar 27,8 FPS, dan waktu respons yang tercatat 9,12-9,31 ms. Seluruh payload MQTT diterima ESP32 dan dieksekusi dengan benar oleh relay maupun motor servo. Hasil tersebut menunjukkan bahwa pendekatan rule-based yang ringan dapat menjadi antarmuka kendali tanpa sentuhan yang praktis bagi perangkat IoT sederhana pada kondisi pengujian yang digunakan.

Kata Kunci: *MediaPipe Hands, Pengenalan Gesture Tangan, Internet of Things, MQTT, ESP32, Rule-Based*

1. PENDAHULUAN

Internet of Things (IoT) memungkinkan berbagai perangkat elektronik saling terhubung melalui jaringan sehingga data dapat dipertukarkan dan perangkat dapat dikendalikan dari jarak jauh. Teknologi ini telah diterapkan pada banyak bidang, antara lain pemantauan tandon air [1], pertanian [2], pemantauan parameter listrik [3], sistem peringatan dini [4], kelas cerdas [5], serta rumah pintar (*smart home*) yang mengendalikan lampu, pintu, dan pendingin ruangan melalui mikrokontroler ESP32 [6].

Seiring meluasnya penerapan IoT, cara pengguna berinteraksi dengan perangkat menjadi aspek yang semakin penting. Pengendalian perangkat elektronik umumnya masih mengandalkan sakelar fisik, *remote control*, atau aplikasi pada *smartphone*. Cara-cara tersebut menuntut kontak fisik atau perangkat tambahan sehingga kurang praktis pada kondisi tertentu, misalnya ketika pengguna sedang beristirahat atau tidak dapat menjangkau sakelar. Interaksi berbasis gerakan tangan (*gesture*) menawarkan alternatif tanpa sentuhan yang lebih alami dan intuitif [7].

Pengenalan gesture dapat dilakukan dengan sensor non-visual, misalnya penginderaan kapasitif nirsentuh [8], maupun dengan *computer vision* berbasis kamera yang kini banyak memanfaatkan arsitektur *deep convolutional neural network* [9]. Pendekatan berbasis kamera tidak memerlukan perangkat yang dikenakan pengguna dan dapat dibangun dari perangkat keras yang mudah diperoleh. Kerangka kerja MediaPipe Hands mampu melacak 21 titik *landmark* tangan



secara *real-time* pada perangkat dengan sumber daya terbatas [10], dan pengembangannya telah diperluas ke pengenalan gesture pada perangkat (*on-device*) [11] termasuk gesture yang dikustomisasi [12].

Sejumlah penelitian telah memanfaatkan pendekatan tersebut. Nur Budiman dkk. [13] mengendalikan volume komputer dengan gesture jari menggunakan OpenCV dan MediaPipe dan melaporkan tingkat keberhasilan 88,89%. Dewangga dkk. [14] dan Yang dkk. [15] menerapkan pengenalan gesture untuk mengendalikan perangkat rumah pintar, sedangkan Maruthai dkk. [16] mengombinasikan ekstraksi titik kunci MediaPipe dengan model jaringan saraf yang dioptimalkan untuk kendali cerdas secara luring. Pendekatan pembelajaran mendalam juga digunakan untuk meningkatkan ketahanan pengenalan, misalnya melalui augmentasi citra [17], optimasi data dan rekonstruksi 3D [18], atau pada aplikasi IoT untuk membantu penyandang tunanetra [19].

Pendekatan berbasis pembelajaran mesin umumnya memerlukan data latih dan proses pelatihan model, sementara pada perangkat IoT sederhana kebutuhan utamanya adalah mengenali sejumlah kecil gesture statis secara cepat, ringan, dan dengan logika keputusan yang mudah dijelaskan. Selain itu, pada penelitian yang ditinjau dalam naskah ini, evaluasi kuantitatif yang mencakup seluruh rantai sistem, mulai dari deteksi gesture, pengiriman perintah melalui protokol komunikasi IoT, hingga aksi aktuator, belum disajikan secara terpadu. Hal ini menjadi celah penelitian yang ingin diisi.

Penelitian ini mengembangkan dan mengevaluasi sistem pengenalan gesture tangan berbasis MediaPipe Hands dengan klasifikasi *rule-based* sebagai antarmuka kendali perangkat IoT. Kontribusi penelitian ini adalah (1) rancangan sistem terintegrasi yang menghubungkan deteksi landmark, klasifikasi tanpa pelatihan model, mekanisme stabilisasi gesture, komunikasi MQTT, dan mikrokontroler ESP32, serta (2) evaluasi kuantitatif atas keberhasilan pengenalan, nilai *confidence*, *frame rate*, waktu respons, dan keberhasilan pengendalian perangkat.

2. METODE PENELITIAN

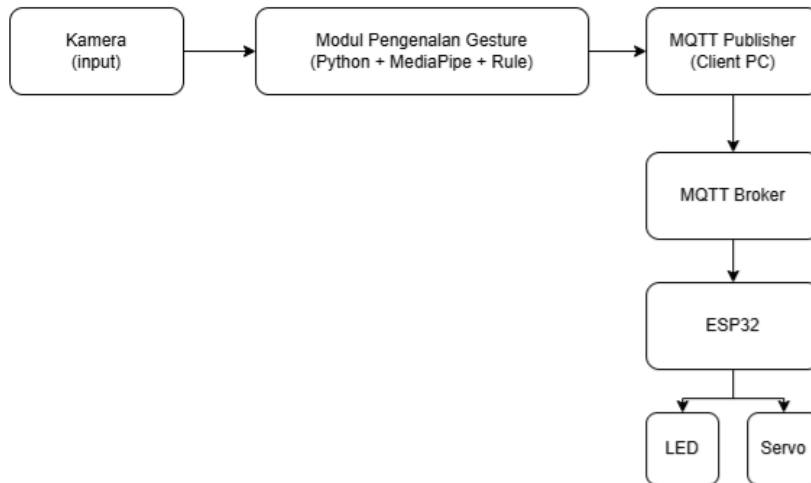
2.1 Rancangan Penelitian

Penelitian ini menggunakan pendekatan eksperimental dengan karakter rekayasa sistem (*engineering research*), yaitu merancang, membangun, dan menguji sistem yang berfungsi secara nyata. Tahapan penelitian meliputi identifikasi kebutuhan sistem, pemilihan kerangka kerja deteksi tangan, perancangan aturan klasifikasi gesture, integrasi dengan sistem IoT melalui protokol MQTT, implementasi perangkat keras dan perangkat lunak, serta pengujian berdasarkan parameter terukur. Penelitian tidak melibatkan pelatihan model sehingga tidak ada pembagian data latih dan data uji.

2.2 Arsitektur Sistem dan Perangkat

Sistem terdiri atas empat modul, yaitu masukan (kamera), pemrosesan gesture (Python, OpenCV, dan MediaPipe Hands), komunikasi (klien MQTT dan *broker*), serta perangkat IoT (ESP32 dengan relay dan motor servo). Aplikasi pada laptop berperan sebagai *publisher* yang mengirim perintah ke *broker* MQTT publik HiveMQ, sedangkan ESP32 berperan sebagai *subscriber* yang menerima

perintah dan menggerakkan aktuator. Pembagian peran ini mengikuti pola arsitektur berlapis IoT, yaitu lapisan persepsi, jaringan, pemrosesan, dan aplikasi [20]. Arsitektur sistem ditunjukkan pada Gambar 1.



Gambar 1. Arsitektur sistem kendali IoT berbasis gesture tangan

Perangkat keras yang digunakan adalah laptop ASUS TUF Gaming F15 (Intel Core i5 generasi ke-11, RAM 16 GB, NVIDIA GeForce RTX 3050), kamera USB 1080p, mikrokontroler ESP32 DevKit V1, modul relay satu kanal yang mengendalikan lampu miniatur AC 220 V, motor servo SG90 yang merepresentasikan mekanisme pintu, dan jaringan WiFi IEEE 802.11 b/g/n. Perangkat lunak yang digunakan adalah Windows 11 (24H2), Python 3.14.3, OpenCV 4.13.0, MediaPipe Hands 0.10.35, Paho MQTT 2.1.0, dan Arduino IDE 2.3.10. Rangkaian perangkat keras ditunjukkan pada Gambar 2.



Gambar 2. Diagram rangkaian perangkat keras sistem

2.3 Pengenalan Gesture

Setiap *frame* yang diambil dari kamera dibaca dengan OpenCV, dikonversi ke ruang warna RGB, lalu diproses MediaPipe Hands untuk memperoleh 21 titik *landmark* tangan. Untuk menjaga keandalan pada pencahayaan rendah, sistem menerapkan peningkatan kualitas citra setelah pengambilan gambar berupa koreksi gamma, peningkatan kontras, *Contrast Limited Adaptive Histogram Equalization* (CLAHE), dan penapis bilateral pada kondisi sangat gelap. Deteksi diabaikan apabila tingkat kecerahan berada di bawah batas minimum yang ditetapkan.

Klasifikasi gesture dilakukan secara *rule-based*, yaitu membandingkan posisi relatif ujung jari terhadap sendi di bawahnya untuk menentukan apakah suatu jari terbuka atau tertutup. Empat gesture statis digunakan dan dipetakan ke perintah perangkat seperti pada Tabel 1. Kombinasi jari lain diklasifikasikan sebagai tidak valid dan tidak menghasilkan perintah.

Tabel 1. Pemetaan gesture terhadap perintah dan aksi perangkat

Gesture	Kriteria landmark (rule-based)	Payload MQTT	Aksi perangkat
Open Palm	Semua ujung jari berada di atas sendi proksimal	LAMP_ON	Relay aktif, lampu menyala
Fist	Semua ujung jari berada di bawah sendi proksimal	LAMP_OFF	Relay nonaktif, lampu mati
One Finger	Telunjuk terbuka, jari lain tertutup	SERVO_A110	Servo membuka pintu (sekitar 90°)
Two Finger	Telunjuk dan jari tengah terbuka, jari lain tertutup	SERVO_A0	Servo menutup pintu

Hasil klasifikasi tidak langsung dikirim sebagai perintah. Mekanisme stabilisasi menyimpan riwayat hasil klasifikasi pada beberapa *frame* terakhir, dan suatu gesture dinyatakan valid hanya



apabila dikenali secara konsisten selama tiga *frame* berturut-turut. Mekanisme ini mengurangi pengiriman perintah akibat gerakan tangan yang terlalu cepat atau gangguan citra.

2.4 Komunikasi MQTT dan Kendali Perangkat

MQTT adalah protokol ringan berbasis model *publish-subscribe* yang menggunakan *broker* sebagai perantara antara pengirim dan penerima pesan [21]. Setelah *gesture* stabil, aplikasi mengonversinya menjadi *payload* teks dan mempublikasikannya ke satu topik pada *broker* dengan pustaka Paho MQTT. Pengiriman dibatasi oleh mekanisme *cooldown* agar perintah tidak terkirim berulang dalam waktu sangat singkat, dan koneksi yang terputus dipulihkan melalui mekanisme *reconnect*. ESP32 yang telah berlangganan pada topik yang sama menerjemahkan *payload* pada fungsi *callback*: LAMP_ON dan LAMP_OFF mengubah keadaan relay, sedangkan *payload* berawalan SERVO_A diikuti nilai sudut yang diteruskan ke fungsi penggerak servo. Nilai 110° pada SERVO_A110 merupakan hasil kalibrasi yang menghasilkan bukaan fisik pintu sekitar 90°.

2.5 Skenario Pengujian dan Evaluasi

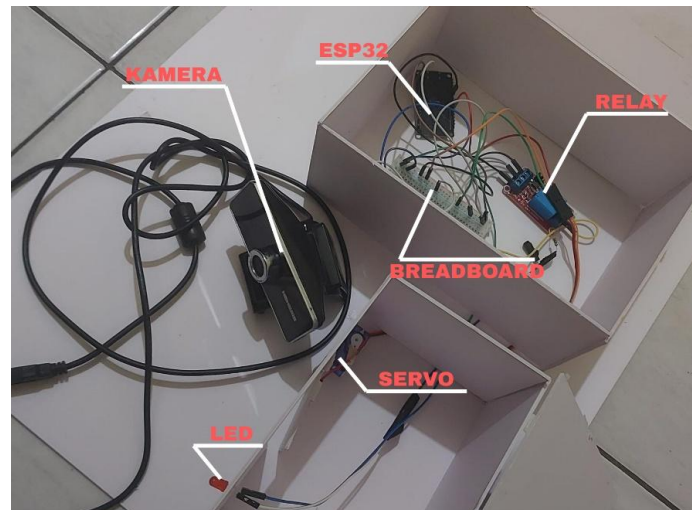
Pengujian dilakukan secara *real-time* dengan menampilkan *gesture* di depan kamera eksternal pada kondisi lingkungan terkontrol. Setiap *gesture* diuji berulang, yaitu 21 percobaan untuk Open Palm, 20 untuk Fist, 21 untuk One Finger, dan 21 untuk Two Finger (total 83 percobaan). Aplikasi mencatat secara otomatis jenis *gesture*, *payload*, nilai *confidence*, FPS, waktu inferensi, waktu respons (*delay*), status komunikasi MQTT, dan status perangkat ke berkas CSV. Parameter yang dievaluasi adalah tingkat keberhasilan pengenalan, nilai *confidence*, FPS, *delay*, serta keberhasilan penerimaan *payload* dan aksi aktuator. Tingkat keberhasilan dihitung dengan Persamaan (1).

$$\text{Tingkat keberhasilan (\%)} = (\text{jumlah gesture dikenali dengan benar} / \text{total percobaan}) \times 100\% \quad (1)$$

3. HASIL DAN PEMBAHASAN

3.1 Implementasi Sistem

Sistem berhasil diimplementasikan sebagai prototipe rumah pintar mini dengan lampu yang dikendalikan relay dan pintu yang digerakkan servo (Gambar 3). Antarmuka aplikasi menampilkan citra kamera, kotak pembatas (*bounding box*) dan 21 titik *landmark* tangan, hasil klasifikasi *gesture*, nilai *confidence*, indikator stabilitas, status koneksi MQTT, status relay dan servo, FPS, serta waktu inferensi. Gambar 4 memperlihatkan keempat *gesture* yang dikenali beserta *payload* yang dihasilkan; pada contoh *gesture* Open Palm, sistem menampilkan *payload* LAMP_ON dengan *confidence* 98% dan indikator stabilitas 3/3 sebelum perintah dikirim.



Gambar 3. Prototipe sistem kendali rumah pintar berbasis gesture tangan



Gambar 4. Tampilan deteksi gesture dan payload yang dihasilkan

3.2 Hasil Pengenalan Gesture

Hasil pengujian pengenalan gesture disajikan pada Tabel 2. Gesture Open Palm dan Fist dikenali pada seluruh percobaan (100%), sedangkan One Finger dan Two Finger masing-masing gagal satu kali dari 21 percobaan (95,24%). Secara keseluruhan, 81 dari 83 percobaan berhasil, sehingga tingkat keberhasilan keseluruhan sebesar 97,59%.

Tabel 2. Hasil pengujian pengenalan gesture

Gesture	Jumlah percobaan	Berhasil	Gagal	Tingkat keberhasilan (%)
Open Palm	21	21	0	100,00
Fist	20	20	0	100,00
One Finger	21	20	1	95,24
Two Finger	21	20	1	95,24
Total	83	81	2	97,59

Open Palm dan Fist dibedakan oleh kondisi semua jari yang berlawanan (seluruhnya terbuka atau seluruhnya tertutup) sehingga aturan geometrisnya paling tegas. One Finger dan Two Finger hanya berbeda pada status jari tengah dan karenanya lebih sensitif terhadap posisi jari yang belum sepenuhnya terbuka atau tertutup. Penyebab dua kegagalan yang terjadi tidak dianalisis secara terpisah dalam pengujian; kemungkinan penyebabnya adalah perubahan posisi jari yang terlalu cepat, sudut tangan terhadap kamera, atau variasi pencahayaan yang memengaruhi deteksi *landmark*.

3.3 Performa Sistem

Performa sistem selama pengenalan berlangsung ditunjukkan pada Tabel 3. Nilai *confidence* berada pada rentang 90,67% hingga 99,02%, FPS berkisar 27,76-27,78, dan *delay* yang dicatat aplikasi berkisar 9,12-9,31 ms. Nilai FPS dan *delay* hampir identik antar gesture, yang menunjukkan bahwa beban komputasi klasifikasi *rule-based* tidak bergantung pada jenis gesture dan sistem berjalan stabil pada perangkat uji.

Tabel 3. Nilai confidence, delay, dan FPS per gesture

Gesture	Confidence (%)	Delay (ms)	FPS
Open Palm	90,67	9,12	27,77
Fist	95,38	9,21	27,78
One Finger	98,16	9,21	27,78
Two Finger	99,02	9,31	27,76

Perlu dicatat bahwa *delay* pada Tabel 3 merupakan waktu respons yang dicatat pada sisi aplikasi. Nilai ini belum mencakup waktu rambat jaringan melalui *broker* publik maupun waktu gerak mekanis aktuator, sehingga tidak dapat ditafsirkan sebagai latensi total dari gesture hingga lampu menyala atau pintu bergerak.

3.4 Komunikasi MQTT dan Pengendalian Perangkat IoT

Pada pengujian komunikasi, setiap gesture yang lolos stabilisasi menghasilkan *payload* yang sesuai rancangan dan diterima ESP32 (Tabel 4). *Payload* LAMP_ON dan LAMP_OFF masing-masing menyalakan dan mematikan lampu melalui relay, sedangkan SERVO_A110 dan SERVO_A0 membuka pintu sekitar 90° dan menutupnya kembali. Hasil ini menunjukkan bahwa rantai sistem dari gesture, MQTT, ESP32, hingga aktuator berfungsi sesuai tujuan penelitian.

Tabel 4. Hasil pengujian komunikasi MQTT dan pengendalian perangkat IoT

Gesture	Payload MQTT	Perangkat	Kondisi perangkat	Hasil
Open Palm	LAMP_ON	Relay	Lampu menyala	Berhasil
Fist	LAMP_OFF	Relay	Lampu mati	Berhasil
One Finger	SERVO_A110	Motor servo	Pintu terbuka sekitar 90°	Berhasil
Two Finger	SERVO_A0	Motor servo	Pintu tertutup	Berhasil



3.5 Pembahasan

Tingkat keberhasilan keseluruhan 97,59% menunjukkan bahwa klasifikasi *rule-based* atas landmark MediaPipe Hands cukup untuk membedakan empat gesture statis pada kondisi pengujian, tanpa data latih maupun pelatihan model. Angka ini lebih tinggi daripada 88,89% yang dilaporkan Nur Budiman dkk. [13] untuk pengendalian volume dengan gesture jari, tetapi kedua penelitian berbeda dalam jumlah gesture, skenario, dan kondisi pengujian sehingga perbandingan tersebut hanya bersifat indikatif. Hasil ini juga sejalan dengan kemampuan MediaPipe Hands untuk berjalan *real-time* pada perangkat dengan sumber daya terbatas [10], yang tercermin pada FPS sekitar 27,8 dengan *delay* tercatat di bawah 10 ms.

Dibandingkan pendekatan pembelajaran mendalam [7], [17], [19], pendekatan *rule-based* lebih ringan, transparan karena setiap gesture memiliki definisi logis yang eksplisit, dan mudah dikalibrasi. Sebagai konsekuensinya, jumlah dan variasi gesture yang dapat dibedakan terbatas pada pola posisi jari yang dapat dinyatakan dengan aturan sederhana. Stabilisasi tiga *frame* dan *cooldown* pengiriman menambah keandalan dengan mencegah perintah akibat gerakan tidak disengaja, sementara pada sisi IoT, pola *publish-subscribe* MQTT memisahkan aplikasi pengenalan dan ESP32 sehingga keduanya tidak perlu terhubung langsung.

Penelitian ini memiliki beberapa keterbatasan. Pertama, jumlah percobaan terbatas (83) dan dilakukan pada kondisi terkontrol, sehingga pengaruh jarak pengguna, posisi kamera, dan variasi pencahayaan belum dievaluasi secara kuantitatif. Kedua, *delay* yang diukur belum mencakup jaringan dan gerak aktuator. Ketiga, komunikasi memakai *broker* publik tanpa autentikasi, enkripsi, atau kontrol akses sehingga belum memadai untuk lingkungan nyata. Keempat, sistem hanya mengenali empat gesture statis.

KESIMPULAN

Sistem pengenalan gesture tangan berbasis MediaPipe Hands dengan klasifikasi *rule-based*, stabilisasi gesture, protokol MQTT, dan ESP32 berhasil dirancang dan diimplementasikan sebagai antarmuka kendali perangkat IoT. Sistem mengenali empat gesture dengan tingkat keberhasilan 100% untuk Open Palm dan Fist serta 95,24% untuk One Finger dan Two Finger (97,59% dari 83 percobaan), dengan *confidence* 90,67%-99,02%, FPS sekitar 27,8, dan *delay* tercatat 9,12-9,31 ms. Seluruh *payload* MQTT diterima ESP32 dan dieksekusi dengan benar oleh relay dan motor servo.

Penelitian selanjutnya disarankan untuk (1) mengevaluasi pengaruh jarak, sudut kamera, intensitas cahaya, latar belakang, dan jumlah pengguna dengan skenario yang lebih luas; (2) memperluas variasi gesture dan membandingkan pendekatan *rule-based* dengan *machine learning* atau *deep learning*; (3) mengukur latensi dari gesture hingga aksi aktuator; (4) menggunakan *broker* privat atau lokal seperti Eclipse Mosquitto dengan autentikasi dan enkripsi TLS; serta (5) menerapkan sistem pada perangkat IoT lain seperti kunci pintar, kipas, atau tirai otomatis.

**DAFTAR PUSTAKA**

- [1] Basri, M., Manurung, A. A., & Hasibuan, A. R. (2024). Internet of Things: Sistem Monitoring Tandon Air Berbasis ESP-8266. *Portal Riset dan Inovasi Sistem Perangkat Lunak*, 2(1), 35-44.
<https://doi.org/10.59696/prinsip.v2i1.72>
- [2] Sari, I. P., Novita, A., Al-Khowarizmi, A.-K., Ramadhani, F., & Satria, A. (2024). Pemanfaatan Internet of Things (IoT) pada Bidang Pertanian Menggunakan Arduino Uno R3. *Blend Sains Jurnal Teknik*, 2(4), 337-343.
<https://doi.org/10.56211/blendsains.v2i4.505>
- [3] Rangkuti, T. H., & Martiano, M. (2025). Design of IoT Based Electrical Parameter Monitoring System using NodeMCU V3 and PZEM-004T V3 Sensor. *Tsabit Journal of Computer Science*, 2(1), 1-8.
<https://doi.org/10.56211/tsabit42>
- [4] 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>
- [5] 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>
- [6] Mubarak, M. A., Fattah, F., & Azis, H. (2025). Prototipe Smart Home Berbasis ESP32 dengan Fitur Keamanan Pintu, Lampu, dan AC Otomatis Berbasis IoT. *LINIER: Literatur Informatika dan Komputer*, 2(3), 421-437.
<https://doi.org/10.33096/linier.v2i3.3152>
- [7] Sen, A., Mishra, T. K., & Dash, R. (2023). Deep learning based hand gesture recognition system and design of a Human-Machine Interface. *arXiv preprint*.
<http://arxiv.org/abs/2207.03112>
- [8] Lee, H., Mandivarapu, J. K., Ogbazghi, N., & Li, Y. (2022). Real-time Interface Control with Motion Gesture Recognition based on Non-contact Capacitive Sensing. *arXiv preprint*.
<http://arxiv.org/abs/2201.01755>
- [9] Khan, A., Sohail, A., Zahoora, U., & Qureshi, A. S. (2020). A survey of the recent architectures of deep convolutional neural networks. *Artificial Intelligence Review*, 53(8), 5455-5516.
<https://doi.org/10.1007/s10462-020-09825-6>
- [10] Zhang, F., Bazarevsky, V., Vakunov, A., Tkachenka, A., Sung, G., Chang, C.-L., & Grundmann, M. (2020). MediaPipe Hands: On-device Real-time Hand Tracking. *arXiv preprint*.
<http://arxiv.org/abs/2006.10214>
- [11] Sung, G., Sokal, K., Uboweja, E., Bazarevsky, V., Baccash, J., Bazavan, E. G., Chang, C.-L., & Grundmann, M. (2021). On-device Real-time Hand Gesture Recognition. *arXiv preprint*.
<http://arxiv.org/abs/2111.00038>
- [12] Uboweja, E., Tian, D., Wang, Q., Kuo, Y.-C., Zou, J., Wang, L., Sung, G., & Grundmann, M. (2023). On-device Real-time Custom Hand Gesture Recognition. *arXiv preprint*.
<http://arxiv.org/abs/2309.10858>
- [13] Nur Budiman, S., Lestanti, S., Marselius Evvandri, S., & Kartika Putri, R. (2022). Pengenalan Gestur Gerakan Jari untuk Mengontrol Volume di Komputer Menggunakan Library OpenCV dan MediaPipe. *Antivirus: Jurnal Ilmiah Teknik Informatika*, 16(2), 223-232.
<https://doi.org/10.35457/antivirus.v16i2.2508>



- [14] Dewangga, S., Subianto, M., & Swastika, W. (2024). Implementation of Hand Gesture Recognition as Smart Home Devices Controller. *Journal of Intelligent System and Computation*, 6(2), 63-68.
<https://doi.org/10.52985/insyst.v6i2.372>
- [15] Yang, C. Y., Lin, Y. N., Wang, S. K., Shen, V. R. L., Tung, Y. C., Shen, F. H. C., & Huang, C. H. (2023). Smart Control of Home Appliances Using Hand Gesture Recognition in an IoT-Enabled System. *Applied Artificial Intelligence*, 37(1).
<https://doi.org/10.1080/08839514.2023.2176607>
- [16] Maruthai, S., S, S., & S, V. (2026). Mediapipe-based keypoint extraction with optimized neural models for offline smart control. *ITM Web of Conferences*, 82, 01011.
<https://doi.org/10.1051/itmconf/20268201011>
- [17] Awaluddin, B. A., Chao, C. T., & Chiou, J. S. (2024). A Hybrid Image Augmentation Technique for User- and Environment-Independent Hand Gesture Recognition Based on Deep Learning. *Mathematics*, 12(9).
<https://doi.org/10.3390/math12091393>
- [18] Mustafa, Z., Nsour, H., & ud din Tahir, S. B. (2023). Hand gesture recognition via deep data optimization and 3D reconstruction. *PeerJ Computer Science*, 9.
<https://doi.org/10.7717/PEERJ-CS.1619>
- [19] Mengash, H. A., Alqadi, B. S., & Marzouk, R. (2025). Enhancing gesture recognition for assisting visually impaired persons using deep learning in an IoT environment-based improved snake optimisation algorithm. *Scientific Reports*, 15(1).
<https://doi.org/10.1038/s41598-025-22070-7>
- [20] Maurya, S. K., Pal, O. P., & Sarvakar, K. (2024). Layered architecture of IoT. In *Secure and Intelligent IoT-Enabled Smart Cities* (pp. 164-194). IGI Global.
<https://doi.org/10.4018/979-8-3693-2373-1.ch009>
- [21] Dizdarevic, J., Michalke, M., & Jukan, A. (2023). Engineering and Experimentally Benchmarking Open Source MQTT Broker Implementations. *arXiv preprint*.
<http://arxiv.org/abs/2305.13893>