

Rancang Bangun Daftar Pencarian Orang Berbasis CCTV dengan Face Recognition

Akmal Satrio Nugraha
Universitas Telkom
Fakultas Ilmu Terapan
Teknik Telekomunikasi
Jakarta, Indonesia

akmalsatrio@student.telkomuniversity.ac.id

Suyatno
Universitas Telkom
Fakultas Ilmu Terapan
Teknik Telekomunikasi
Jakarta, Indonesia

suyatnobudiharjo@telkomuniversity.ac.id

Yus Natali
Universitas Telkom
Fakultas Ilmu Terapan
Teknik Telekomunikasi
Jakarta, Indonesia

yusnatali@telkomuniversity.ac.id

Abstrak — Face Recognition adalah teknologi yang digunakan untuk mengidentifikasi dan memverifikasi identitas seseorang berdasarkan ciri-ciri wajah. Proses ini melibatkan akuisisi gambar wajah, ekstraksi fitur wajah, dan perbandingan dengan data yang ada dalam basis data. Metode ini telah menjadi fokus penelitian yang penting dalam bidang pengenalan pola dan kecerdasan buatan. Penggunaan teknik-teknik seperti deteksi wajah, ekstraksi fitur, dan algoritma pembelajaran mendalam seperti Convolutional Neural Networks (CNN) telah menghasilkan kemajuan signifikan dalam meningkatkan akurasi pengenalan wajah. Aplikasi dari face recognition mencakup keamanan, pengenalan emosi, pengawasan, dan otentikasi perangkat. Meskipun masih ada tantangan seperti variasi pose, pencahayaan, dan usia, perkembangan teknologi ini terus meningkatkan kinerja dan relevansinya dalam berbagai bidang. Saat aplikasi ini memproses video dan menemukan orang yang dicari, hasilnya akan dimasukkan ke dalam logs TXT sebagai output dari proses tersebut.

Kata kunci— face recognition, python

I. PENDAHULUAN

Dalam beberapa tahun terakhir, teknologi pengenalan wajah telah membawa dampak positif bagi kehidupan manusia. Salah satu bidang yang banyak mendapatkan manfaat dari teknologi ini adalah keamanan rumah atau security home. Face recognition menjadi alternatif menarik untuk menggantikan metode keamanan tradisional seperti kunci dan kartu akses.

Metode pengenalan wajah telah berkembang dari waktu ke waktu dan sekarang tampil di tingkat manusia. Terlepas dari kenyataan bahwa evaluasi metode ini bergantung pada dataset yang ditangkap di alam liar, evaluasi ini tidak mengatasi tantangan seperti oklusi karena dataset ini sebagian besar terdiri dari gambar yang jelas[1].

Pengenalan wajah telah menjadi topik penelitian hangat dalam visi komputer selama bertahun-tahun. Baru-baru ini, metode berbasis pembelajaran mendalam telah mencapai kinerja yang sangat baik, bahkan melampaui manusia dalam beberapa skenario, dengan memberdayakan model pengenalan wajah dengan jaringan saraf yang dalam.

Kebijakan tradisional adalah melatih model pengenalan wajah untuk meningkatkan kekompakan intra-kelas dengan sejumlah besar data dan metrik berbasis margin untuk meningkatkan kinerja pengenalan[2].

Sistem pengenalan wajah (Face Recognition/FR) menyebar ke seluruh dunia dan memiliki efek yang semakin besar pada kehidupan kita sehari-hari. Karena sistem ini semakin terlibat dalam proses pengambilan keputusan kritis, seperti dalam forensik dan penegakan hukum, ada kebutuhan yang semakin besar untuk membuat proses FR dapat dijelaskan kepada manusia. Terutama di lingkungan yang tidak dibatasi, sistem FR harus berurusan dengan variabilitas yang besar, seperti kondisi akuisisi citra (iluminasi, latar belakang) dan faktor wajah (pose, oklusi, ekspresi), yang dapat mengakibatkan keputusan pencocokan yang cacat. Dampak variabilitas ini pada kinerja FR diukur dalam bentuk kualitas gambar wajah (FIQ)[3].

Dalam beberapa tahun terakhir, sistem aplikasi pengenalan wajah telah berkembang pesat sebagai teknologi keamanan komputer di dunia, terutama saat ini, ketika kegiatan teroris merajalela, teknologi ini semakin mendapat perhatian. Teknologi pengenalan wajah memiliki banyak aplikasi khas di bidang keamanan publik, ekonomi sipil, dan hiburan rumah[4].

Metode pengenalan wajah canggih saat ini didasarkan pada model pembelajaran mendalam, yang telah mampu mencapai kinerja yang mengesankan pada tolak ukur publik. Namun, pengenalan wajah dalam tidak berfungsi dengan baik dalam kondisi baru yang menantang, seperti oklusi yang disebabkan oleh wajah bertopeng. Oklusi wajah telah banyak dibahas oleh komunitas riset dalam lingkup solusi pengenalan wajah[5].

II. KAJIAN TEORI

Dalam penelitian ini terdapat beberapa teori yang dikaji untuk memncapai tujuan penelitian, diantaranya teori mengenai face recognition, Haar Cascade Frontalface, python dan library python yang digunakan.

A. Face Recognition

Pengenalan wajah adalah topik penelitian yang dinamis dalam visi komputer, menarik minat peneliti dengan hasil menjanjikan menggunakan metode konvensional dan deep learning, terutama CNN. Meskipun kemajuan telah dicapai dalam lingkungan terkendali maupun tidak, variasi pose, pencahayaan, dan ekspresi wajah tetap menjadi tantangan. Artikel ini membahas rancangan sistem absensi berbasis pengenalan wajah untuk pengolahan video real-time, dengan fokus pada akurasi, stabilitas, keamanan, dan antarmuka. Pendekatan deep learning telah memberikan hasil luar biasa dalam tugas pengenalan wajah, yang melibatkan deteksi wajah, penyetoran, dan representasi wajah. Deteksi wajah, tahap pertama, penting untuk mengidentifikasi posisi wajah dengan koordinat kotak pembatas dan skor kepercayaan. Tahap kedua, penyetoran wajah, normalisasi gambar untuk proses selanjutnya. Berbagai metode deteksi wajah telah diuji dalam situasi dunia nyata, menunjukkan hasil yang bervariasi.

B. Haar Cascade Frontalface

Haar-like features adalah pola piksel yang digunakan untuk mengenali fitur-fitur tertentu dalam gambar, seperti tepi, sudut, dan variasi intensitas. Dalam deteksi wajah, pola ini dirancang untuk mendeteksi perbedaan intensitas antara area mata, hidung, dan mulut dengan area sekitarnya.

C. Python

Python merupakan bahasa pemrograman tingkat tinggi yang digunakan secara luas. Bahasa ini dikembangkan oleh Guido van Rossum pada tahun 1991 dan memiliki filosofi desain yang mengutamakan keterbacaan kode. Python mendukung berbagai paradigma pemrograman, termasuk berorientasi objek, imperatif, fungsional, dan prosedural. Selain itu, Python memiliki standar yang luas dan perpustakaan yang komprehensif. Versi pertamanya adalah Python 1.0 yang dirilis pada tahun 1991, diikuti oleh Python 2.0 pada tahun 2000 dan Python 3.0 pada tahun 2008. Saat ini, versi terbaru yang tersedia adalah Python 3.7. Python merupakan pilihan yang sangat baik bagi peneliti di berbagai bidang ilmiah dan komunitas karena memiliki berbagai keunggulan.

D. Library Python

Penggunaan bahasa pemrograman dalam pengembangan program Face Recognition dengan library seperti dlib, Face Recognition, numpy, dan OpenCV sangat penting dan dipilih berdasarkan pertimbangan tertentu. Bahasa pemrograman yang digunakan dalam kasus ini adalah Python. penggunaan bahasa Python dalam pengembangan program Face Recognition dengan library dlib, Face Recognition, Numpy, dan OpenCV adalah pilihan yang masuk akal karena kombinasi kejelasan kode, ekosistem library yang kuat, dukungan komunitas, dan kemampuan integrasi yang baik yang dimilikinya. Kombinasi ini memungkinkan penulis untuk fokus pada masalah pengenalan wajah yang sebenarnya tanpa terlalu banyak terhambat oleh masalah teknis dalam bahasa pemrograman.

III. METODE

Untuk membuat aplikasi peneteksi wajah, ada beberapa metode yang diterapkan dalam aplikasi. Diantaranya:

A. Deteksi Wajah (Face Detection)

Kode ini menggunakan metode deteksi wajah dengan mengimplementasikan algoritma Haar Cascade Classifier yang disediakan oleh OpenCV. Algoritma ini bekerja dengan mendeteksi pola wajah berdasarkan beberapa fitur seperti mata, hidung, dan mulut. Kode menggunakan berkas XML yang berisi model pelatihan untuk mendeteksi wajah yang disebut 'haarcascade_frontalface_default.xml'. Selanjutnya, kode membaca video input frame per frame, mengubah setiap frame menjadi citra keabuan (grayscale), dan kemudian mencari wajah dalam citra tersebut dengan menggunakan metode detectMultiScale.

B. Pengenalan Wajah (Face Recognition)

Program juga melakukan pengenalan wajah dengan mencocokkan wajah yang terdeteksi dengan wajah referensi yang telah disimpan sebelumnya dalam folder 'Face_Data'. Untuk melakukan ini, kode menggunakan metode pencocokan template (template matching) dengan algoritma TM_CCOEFF_NORMED. Kode membandingkan wajah yang terdeteksi dengan setiap wajah referensi dalam citra keabuan dan jika kesamaan antara wajah terdeteksi dan wajah referensi melebihi ambang batas tertentu (threshold), maka informasi pengenalan wajah tersebut dicatat dalam file 'logs.txt'.

C. Tampilan GUI (Graphic User Interface)

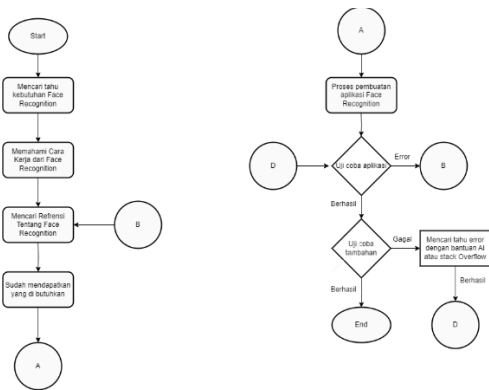
Untuk memberikan antarmuka yang lebih baik kepada pengguna, program menggunakan modul tkinter untuk membuat jendela progres bar (progress bar) yang menunjukkan kemajuan proses deteksi dan pengenalan wajah pada video. Jendela ini juga digunakan untuk menampilkan pesan "Well done!" setelah proses selesai.

Selain itu, aplikasi yang dibuat memiliki blok diagram alur agar program yang dibuat sesuai dengan fungsinya. Berikut adalah blok diagram alur aplikasi:



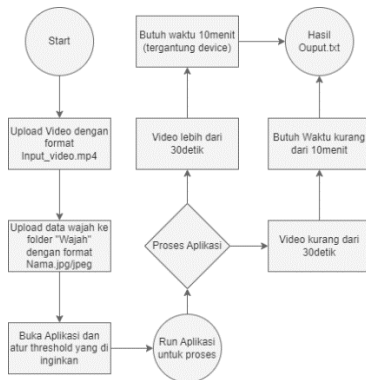
Gambar 1
(Alur proses berjalannya aplikasi)

Gambar 1 menjelaskan tentang proses dari awal berjalannya program Face Recognition, dimulai dari menjalankan aplikasi, kemudian mendeteksi wajah atau tidak mendeteksi wajah, setelah itu keluar dari program dan hasil dari program tersebut tersimpan di dalam folder yang sudah ditempatkan.



Gambar 2
(Alur proses pembuatan sistem aplikasi)

Gambar 2 menjelaskan tentang proses pembuatan system Face Recognition secara bertahap menggunakan bahasa pemrograman Python. Pada tahap awal menjelaskan tentang mencari referensi dan mengetahui tentang Face Recognition dan cara kerja Face Recognition itu sendiri, pada tahap kedua menjelaskan saat uji coba program yang sedang di buat sampai pada tahap selesai.



Gambar 3
(Cara kerja face recognition)

Gambar 3 menjelaskan cara kerja sistem Face Recognition setelah sistem Face Recognition yang menggunakan Bahasa python. Sebelum menjalankan aplikasi upload terlebih dahulu video dan data wajah yang dibutuhkan setelah di upload video dan wajah, buka aplikasi dan atur threshold yang diinginkan setelah mengatur threshold jalankan aplikasi, dan proses waktu yang dibutuhkan tergantung pada durasi video kemudian hasil akan berupa output.txt

IV. HASIL DAN PEMBAHASAN

Setelah aplikasi dibuat, penulis melakukan beberapa percobaan. Berikut adalah

A. Pengujian Tingkat Akurasi

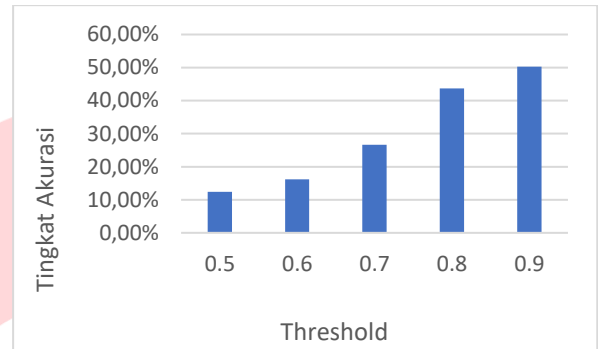
Pada pengujian Tingkat Akurasi, disini penulis menguji berdasarkan penggunaan Threshold mulai dari 0.5 sampai 0.9, dimana dari hasil yang di keluarkan akan di hitung sejauh mana Akurasi pada Software Face Recognition tersebut.

Pada table di atas bisa di lihat semua pengujian menggunakan durasi video yang sama dan video yang sama,

agar pengujian pada tingkat akurasi dari sebuah program dapat diketahui.

Pada perhitungan yang digunakan, penulis mengambil data dari logs yang dibuat otomatis oleh program Face Recognition, data yang di ambil berdasarkan jumlah data yang benar dan data yang salah, kemudian di hitung jumlah yang benar dan akan mendapatkan total akurasi.

Pengujian yang dilakukan untuk tingkat akurasi di uji pada dalam ruangan dengan cahaya yang normal (Ada 2 Lampu di dalam ruangan), dan pada pengujian akurasi menggunakan video resolusi 720p dengan 30fps.



Gambar 4
(Diagram pengaruh Threshold terhadap akurasi)

B. Pengujian pencahayaan

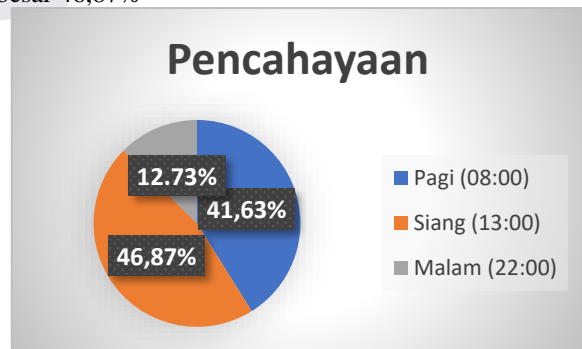
Pengujian pada tingkat pencahayaan, disini penulis mengambil data dari kondisi yang berbeda yaitu dari pagi, siang, dan malam. Penulis juga menggunakan durasi video yang sama agar datanya dapat mudah di ambil oleh penulis.

Pada pagi hari program mendapatkan akurasi sebesar 41,63% dimana jumlah akurasi yang di dapatkan cukup ideal dikarenakan cahaya yang tidak begitu terlalu cerah.

Pada siang hari yaitu tepat jam 13:00PM tingkat akurasi di dapatkan sebesar 46,87% dimana cuacanya sangat cerah dan jelas.

Kemudian pada malam hari tepat jam 22:00PM mendapatkan tingkat akurasi yang tidak begitu bagus dari pagi dan siang, dikarenakan tidak adanya sinar cahaya yang cukup saat pengujian di luar ruangan tidak seperti pagi dan siang hari.

Penulis akan menjelaskan kenapa akurasi pada siang hari sangat besar di bandingkan dengan pagi dan malam, dikarenakan siang hari memiliki pencahayaan yang cukup maka dari itu pada siang hari mendapatkan tingkat akurasi sebesar 46,87%



Gambar 5
(Pengaruh pencahayaan terhadap akurasi)

C. Pengujian Face Recognition menggunakan Aksesoris

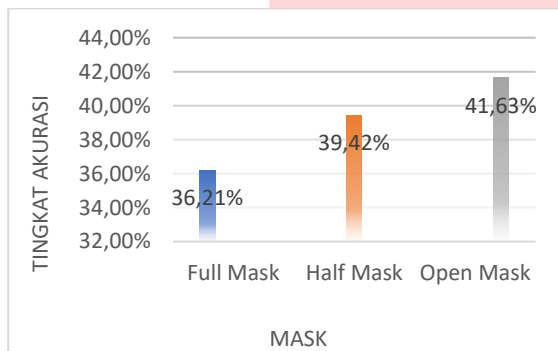
Pada pengujian Face Recognition, penulis menguji beberapa hal yaitu Menggunakan masker, topi, dan kacamata.

Pada pengujian masker untuk Full Mask, masker digunakan dengan Full Mask, yang hanya terlihat mata sampai jidat kepala saja, dan dari hasil yang di dapatkan sebesar 36,21%.

Pada pengujian Half Mask, masker yang digunakan hanya sampai hidung, yang dimana hidung, mata, dan Jidat terlihat dengan jelas, disini tingkat akurasi yang di dapatkan sebesar 48,62%.

Pada Open Mask, dimana masker hanya digunakan di bawah dagu dan wajah akan terlihat dengan jelas, disini program mendapatkan akurasi sebesar 51,11%.

Pada pengujian penggunaan Masker, penulis menggunakan tiga tahapan yaitu Full Mask, Half Mask, Open Mask. Dimana pada tiga tahapan tersebut akan di coba tingkat akurasinya menggunakan threshold yang sama yaitu sebesar 0.8

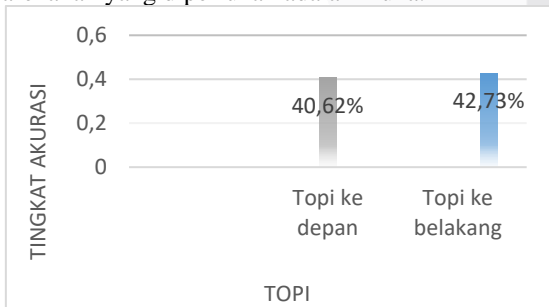


Gambar 6

(Pengaruh Aksesoris masker terhadap akurasi)

Pada pengujian topi, tingkat akurasi yang di dapatkan sebesar 50,23%. Dari hasil akurasi yang di dapat topi tidak terlalu begitu berpengaruh untuk pendeteksi wajah dikarenakan yang diperlukan adalah muka.

Pada pengujian topi, tingkat akurasi yang didapatkan sebesar 50,11%. Dari hasil akurasi yang di dapat topi tidak terlalu begitu berpengaruh untuk pendeteksi wajah dikarenakan yang diperlukan adalah muka.

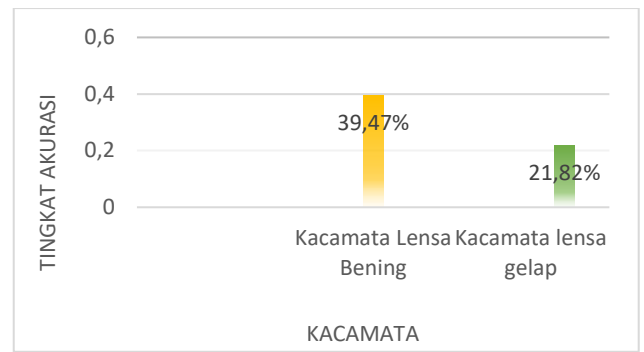


Gambar 7

(Pengaruh Aksesoris topi terhadap akurasi)

Pada pengujian kacamata lensa bening, tingkat akurasi yang didapat sebesar 47,83% hal ini dikarenakan Face Recognition mendeteksi seluruh wajah termasuk mata.

Pada pengujian menggunakan kacamata gelap, tingkat akurasi yang di dapatkan 21,82% sangat rendah dari pada lensa kaca bening, dikarenakan mata yang tertutup oleh lensa gelap.



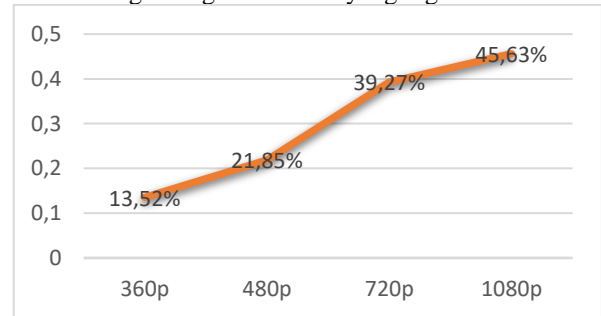
Gambar 8

(Pengaruh Aksesoris kacamata terhadap akurasi)

D. Pengujian Resolusi Video

Pengujian resolusi video, disini penulis menggunakan resolusi video yang berbeda-beda, mulai dari 360p, 480p, 720p, 1080p dengan Threshold 0.8, Durasi video 16detik dan video yang sama.

Pada pengujian resolusi video, Untuk resolusi tergantung dari jenis kamera bisa seperti kamera CCTV, Handphone, Webcam, dan beberapa jenis kamera lainnya. Resolusi juga bisa di atur tergantung dari device yang digunakan.



Gambar 9

(Pengaruh Resolusi video terhadap akurasi)

E. Pengujian tanpa database wajah

Pengujian mencoba menjalankan program yang di buat tanpa ada data di dalam database wajah, hasil yang di keluarkan error dikarenakan tidak adanya data pada database yang seharusnya di isi wajah orang yang ingin dicari..

F. Running Time

Saat menjalankan Program Face Recognition, Progress Bar atau Running Time akan muncul untuk memberi tahu berapa persen yang telah di proses, agar lebih mudah untuk di ketahui seberapa lama waktu yang dibutuhkan untuk memproses suatu video.

V. KESIMPULAN

Dalam kode Python yang diberikan, digunakan library Face Recognition untuk melakukan pengenalan wajah dalam video. Pertama, library OpenCV digunakan untuk membaca video input. Selanjutnya, cascade classifier (haarcascade_frontalface_default.xml) dari OpenCV digunakan untuk mendeteksi wajah dalam setiap frame video. Setelah wajah terdeteksi, kode akan membandingkannya dengan data wajah referensi yang telah dimuat sebelumnya. Teknologi Face Recognition memungkinkan sistem ini untuk

mengenali wajah yang sesuai dengan data referensi. Hasil pengenalan wajah dicatat dalam file log. Proses ini juga dilengkapi dengan tampilan progress bar untuk memantau kemajuan pengenalan wajah dalam video.

Keuntungan utama dari menggunakan teknologi Face Recognition dalam sistem pengawasan CCTV untuk keamanan rumah adalah meningkatnya tingkat keamanan dan kenyamanan. Dengan teknologi ini, rumah dapat mengenali siapa yang memasuki atau bergerak di sekitarnya secara otomatis. Hal ini memungkinkan pemilik rumah untuk dengan cepat mengetahui identitas orang-orang yang ada di properti mereka. Selain itu, teknologi Face Recognition dapat diintegrasikan dengan perangkat lain, seperti pintu otomatis yang terkunci atau memberikan izin akses terbatas kepada individu tertentu. Ini membuat rumah lebih aman dan efisien dalam pengelolaan akses.

Cara untuk menyimpan data wajah yang telah dikenali dan digunakan sebagai referensi di masa yang akan datang dijelaskan dalam kode dengan pembuatan folder "Face_Data" jika folder tersebut belum ada. Setiap kali wajah baru dikenali, gambar wajah tersebut akan disimpan dalam folder ini. Data referensi ini dapat dikelola dengan memberikan nama file yang jelas dan unik, yang akan memudahkan identifikasi di masa mendatang. Selain itu, kode ini juga mencatat informasi tambahan seperti nama orang yang dikenali, waktu, dan tanggal pengenalan wajah tersebut dalam file log. Ini membantu dalam pelacakan dan manajemen data wajah dengan lebih baik

Tingkat akurasi dari sistem Face Recognition yang digunakan dalam sistem pengawasan CCTV untuk keamanan rumah dapat diatur dengan menggunakan akurasi threshold sebesar 80%. Dengan threshold ini, sistem hanya akan menganggap wajah sebagai dikenali jika tingkat kesamaan antara wajah yang dikenali dengan data referensi melebihi 80%. Dengan kata lain, wajah yang terdeteksi harus sangat mirip dengan data referensi yang ada agar dianggap sebagai hasil pengenalan yang valid. Penggunaan threshold ini membantu mengurangi kesalahan pengenalan dan meningkatkan keamanan sistem pengawasan CCTV untuk rumah Anda, karena hanya wajah yang sangat cocok yang akan diakui oleh sistem.

REFERENSI

[1] P. C. P. Neto, J. R. Pinto, F. Boutros, N. Damer, A. F. Sequeira, and J. S. Cardoso, "Beyond Masks: On the

- Generalization of Masked Face Recognition Models to Occluded Face Recognition," *IEEE Access*, vol. 10, pp. 86222–86233, 2022, doi: 10.1109/ACCESS.2022.3199014.
- [2] H. Yang and X. Han, "Face recognition attendance system based on real-time video processing," *IEEE Access*, vol. 8, pp. 159143–159150, 2020, doi: 10.1109/ACCESS.2020.3007205.
- [3] P. Terhorst, M. Huber, N. Damer, F. Kirchbuchner, K. Raja, and A. Kuijper, "Pixel-Level Face Image Quality Assessment for Explainable Face Recognition," *IEEE Trans Biom Behav Identity Sci*, vol. 5, no. 2, pp. 288–297, Apr. 2023, doi: 10.1109/TBIOM.2023.3263186.
- [4] Z. Huang, J. Zhang, and H. Shan, "When Age-Invariant Face Recognition Meets Face Age Synthesis: A Multi-Task Learning Framework and a New Benchmark," *IEEE Trans Pattern Anal Mach Intell*, vol. 45, no. 6, pp. 7917–7932, Jun. 2023, doi: 10.1109/TPAMI.2022.3217882.
- [5] Y. Martínez-Díaz, H. Méndez-Vázquez, L. S. Luevano, M. Nicolás-Díaz, L. Chang, and M. González-Mendoza, "Towards Accurate and Lightweight Masked Face Recognition: An Experimental Evaluation," *IEEE Access*, vol. 10, pp. 7341–7353, 2022, doi: 10.1109/ACCESS.2021.3135255.
- [6] Z. An, W. Deng, J. Hu, Y. Zhong, and Y. Zhao, "APA: Adaptive Pose Alignment for Pose-Invariant Face Recognition," *IEEE Access*, vol. 7, pp. 14653–14670, 2019, doi: 10.1109/ACCESS.2019.2894162.
- [7] N. Li et al., "Chinese face dataset for face recognition in an uncontrolled classroom environment," *IEEE Access*, 2023, doi: 10.1109/ACCESS.2023.3302919.
- [8] M. Gupta, K. Bisht, A. Sharma, and D. Upadhyay, "HaarCascade and LBPH Algorithms in Face Recognition Analysis," in *2023 World Conference on Communication & Computing (WCONF)*, IEEE, Jul. 2023, pp. 1–4. doi: 10.1109/WCONF58270.2023.10235019.
- [9] Abhishek Jaiswal, "Face Detection using Haar-Cascade using Python," *Analytics Vidhya*. Accessed: Sep. 26, 2023. [Online]. Available: [https://www.analyticsvidhya.com/blog/2022/10/face-detection-using-haar-cascade-using-python/#:~:text=Haar%20cascade%20uses%20the%20casca de,working%2C%20refer%20to%20this%20link](https://www.analyticsvidhya.com/blog/2022/10/face-detection-using-haar-cascade-using-python/#:~:text=Haar%20cascade%20uses%20the%20cascade,working%2C%20refer%20to%20this%20link).
- [10] Kuster Boris, *Face Detection and Face Recognition in Python Programming Language*. Society of Digital Information and Wireless Communications (SDIWC), 2018.