

PENGUJIAN PENGENALAN WAJAH REAL-TIME DENGAN DLIB PADA RASPBERRY PI 5

Syarif Husin¹, Iskandar Lutfi², Masayu Anisah³

Teknik Elektro, Jurusan Teknik Elektro – Politeknik Negeri Sriwijaya^{1,2,3}

Email: [user.ucin@gmail.com¹](mailto:user.ucin@gmail.com)

Informasi	Abstract
Volume : 2	<i>This research discusses testing the Raspberry Pi 5-based face recognition system using the dlib library with the Histogram of Oriented Gradients (HOG) method. The main purpose of this test is to evaluate the performance of the system in various lighting and distance conditions. Tests were conducted on two face samples in four scenarios: near-bright, far-bright, near-dark, and far-dark. The results show that the system performs optimally in bright conditions and close distances, but its performance degrades in low lighting and far distances. The accuracy rate obtained is 25% for the first sample and 50% for the second sample. The advantage of this system lies in computational efficiency as the entire process is performed locally on the Raspberry Pi 5 without connection to an external server. The drawback is the sensitivity to variations in lighting and face viewing angle. This research shows that real-time face recognition on edge devices can be done cost-effectively, and further development is recommended through increasing datasets and integration with light sensors for automatic adaptation to environmental conditions.</i>
Nomor : 6	
Bulan : Juni	
Tahun : 2025	
E-ISSN : 3062-9624	

Keywords : Raspberry Pi 5, Face Recognition Real-time, HOG (Histogram of Oriented Gradients), dlib

Abstrak

Penelitian ini membahas pengujian sistem pengenalan wajah berbasis Raspberry Pi 5 menggunakan pustaka dlib dengan metode Histogram of Oriented Gradients (HOG). Tujuan utama pengujian ini adalah mengevaluasi performa sistem dalam berbagai kondisi pencahayaan dan jarak. Pengujian dilakukan terhadap dua sampel wajah dalam empat skenario: dekat-terang, jauh-terang, dekat-gelap, dan jauh-gelap. Hasil menunjukkan bahwa sistem bekerja optimal pada kondisi terang dan jarak dekat, namun performanya menurun pada pencahayaan rendah dan jarak jauh. Tingkat akurasi yang diperoleh adalah 25% untuk sampel pertama dan 50% untuk sampel kedua. Kelebihan sistem ini terletak pada efisiensi komputasi karena seluruh proses dilakukan secara lokal di Raspberry Pi 5 tanpa koneksi ke server eksternal. Kekurangannya adalah sensitivitas terhadap variasi pencahayaan dan sudut pandang wajah. Penelitian ini menunjukkan bahwa pengenalan wajah real-time pada perangkat edge dapat dilakukan secara cost-effective, dan direkomendasikan pengembangan lebih lanjut melalui peningkatan dataset serta integrasi dengan sensor cahaya untuk adaptasi otomatis terhadap kondisi lingkungan.

Kata Kunci : Raspberry Pi 5, Pengenalan Wajah Real-Time, HOG (Histogram of Oriented Gradients), dlib.

A. PENDAHULUAN

Kemajuan pesat dalam teknologi pengenalan wajah telah menjadikannya tulang punggung banyak aplikasi kontemporer, termasuk kehadiran digital, sistem keamanan biometrik, kontrol akses otomatis, dan interaksi manusia-komputer dalam lingkungan cerdas. Pengenalan orang yang cepat tanpa sentuhan fisik atau keterlibatan pengguna dimungkinkan karena sifatnya yang tidak mengganggu, yang merupakan manfaat utamanya [1]. Hal ini memberikan efisiensi operasional yang dapat diskalakan sekaligus meningkatkan kemudahan pengguna. Namun, ada banyak rintangan komputasi yang harus diatasi saat menerapkan sistem pengenalan wajah waktu nyata, khususnya selama fase kategorisasi gambar, penyelarasan landmark, dan deteksi fitur [2]. Saat digunakan pada perangkat komputasi tepi dengan sumber daya terbatas, kompleksitas ini menjadi lebih penting.

Raspberry Pi 5, sebagai perangkat komputasi tepi (*edge device*) generasi terbaru, menawarkan peningkatan kinerja hingga 2-3 kali lipat dibandingkan pendahulunya melalui arsitektur CPU Broadcom BCM2712 quad-core 64-bit dan peningkatan bandwidth GPU [3]. Kemampuan ini membuka peluang untuk menjalankan algoritma computer vision yang sebelumnya hanya feasible pada komputer berdaya tinggi. Namun, optimasi tetap menjadi prasyarat utama mengingat keterbatasan memori (RAM 4-8GB) dan konsumsi daya maksimal 27W yang dimilikinya [4]. Tantangan ini diperparah oleh kebutuhan sistem pengenalan wajah real-time yang memerlukan laju pemrosesan minimal 24-30 FPS (frames per second) untuk mempertahankan fluiditas interaksi [5].

Algoritma pengenalan wajah yang dioptimalkan untuk perangkat berdaya rendah milik Dlib berdasarkan Histogram of Oriented Gradients (HOG) menonjol sebagai opsi yang memungkinkan. Dengan menggunakan pendekatan berbasis fitur dengan kompleksitas algoritma $O(n)$, teknik HOG di dlib lebih cocok untuk perangkat edge daripada pendekatan Convolutional Neural Network (CNN), yang membutuhkan komputasi besar [6]. Frame dip yang disebabkan oleh beban komputasi pada resolusi tinggi dan fluktuasi dalam keadaan sekitar seperti perubahan pencahayaan, oklusi parsial, dan sudut kemiringan wajah tetap menjadi hambatan laten untuk implementasi HOG pada Raspberry Pi 5 [7].

Penelitian ini mengusulkan strategi optimasi holistik untuk sistem pengenalan wajah real-time berbasis dlib pada Raspberry Pi 5 melalui tiga pendekatan utama:

Optimasi Pra-pemrosesan Citra: Mengoptimalkan beban komputasi tanpa mengorbankan akurasi deteksi dengan mengintegrasikan algoritma penskalaan gambar dinamis adaptif berdasarkan jarak subjek ke kamera dan konversi skala abu-abu selektif.

Manajemen Sumber Daya: Dengan menggunakan pemrosesan utas paralel, proses pengambilan gambar, identifikasi wajah, dan ekstraksi fitur dipartisi menjadi utas berbeda yang dijalankan pada inti CPU terpisah.

Adaptasi Dinamis: Mekanisme penyesuaian parameter deteksi HOG (*sliding window step size, scale factor*) secara *runtime* berdasarkan beban komputasi aktual.

Kerangka kerja yang mengintegrasikan pengoptimalan perangkat lunak dengan penggunaan optimal desain perangkat keras Raspberry Pi 5 merupakan kontribusi utama dari studi ini. Dengan penggunaan metrik presisi, recall, dan skor F1, penilaian kinerja menguji akurasi deteksi. Ia juga menganalisis stabilitas sistem dengan memvariasikan kondisi pencahayaan (100-1000 lux), sudut deviasi wajah (0°-90°), dan situasi termasuk beban komputasi multitugas. Kami mengantisipasi bahwa temuan eksperimental akan berfungsi sebagai panduan untuk mengembangkan sistem pengenalan wajah yang terjangkau yang memenuhi persyaratan waktu nyata untuk perangkat edge.

B. TINJAUAN PUSTAKA

Teknologi Pengenalan Wajah

Salah satu subbidang AI dan pemrosesan citra digital, pengenalan wajah berupaya mengidentifikasi individu berdasarkan karakteristik wajah mereka. Mendeteksi wajah, mengekstraksi fitur, dan mengklasifikasikan identitas adalah langkah-langkah umum dalam pendekatan ini. Histogram Gradien Berorientasi (HOG) adalah pendekatan identifikasi wajah populer yang menemukan bentuk atau tepi objek dengan mendeteksi variasi intensitas piksel [8]. Pendekatan HOG ideal untuk penggunaan pada perangkat berbasis tepi seperti Raspberry Pi karena melakukan identifikasi wajah di pustaka dlib dengan kebutuhan komputasi sederhana [6].

Model Convolutional Neural Network (CNN) adalah salah satu opsi pengenalan wajah berbasis pembelajaran mendalam dlib; yang lainnya adalah HOG. Untuk membuat representasi vektor fitur (penyisipan wajah 128D) dari foto wajah, model CNN di dlib telah dilatih menggunakan kumpulan data wajah skala besar [9]. Prosedur untuk mencocokkan dua wajah melibatkan pencarian jarak antara dua vektor penyematan menggunakan metode Euclidean. Jika jaraknya kurang dari ambang batas tertentu, yang dilambangkan sebagai $\theta \backslash \theta$, maka

kedua sisinya dianggap sama. Saat menentukan jarak dalam istilah geometri, rumus standarnya adalah:

$$d(p, q) = \sqrt{\sum_{i=1}^n (P_i - Q_i)^2}$$

dengan p dan q masing-masing merupakan vektor representasi dua wajah yang dibandingkan, dan nnn adalah jumlah dimensi vektor (dalam kasus ini n=128n = 128n=128) [4].

Raspberry Pi 5

Raspberry Pi 5 sebagai platform hardware yang digunakan dalam penelitian ini membawa peningkatan besar dibanding generasi sebelumnya, dengan prosesor quad-core ARM Cortex-A76 dan dukungan RAM hingga 8GB [2]. Peningkatan ini memungkinkan Raspberry Pi 5 untuk menjalankan model pengenalan wajah dengan kinerja real-time, terutama jika didukung dengan optimasi seperti resizing frame, penggunaan mode grayscale, dan pemanfaatan multi-threading. Optimalisasi ini penting mengingat kendala utama di edge computing adalah keterbatasan sumber daya dan kebutuhan respons waktu cepat tanpa ketergantungan pada server eksternal [6].

Dengan kombinasi metode ringan dari *dlib* dan kemampuan pemrosesan dari Raspberry Pi 5, penelitian ini berusaha memaksimalkan kinerja sistem pengenalan wajah real-time untuk kebutuhan praktis di lingkungan edge. Penggunaan metode deteksi cepat dan pengolahan data minim bertujuan untuk mencapai sistem yang stabil, responsif, dan akurat dalam berbagai kondisi pencahayaan dan orientasi wajah



Gambar 1. Raspberry Pi 5

Kamera Webcam

Sumber masukan utama untuk sistem pengenalan wajah waktu nyata adalah perangkat penangkap gambar, itulah sebabnya mengapa hal ini sangat penting. Agar sistem dapat menganalisis foto wajah, kamera digunakan untuk mengambil gambar subjek secara waktu nyata. Untuk penelitian ini, kami menggunakan kamera USB eksternal 720p (1280 x 720 piksel) standar yang kompatibel dengan USB 3.0 dan berfungsi dengan Raspberry Pi 5. Instalasi yang fleksibel, sudut pengambilan gambar yang fleksibel, dan kompatibilitas plug-and-play tanpa memerlukan pengaturan driver tambahan adalah manfaat utama dari penggunaan webcam eksternal sebagai pengganti kamera terintegrasi seperti Pi Camera [10].

Sensor gambar, sering kali CMOS, merekam cahaya yang dipantulkan dari suatu objek dan mengubahnya menjadi keluaran digital; ini adalah teknologi mendasar di balik webcam. Sistem kemudian mengubah sinyal menjadi bingkai gambar yang dapat ditangani oleh algoritma dlib, baik dalam format RGB maupun skala abu-abu. Dalam putaran cepat, bingkai kamera yang diperoleh diproses secara berurutan menggunakan OpenCV, antarmuka pemrosesan gambar, dan dikirim ke memori utama Raspberry Pi. Resolusi, frame rate, dan pencahayaan adalah tiga faktor penting webcam yang secara signifikan memengaruhi kinerja sistem dalam penelitian ini. Untuk mencapai kinerja optimal dalam hal kualitas gambar dan kecepatan pemrosesan, resolusi ditetapkan pada 640 x 480 piksel. Selain itu, untuk mengurangi noise dan memaksimalkan akurasi dalam identifikasi fitur wajah, kondisi pencahayaan selama pengambilan gambar dijaga agar tetap sedang dan berkelanjutan [11].

Secara keseluruhan, pemilihan webcam sebagai sensor input citra memberikan solusi efisien dan ekonomis dalam sistem berbasis Raspberry Pi. Dengan resolusi yang cukup dan dukungan software yang luas, webcam dapat diintegrasikan secara optimal dalam sistem pengenalan wajah berbasis dlib untuk mendukung proses identifikasi real-time.



Gambar 2. Kamera Webcam

C. METODE PENELITIAN

Penulis menggunakan pendekatan penelitian kuantitatif dalam pembuatan produk ini. Menemukan dan mengidentifikasi wajah yang tersimpan di Raspberry Pi 5 merupakan topik utama penelitian ini.

Alat dan komponen yang dibutuhkan

Berikut alat dan komponen yang digunakan pada penelitian ini :

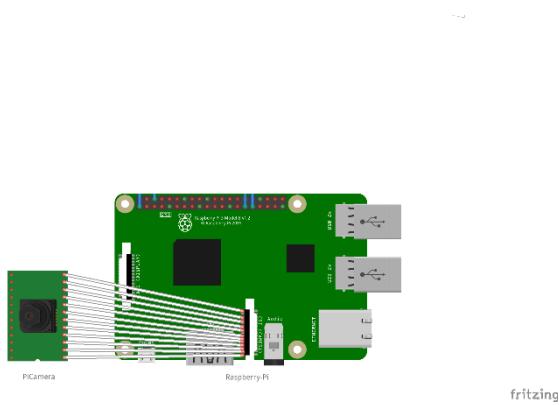
1. Laptop
2. Raspberry Pi 5
3. Kamera Webcam.
4. MicroSd 16GB *class 10*

Sedangkan *software* yang digunakan adalah sebagai berikut pyhton Geany

3.3 *Wiring* diagram rangkaian

Wiring diagram rangkaian alat yang dirancang penulis yang dilampirkan pada gambar

3.1 Seperti berikut



Gambar 3.1 *Wiring* diagram rangkaian

Pada gambar 3.1 dapat kita ketahui bahwa terdapat modul kamera yang akan digunakan untuk mendeteksi serta mengenali wajah yang nantinya akan kita analisa

3.4 Perancangan Sistem *Face recognition*

Pada tahap ini, data citra wajah dikumpulkan menggunakan kamera webcam yang terhubung ke Raspberry Pi 5. Citra wajah disimpan dalam format JPG atau PNG dalam satu folder dataset, dengan struktur per folder berdasarkan nama masing-masing individu. Setiap label (individu) terdiri dari 10 hingga 15 gambar wajah yang diambil dari berbagai sudut dan ekspresi wajah. Proses ini dilakukan dalam pencahayaan cukup agar kualitas citra optimal untuk proses ekstraksi fitur

1. Pengumpulan dataset

Penulis telah mengumpulkan dataset, dalam hal ini telah menyimpan beberapa foto wajah yang nantinya akan diujikan dengan menggunakan metode dlib. Berikut foto-foto wajah yang telah dikumpulkan yang ditunjukkan pada gambar 3.2 berikut :



Gambar 3.2 dataset foto

2. Pemrosesan Citra

Seluruh gambar dalam dataset akan diubah ke dalam mode grayscale untuk mengurangi beban komputasi. Kemudian, dilakukan proses resize citra menjadi resolusi 150x150 piksel agar mempercepat proses pelatihan dan pengenalan. Selain itu, dilakukan pula deteksi wajah awal menggunakan HOG (*Histogram of Oriented Gradients*) sebelum fitur diekstrak.

3. Ekstraksi dan penyandian fitur (Encoding)

Fitur wajah diekstrak menggunakan algoritma face embedding dari *dlib* yang menghasilkan vektor numerik berdimensi 128 untuk setiap wajah. Vektor-vektor inilah yang digunakan dalam proses pembandingan wajah (face matching). Proses ini hanya dilakukan sekali pada saat pelatihan, dan hasil encoding disimpan dalam file .pickle untuk digunakan saat inferensi real-time.

4. Implementasi Real time recognition

Setelah model dikenali, webcam menangkap gambar secara terus-menerus dan mengekstrak wajah dari setiap frame. Kemudian, vektor wajah input dibandingkan dengan database vektor encoding yang telah disimpan. Jika jarak Euclidean antara vektor input dan database lebih kecil dari ambang batas tertentu, maka wajah dianggap cocok dan ditampilkan hasil identifikasinya.

5. Arsitektur model dlib (resnet-based)

Tabel 3.1 arsitektur dlib (resnet-based)

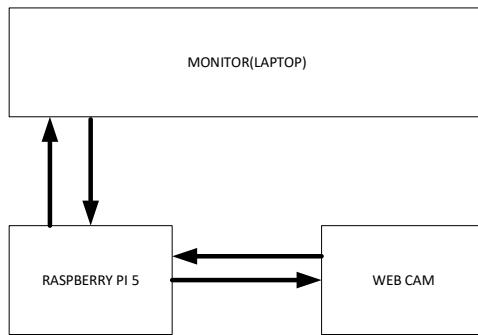
Layer	Detail
Input layer	Input(shape=(150, 150, 3))

Conv layer 1	Conv(64, 7x7), stride=2, padding=3
Max pooling	MaxPool(3x3), stride=2
Residual Block x3	3 x BasicBlock (64)
Residual Block x4	4 x BasicBlock(128)
Residual Block x6	6 x BasicBlock(256)
Average pooling	Global Average Pooling
Fully conected	Fully Connected Layer (output: 128-dimensions)
L2 Normalization	Normalize embedding vector to unit length

Library dlib menggunakan model Convolutional Neural Network (CNN) berbasis ResNet (Residual Network) yang telah dilatih sebelumnya pada dataset wajah yang sangat besar bernama "FaceNet-style dataset" untuk menghasilkan 128-dimensi face embedding vector. Model ini memanfaatkan pendekatan deep metric learning, di mana dua wajah yang serupa akan memiliki representasi vektor yang dekat dalam ruang vektor Euclidean, dan sebaliknya.

Arsitektur yang digunakan oleh dlib memiliki 29 konvolusi, 1 fully connected layer, dan beberapa residual block, serta menghasilkan embedding akhir berdimensi 128.

3.5 Diagram Blok



Gambar 3.3 Blok Diagram

(sumber : *Dokumentasi, 2025*)

Penjelasan Diagram blok pada Gambar 3.3 menggambarkan hubungan antar perangkat keras yang digunakan dalam sistem *face recognition* berbasis Raspberry Pi 5. Sistem ini dirancang untuk mengenali wajah secara real-time menggunakan kamera webcam, dengan pemrosesan utama dilakukan oleh Raspberry Pi 5, serta hasilnya ditampilkan melalui monitor (laptop) yang terhubung melalui jaringan lokal. Berikut adalah uraian dari tiap komponen:

1. Webcam

Webcam berfungsi sebagai sensor input yang menangkap citra wajah secara langsung dari lingkungan. Kamera ini terhubung dengan Raspberry Pi 5 melalui port USB dan mengirimkan video secara real-time untuk diproses.

2. Raspberry Pi 5

Raspberry Pi 5 bertindak sebagai pusat pemrosesan. Di dalamnya telah dipasang program berbasis Python menggunakan pustaka *dlib* untuk mendeteksi serta mengenali wajah. Raspberry Pi menerima input dari webcam, melakukan deteksi serta ekstraksi fitur wajah, dan menentukan identitas berdasarkan model deep learning yang telah dilatih sebelumnya. Seluruh pemrosesan dilakukan secara lokal tanpa koneksi internet.

3. Monitor (laptop)

Monitor atau laptop digunakan untuk menampilkan hasil deteksi wajah. Raspberry Pi mengirimkan hasil tersebut melalui jaringan lokal (LAN atau WiFi) menggunakan protokol remote desktop seperti VNC, SSH X11 forwarding, atau web server lokal. Dengan pendekatan

ini, tampilan antarmuka pengguna dapat diakses tanpa perlu sambungan HDMI fisik, meningkatkan fleksibilitas sistem.

D. HASIL DAN PEMBAHASAN

Pengujian Alat

Pengujian alat ini bertujuan untuk mengamati dan menganalisis performa dari kinerja face recognition menggunakan librarry dlib menggunakan metode HOG atau CNN

Penulis menguji alat dengan menggunakan 2 data sampel. Untuk mengetahui tingkat keberhasilan pengujian kedua sampel tersebut digunakan perhitungan akurasi sampel dengan rumus berikut

$$\text{Akurasi sampel} = \frac{\text{Pengujian berhasil}}{\text{Total Pengujian}} \times 100 = \%$$

Sampel Advent

Tabel 4.1 Sampel Advent

No	Citra	Kondisi		Ket
		Jarak	Cahaya	
1		Dekat	Terang	Berhasil
2		Jauh	Terang	Tidak berhasil

3		Dekat	Gelap	Tidak Berhasil
4		Dekat	Gelap	Berhasil

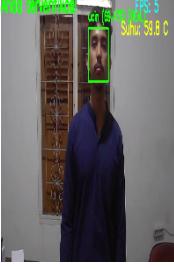
Dapat kita ketahui pengujian sampel advent pada tabel 4.1 mengalami 4 kondisi yaitu dekat-terang, jauh-terang, dekat-gelap, dan jauh-gelap, yang dimana kondisi-kondisi tersebut mempengaruhi untuk melakukan pengenalan wajah.

Berikut perhitungan akurasi untuk sampel advent

$$\text{Akurasi advent} = \frac{1}{4} \times 100 = 25\%$$

Sampel Husin.

Tabel 4.2 Sampel Husin

No	Citra	Kondisi		Ket
		Jarak	Cahaya	
1		Dekat	Terang	Berhasil
2		Jauh	Terang	Tidak berhasil
3		Dekat	Gelap	Berhasil
4		Jauh	Gelap	Tidak berhasil

Dapat kita ketahui pengujian sampel husin pada tabel 4.2 mengalami 4 kondisi yaitu dekat-terang, jauh-terang, dekat-gelap, dan jauh-gelap, yang dimana kondisi kondisi tersebut mempengaruhi untuk melakukan pengenalan wajah.

Berikut perhitungan akurasi untuk sampel Husin

$$\text{Akurasi Husin} = \frac{2}{4} \times 100 = 50\%$$

Analisa Hasil Pengujian

Jika penulis memperoleh angka akurasi yang lebih tinggi dari ambang batas toleransi keberhasilan dan hasil prediksi untuk sistem pengenalan wajah memberikan identitas yang sesuai dengan data pelatihan, maka penulis menetapkan parameter keberhasilan. Dua sampel wajah, Husin dan Advent, digunakan dalam pengujian ini. Setiap sampel diperiksa dalam empat kondisi pencahayaan dan jarak yang berbeda: hampir gelap, sangat gelap, hampir terang, dan sangat terang.

Hanya dalam cahaya yang sangat kuat sistem mampu mengidentifikasi wajah secara akurat dalam sampel Advent; dalam semua tingkat pencahayaan lainnya, sistem gagal total. Artinya, pencahayaan dan jarak dari subjek ke wajah memiliki dampak yang signifikan terhadap kemampuan sistem. Berdasarkan temuan ini, akurasi identifikasi sampel Advent adalah:

$$\text{Akurasi Advent} = \frac{1}{4} \times 100 = 25\%$$

Pada sampel Husin, sistem berhasil mengenali wajah pada kondisi dekat-terang dan dekat-gelap, namun gagal pada kondisi jauh-terang dan jauh-gelap. Artinya, pengaruh jarak menjadi faktor penting terhadap performa face recognition menggunakan library dlib, terutama pada perangkat embedded seperti Raspberry Pi 5 yang memiliki keterbatasan sumber daya. Akurasi pengenalan untuk sampel Husin adalah:

$$\text{Akurasi Husin} = \frac{2}{4} \times 100 = 50\%$$

Berdasarkan kedua pengujian tersebut, maka akurasi rata-rata sistem dapat dihitung dengan rumus:

$$\text{Akurasi Total} = \frac{25+50}{2} = 37.5\%$$

Temuan ini menunjukkan bahwa sistem masih memiliki ruang untuk perbaikan dalam menghadapi skenario dunia nyata, khususnya yang melibatkan cahaya redup dan jarak yang jauh. Kuantitas foto, variasi posisi, dan kualitas pencahayaan dalam set data pelatihan merupakan faktor-faktor yang dapat berkontribusi terhadap masalah ini.

Meskipun HOG lebih ringan dan lebih cepat diproses oleh Raspberry Pi, pendekatan ini tidak seakurat pendekatan CNN saat digunakan pada dlib, khususnya dalam cahaya redup atau dengan gambar yang kabur. Oleh karena itu, pemilihan model deteksi dan penambahan data pelatihan dengan variasi kondisi sangat disarankan untuk meningkatkan generalisasi model dan menghindari permasalahan seperti *overfitting* atau *miss-classification*.

Temuan ini sejalan dengan penelitian sebelumnya yang menyatakan bahwa variasi jarak, pencahayaan, dan sudut pandang merupakan tantangan utama dalam pengenalan wajah [3].

E. KESIMPULAN

Sistem deteksi wajah berdasarkan teknik Histogram of Oriented Gradients (HOG) dan pustaka Dlib dibuat dan diimplementasikan dalam penelitian ini. Sistem ini berjalan pada Raspberry Pi 5. Karena seluruh proses dijalankan secara lokal di telepon pintar, sistem ini dapat mendeteksi wajah bahkan saat tidak ada koneksi internet. Temuan ini didasarkan pada empat keadaan pencahayaan dan jarak berbeda yang dinilai pada dua sampel data, Advent dan Husin. Dengan akurasi rata-rata 37,5% dan tingkat presisi 25% untuk sampel Advent dan 50% untuk sampel Husin, temuan ini menunjukkan bahwa sistem bekerja paling baik dalam pengaturan jarak dekat yang terang.

Karena desain metode HOG yang ringan dan kesesuaianya untuk perangkat dengan perangkat keras rendah seperti Raspberry Pi, sistem ini memiliki manfaat karena dapat berjalan secara independen tanpa koneksi server eksternal. Selain itu, sistem ini meningkatkan efisiensi proses. Ketidakmampuan sistem untuk melakukan generalisasi ke wajah-wajah baru atau situasi ekstrem seperti kegelapan atau pose membungkuk disebabkan, sebagian, oleh sensitivitasnya terhadap kondisi pencahayaan dan jarak objek wajah, dan sebagian lagi, oleh terbatasnya kuantitas dan variasi data pelatihan.

F. DAFTAR PUSTAKA

- Ahmad, S., Rahman, M. A., & Junaidi, E. (2022). Comparative Study of Face Recognition Libraries on Edge Devices. *IEEE Access*, 10, 12451–12460.
- Raspberry Pi Foundation. (2023). Raspberry Pi 5 Hardware Overview. <https://www.raspberrypi.com/>
- Sengupta, S., et al. (2020). "Lightweight Face Recognition on Edge Devices: A Comparative Study of Dlib, OpenCV, and MobileNet." *IEEE International Conference on Embedded Systems (ICES)*, 1-6.

- Raspberry Pi Foundation. (2023). "Raspberry Pi 5 Documentation: Technical Specifications." [Online]. from : <https://www.raspberrypi.com/documentation/>
- Patel, K., & Smith, R. (2020). "Real-Time Computer Vision: From Algorithms to Applications." IEEE Transactions on Pattern Analysis and Machine Intelligence, 42(6), 1289-1305.
- King, D. E. (2015). "Dlib-ml: A Machine Learning Toolkit." Journal of Machine Learning Research, 16, 1239-1244.
- Sze, V., Chen, Y., Yang, T., & Emer, J. (2017). "Efficient Processing of Deep Neural Networks: A Tutorial and Survey." Proceedings of the IEEE, 105(12), 2295-2329.
- Dalal, N., & Triggs, B. (2005). Histograms of Oriented Gradients for Human Detection. IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), vol. 1, pp. 886–893
- Parkhi, O. M., Vedaldi, A., & Zisserman, A. (2015). Deep Face Recognition. British Machine Vision Conference (BMVC).
- Pandey, S., & Gupta, A. (2021). Comparative Performance Evaluation of USB and Pi Cameras for Edge AI Applications. Journal of Embedded Systems and Applications, 8(2), 55–61.
- Ghosh, T., & Ali, M. (2020). Optimization of Face Recognition on Low-Cost Embedded Systems. International Journal of Computer Applications, 177(38), 1–6.