

Perancangan Deteksi Wajah pada Aplikasi Berbasis React Native Menggunakan Metode Haar Cascade

Ahmad Fadhlur Rohman¹, Henny Dwi Bhakti²

¹Program Studi Teknik Informatika, Fakultas Teknik, Universitas Muhammadiyah Gresik

*e-mail: 4hmad.fadhlur@gmail.com¹, hennydwi@umg.ac.id²

Received:
06.05.2024

Revised:
20.05.2024

Accepted:
10.06.2024

Available online:
27.06.2024

Abstract: Increasing the security of digital applications in the current era is important, because the threat of cyber crime is increasing. An appropriate security system, such as an authentication system, can prevent cybercrime from occurring in an application. Bangbeli as a startup that operates in digital payment applications, it is important to improve the security system in its application. This research aims to design a face detection system and face verification page for the Bangbeli application based on the React Native framework. The method used to implement the face detection system is the Haar Cascade method and uses the OpenCV library. The Haar Cascade method has the ability to detect an object quickly, such as a face. OpenCV is an open source library that is widely used for digital image processing, and can be implemented in various programming languages. The results of this research showed that the verification page was successfully created and the face detection system could run using the Haar Cascade method which was developed for an Android application based on the React Native framework.

Keywords: Face Detection, Haar Cascade, OpenCV, React Native

Abstrak: Peningkatan keamanan aplikasi digital di era sekarang penting dilakukan, karena ancaman kejahatan siber yang semakin meningkat. Sistem keamanan yang tepat, seperti sistem autentikasi, dapat mencegah terjadinya kejahatan siber pada suatu aplikasi. Bangbeli sebagai salah satu *startup* yang bergerak di aplikasi pembayaran digital, penting untuk meningkatkan sistem keamanan pada aplikasinya. Penelitian ini bertujuan untuk merancang sistem deteksi wajah dan halaman verifikasi wajah untuk aplikasi Bangbeli yang berbasis *framework* React Native. Metode yang digunakan dalam mengimplementasikan sistem deteksi wajah adalah metode Haar Cascade dan menggunakan *library* OpenCV. Metode Haar Cascade memiliki kemampuan mendeteksi suatu objek dengan cepat, seperti objek berupa wajah. OpenCV merupakan *library* sumber terbuka yang banyak digunakan untuk pengolahan citra digital, serta dapat diimplementasikan ke berbagai macam bahasa pemrograman. Hasil pada penelitian ini, didapatkan halaman verifikasi berhasil dibuat dan sistem deteksi wajah dapat berjalan dengan menggunakan metode Haar Cascade yang dikembangkan pada aplikasi android berbasis *framework* React Native.

Kata kunci: Deteksi Wajah, Haar Cascade, OpenCV, React Native

1. PENDAHULUAN

Zaman ini, teknologi terus mengalami kemajuan dan perkembangan yang pesat. Berkat kemajuan teknologi, banyak sistem yang dulunya bersifat konvensional kini dapat dijalankan secara digital. Sebagai contoh, proses pembayaran transaksional saat ini menjadi lebih mudah dilakukan dengan adanya sistem digital. Aplikasi pembayaran digital berkembang pesat dan banyak digunakan karena memiliki sejumlah keunggulan (Aditya, 2021). Salah satunya adalah aplikasi bernama Bangbeli.

Bangbeli merupakan salah satu *startup* di Indonesia yang bergerak di bidang *financial technology*. Sebagai *startup financial technology*, Bangbeli bertanggung jawab pada layanan PPOB (*Payment Point Online Banking*) yang ada pada platform aplikasi Bangbeli. *Payment Point Online Banking* merupakan sebuah layanan pembayaran untuk memudahkan seseorang dalam melakukan pembayaran berbagai jenis tagihan, seperti Token Listrik, Pulsa, BPJS, dan berbagai macam lainnya (R. R. Saputri, 2022).

Semakin berkembangnya teknologi, semakin meningkat juga kejahatan *cyber* yang dapat terjadi. Adanya beberapa kasus kejahatan *cyber* di Indonesia, seperti *hacking*, *phishing*, *malware*, penipuan, dan penyadapan (Handayani et al., 2023), turut menjadi ancaman bagi *startup* yang bergerak di bidang teknologi. Seperti yang sering terjadi akhir-akhir ini yaitu kasus *sniffing*. Pelaku kejahatan mengirimkan aplikasi palsu yang berisi *malware*. Aplikasi palsu tersebut dapat berupa resi, atau undangan pernikahan. Ketika aplikasi tersebut terpasang pada ponsel korban, pelaku dapat mencuri informasi sensitif seperti kata sandi hingga dapat mengakses rekening *m-banking* korban untuk melakukan penarikan uang (Diahwahyuningtyas & Nugroho, 2023). Contohnya pada kasus Derasmus Kenlopo, warga Kelurahan Naimata, Kecamatan Maulafa, Kota Kupang, Nusa Tenggara Timur (NTT), yang kehilangan Rp.14.000.000 akibat modus *sniffing* menggunakan undangan nikah digital (Bupu et al., 2024). Contoh lainnya seperti pada kasus *order* fiktif Lazada senilai Rp.22.000.000 pada tahun 2018,

yang dilakukan pelaku dengan melihat kode verifikasi kartu kredit pengguna yang dikirim ke email pengguna (Simbolon et al., 2021).

Untuk mengatasi risiko kejahatan cyber, penerapan sistem keamanan yang tepat, akan sangat penting dalam pengembangan suatu aplikasi digital. Salah satunya dengan menerapkan sistem autentikasi, yang bertujuan untuk mengidentifikasi pengguna sebelum mengakses layanan tertentu pada suatu aplikasi. Sistem autentikasi dapat berupa kata sandi, OTP (*One Time Password*), hingga biometrik (D. A. E. Saputri et al., 2023). Biometrik merupakan suatu teknologi pengenalan makhluk hidup pada karakteristik uniknya, dimana antara individu tidak akan memiliki ciri khas yang 100% sama. Biometrik juga memiliki karakteristik tidak mudah hilang, dan tidak mudah dicuri. Identifikasi karakteristik pada umumnya dilakukan pada bagian sidik jari, retina mata, dan struktur wajah (Nicco & Fahrudi, 2015). Teknologi ini memiliki tingkat keamanan yang lebih baik jika dibandingkan dengan sistem autentikasi kata sandi atau PIN (Priambodo & Wahjoeono, 2023).

Sebagai *startup* penyedia layanan yang berbasis keuangan, Bangbeli perlu untuk meningkatkan keamanan yang ada pada aplikasinya. Hal ini akan membuat pengguna lebih merasa aman dan dapat mencegah kerugian yang dapat terjadi akibat tindakan kejahatan cyber. Untuk meningkatkan keamanan pengguna aplikasi bangbeli, maka pada penelitian ini dirancang sistem autentikasi tambahan yaitu *Face Verification*. Fitur yang akan dikembangkan penulis berfokus pada pembuatan halaman untuk verifikasi wajah pengguna pada sisi *client* aplikasi Bangbeli. Dengan dirancangnya fitur tersebut diharapkan dapat menjadi alternatif keamanan tambahan dan meningkatkan keamanan aplikasi Bangbeli. Dalam fitur tersebut, diperlukan suatu metode untuk mengidentifikasi object berupa wajah. *Face Detection* berguna untuk mendeteksi keberadaan wajah pada suatu citra, yang kemudian hasilnya dapat digunakan untuk proses verifikasi wajah.

Metode *Face Detection* yang digunakan pada penelitian ini yaitu metode Haar Cascade. Metode ini dipilih karena memiliki akurasi yang cukup baik dalam mengenali suatu object seperti wajah (Yulina, 2021). Pada penelitian terdahulu tentang metode Haar Cascade yang digunakan untuk pendeteksian object berupa masker oleh (Anarki et al., 2021), didapatkan hasil, sistem dapat mendeteksi objek masker dengan bermacam warna dan bahan pada pengujiannya. Hasil pengujiannya juga mendapatkan tingkat akurasi sebesar 88.7% pada faktor skala 1.2. Penelitian lain yang dilakukan oleh (Suharso, 2016), metode Haar Cascade memiliki akurasi sebesar 88.89% dalam mendeteksi wajah pada posisi *frontal* (menghadap depan) dengan derajat sisi kiri 30°, depan 15°, dan kanan 30°. Penelitian ini, metode Haar Cascade diimplementasikan menggunakan *library* yang ada pada OpenCV. OpenCV merupakan *library open source* untuk pemrosesan citra dan juga dapat digunakan pada aplikasi *mobile* yang dikembangkan menggunakan *framework* React Native.

2. METODE

2.1 Studi Literatur

Studi literatur bertujuan untuk mengumpulkan informasi dari artikel ataupun dokumentasi yang terkait dengan pengembangan *Face Detection* dan penerapannya pada aplikasi Android yang berbasis *framework* React Native.

2.1.1 Face Detection

Deteksi wajah merupakan suatu langkah awal yang cukup penting sebelum memulai tahap pengenalan wajah. Proses deteksi wajah dapat dianggap sebagai bentuk klasifikasi pola, dimana citra masukan diidentifikasi dan diklasifikasikan menjadi label kelas berupa wajah dan bukan wajah. Proses ini menjadi dasar untuk proses pengenalan wajah (Piarsa & Hisamuddin, 2010).

2.1.2 OpenCV

OpenCV diluncurkan secara resmi oleh Intel Research pada tahun 1999. Sampai saat ini OpenCV sudah digunakan lebih dari 47 ribu orang dari komunitas pengguna dan juga perusahaan besar seperti Google, Microsoft, Intel, Honda, Toyota. Penggunaannya mencakup berbagai hal, mulai dari penggabungan gambar *street view*, penerapan pada robot untuk navigasi, seni interaktif, hingga pemeriksaan label produk di pabrik-pabrik di seluruh dunia (OpenCV, n.d.-b).

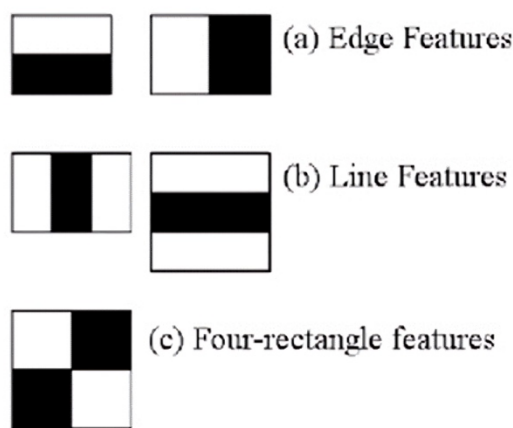
OpenCV (*Open Source Computer Vision*) merupakan *library* terbuka yang dikembangkan oleh Intel bertujuan untuk menyederhanakan pemrograman terkait

pengolahan citra digital (Fitrah, 2020). OpenCV memiliki berbagai fitur dalam pemrosesan citra digital, seperti: manipulasi gambar, pendeteksi objek, pengenalan wajah, dan berbagai macam algoritma pemrosesan citra. OpenCV yang merupakan *open source computer vision* yang dapat diintegrasikan menggunakan bahasa pemrograman C, C++, Python, Java dan Matlab (Susim & Darujati, 2021), sehingga dapat diimplementasikan ke berbagai platform seperti *website* hingga *mobile app*.

2.1.3 Haar Cascade

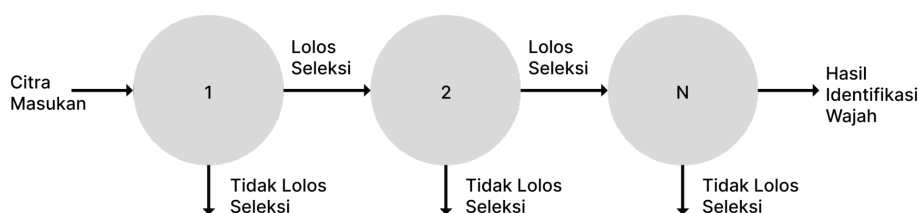
Haar Cascade adalah algoritma pembelajaran mesin yang digunakan untuk mendeteksi suatu objek, algoritma ini diusulkan oleh Paul Viola dan Michael Jones pada tahun 2001. Algoritma Haar Cascade menggunakan fungsi cascade pada pendekatan pembelajarannya. Fungsinya dilatih dari berbagai citra positif dan citra negatif. Citra positif berisi citra yang terdapat objek yang ingin dideteksi, sedangkan citra negatif sebaliknya, tidak terdapat objek yang ingin dideteksi (Yulina, 2021).

Algoritma ini bekerja dengan mengekstraksi sebuah fitur dari gambar, algoritma Haar Cascade menggunakan filter yang mirip dengan konsep *convolutional kernel* yang disebut fitur Haar seperti pada gambar 1 di bawah. Setiap fitur diperoleh dengan mengurangi dari jumlah total kotak piksel putih dan jumlah total piksel hitam untuk didapatkan nilai selisihnya. Untuk perhitungan setiap fiturnya juga menggunakan *integral image* dari citra gambar dalam bentuk *grayscale*. Fitur-fitur yang didapatkan kemudian diseleksi menggunakan *Adaboost* untuk didapatkan fitur yang terbaik dari total lebih dari 160.000 fitur.



Gambar 1. Contoh Fitur Haar (OpenCV, n.d.-a)

Algoritma ini menerapkan konsep *Cascade Classifiers* (Klasifikasi bertingkat) seperti pada gambar 2. Dalam setiap tingkat, subcitra akan diklasifikasikan menggunakan suatu fitur Haar yang sudah diseleksi. Hasil dari klasifikasi, jika subcitra memenuhi fitur tersebut akan dilanjutkan ke tingkat selanjutnya untuk dihitung ulang menggunakan fitur yang lain, subcitra yang tidak memenuhi fitur akan dibuang dan tidak digunakan lagi. Klasifikasi ini dilakukan hingga beberapa tingkat untuk menyeleksi citra dan mendapatkan hasil subcitra yang memenuhi semua fitur pada setiap tingkatan (Viola & Jones, 2001). Hasil subcitra yang didapat akan semakin sedikit hingga tersisa hanya 2% dari citra masukan, berupa area wajah (Yulina, 2021).



Gambar 2. Sistem *Cascade Classifiers* (Santoso & Kristianto, 2020)

2.1.4 React Native

React Native merupakan sebuah *framework* yang membantu pengembang dalam mengembangkan aplikasi *mobile* menggunakan bahasa pemrograman Javascript tanpa mengurangi pengalamannya (Eisenman, 2015). *Framework* React Native bersifat *mobile cross platform*, dimana dengan satu pengkodean dapat digunakan untuk mengembangkan aplikasi yang berbasis android dan ios sekaligus. *Framework* ini juga telah banyak digunakan oleh perusahaan besar di seluruh dunia (Malahella et al., 2020).

2.1.5 API

API atau *Application Programming Interface* merupakan suatu instruksi, fungsi, dan protokol yang dapat digunakan pengembang perangkat lunak dalam merancang aplikasi (Data Basecamp, 2022). Secara sederhana API merupakan serangkaian fungsi pemrograman yang disediakan oleh suatu aplikasi untuk memfasilitasi integrasi layanan dengan aplikasi yang sedang dibangun. Dalam pemrograman android, API umumnya digunakan untuk mempermudah pertukaran data antara aplikasi dan server (Radityatama, 2017).

2.2 Analisis dan Perancangan Sistem

2.2.1 Analisis Metode Haar Cascade

Metode Haar Cascade dipilih karena merupakan metode pengenalan wajah yang paling banyak digunakan, dan juga relative memiliki kinerja yang cepat, akurat, dan efisien (Suharso, 2016). Cara kerja metode haar yaitu dimulai dengan mengubah citra berwarna menjadi grayscale, kemudian memeriksa satu bagian pixel pada satu waktu, yang kemudian setiap bagian putih atau hitam akan dijumlahkan pada tiap bagian dan dihitung selisihnya. Sehingga didapatkan nilai fitur ekstraksinya menggunakan *integral image*.

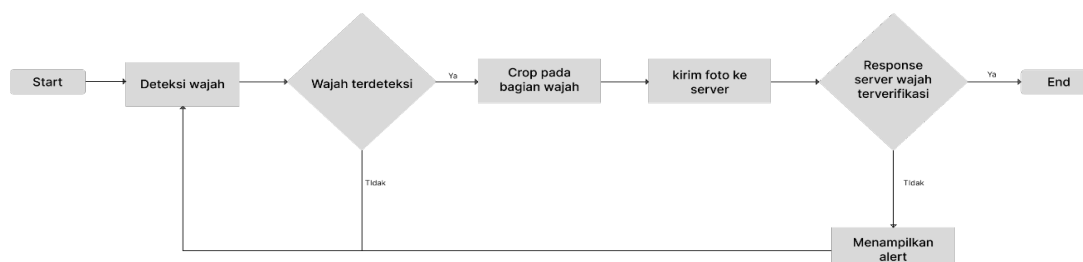
Hasil dari fitur Haar diseleksi menggunakan AdaBoost, kemudian digunakan pada Cascade Classifiers atau klasifikasi bertingkat untuk mendapat hasil area wajah yang lebih akurat. Pada tahap tingkat pertama, setiap subcitra akan diklasifikasikan dengan suatu fitur haar. Apabila hasilnya tidak memenuhi fitur haar tertentu maka akan ditolak, jika terpenuhi akan dilanjut ke tingkat berikutnya. Semakin bertambahnya suatu tingkatan maka perlu untuk menambahkan fitur yang lebih spesifik hingga jumlah subcitra akan berkurang dan didapatkan hasil area wajah.



Gambar 3. Alur Metode Haar Cascade

2.2.2 Perancangan Sistem

Perancangan sistem bertujuan untuk memberikan gambaran mengenai arsitektur yang dikembangkan pada halaman verifikasi yang menggunakan sistem deteksi wajah. Desain sistem ini digambarkan pada diagram alur berikut:



Gambar 4. Desain Sistem Halaman Verifikasi Wajah

Tahap awal ketika membuka halaman verifikasi akan dilakukan pendeteksian wajah menggunakan metode Haar Cascade pada pustaka OpenCV, ketika wajah

terdeteksi kemudian akan otomatis dilakukan pemotongan area wajah pada gambar dan akan dikirim ke server melalui API untuk dilakukan verifikasi, setelah mendapat hasil umpan balik dari server, jika hasilnya wajah tidak terverifikasi maka akan memunculkan pesan gagal memverifikasi dan akan diulang dari proses pendeteksian wajah, jika berhasil terverifikasi maka dapat dilanjut ke halaman berikutnya.

2.3 Implementasi dan Pengujian

Menerapkan *face detection* dengan metode Haar Cascade menggunakan OpenCV *library*. Sistem diterapkan pada Aplikasi yang berbasis React Native berdasar pada hasil analisis dan perancangan yang sudah dilakukan. Sehingga sistem yang telah dirancang dapat beroperasi dan difungsikan secara nyata dan dapat mencapai tujuan yang diinginkan.

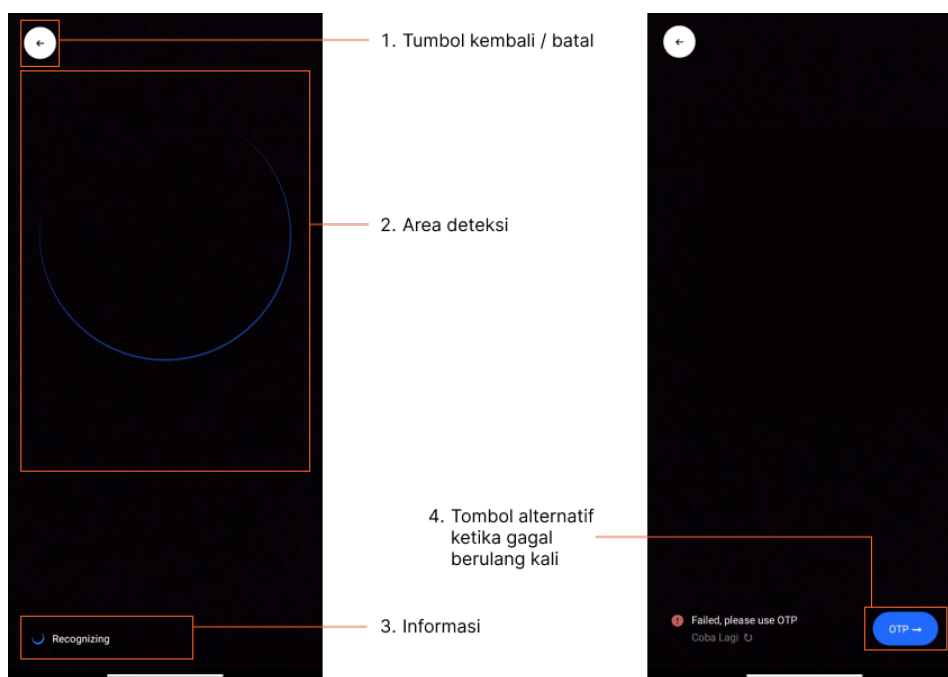
Pengujian dilakukan dengan menguji manual fungsi deteksi wajah metode Haar Cascade pada aplikasi dapat berjalan dan sistem dapat melakukan pemotongan citra pada area wajah yang terdeteksi. Pengujian kedua yaitu menguji fungsi area deteksi wajah menggunakan pengujian *Blackbox*, pengujian ini untuk memastikan wajah terdeteksi hanya pada area yang ditentukan.

2.4 Kesimpulan

Tahap Kesimpulan merupakan tahap akhir yang berisi tentang kesimpulan dari hasil yang didapatkan pada penggunaan metode Haar Cascade untuk pendeteksian wajah pada halaman verifikasi.

3. HASIL DAN PEMBAHASAN

Pengimplementasiannya didasarkan pada desain sistem yang telah disusun sebelumnya, yang dapat dilihat pada gambar 4. Sistem ini diimplementasikan pada aplikasi *mobile* yang dikembangkan menggunakan *framework* React Native. Gambar 5 di bawah ini merupakan hasil implementasi antarmuka halaman verifikasi. Halaman verifikasi terdiri dari beberapa bagian, yaitu area deteksi wajah, bagian informasi deteksi, tombol OTP (One Time Password) sebagai alternatif verifikasi, dan tombol untuk kembali.



Gambar 5. Hasil Implementasi Antarmuka Halaman Verifikasi

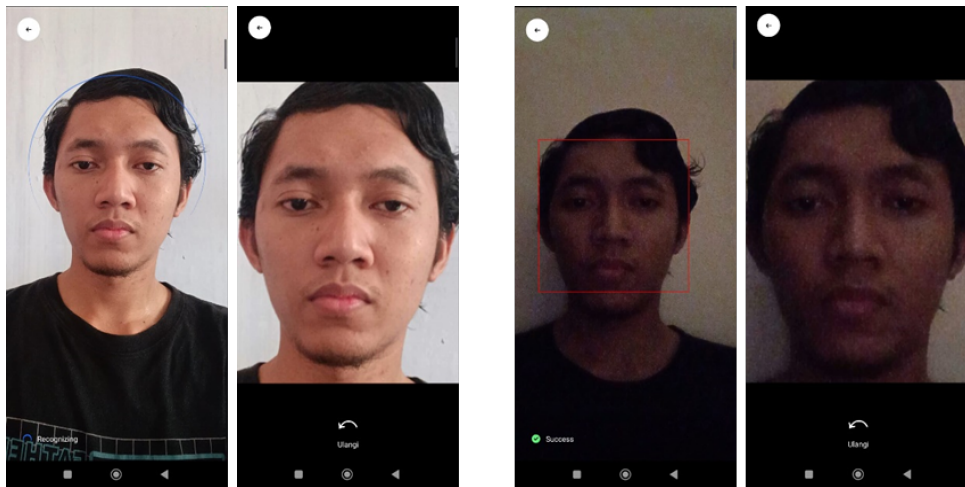
Area deteksi merupakan lokasi untuk mendeteksi adanya suatu wajah dengan beberapa ketentuan agar wajah dapat dideteksi, seperti posisi wajah tepat berada di tengah, tidak terpotong, tidak terlalu kecil dan terlalu besar.

Bagian informasi ditujukan untuk memberikan informasi pada pengguna terkait proses verifikasi wajah, beberapa informasinya yaitu *recognizing* dimana sistem sedang pada posisi *standby* untuk pemrosesan dan verifikasi wajah, kemudian informasi *success* ketika wajah berhasil terverifikasi, dan informasi *failed* ketika wajah berulang kali gagal terverifikasi.

Tombol OTP digunakan untuk alternatif ketika gagal berulang kali dalam verifikasi wajah, fitur tombol ini dapat diabaikan jika verifikasi wajah sangat diperlukan untuk menjaga keamanan data user yang bersifat sensitif. Tombol kembali digunakan untuk membatalkan verifikasi atau kembali ke halaman sebelumnya.

Metode Haar Cascade pada *library* OpenCV diimplementasikan dengan pemrograman *native* Android yaitu bahasa Java. *Module* OpenCV diimport pada proyek React Native, kemudian *import* data *training* pada *library* OpenCV berupa *haarcascade_frontalface_alt.xml* untuk klasifikasi wajah. Program Java OpenCV tersebut kemudian dihubungkan dengan fitur *Native Modules* yang dimiliki oleh *framework* React Native, sehingga dapat saling terhubung dan digunakan pada halaman yang dibuat dengan *framework* React Native yang berbasis bahasa Javascript. Data citra video yang didapat pada halaman antarmuka dikirimkan ke program java untuk diolah menjadi citra keabuan menggunakan fungsi *Imgproc.COLOR_BGR2GRAY* untuk kemudian dilakukan deteksi wajah menggunakan metode Haar Cascade dengan fungsi *CascadeClassifier* yang ada pada *library*, sehingga didapat keberadaan wajah dan posisinya dalam citra tersebut. Hasilnya dikirim kembali ke halaman antarmuka untuk ditampilkan informasi yang didapat, dan apabila diketahui keberadaan wajah, maka akan otomatis dikirim ke server melalui API untuk dilakukan verifikasi wajah.

Hasil percobaan *Face Detection* menggunakan metode Haar Cascade pada halaman verifikasi dapat berjalan dan berhasil mendeteksi keberadaan wajah pada antarmuka, dalam kondisi cahaya yang cukup maupun sedikit. Sistem juga dapat melakukan pemotongan gambar pada area wajah. Hasilnya dapat dilihat pada gambar 6.



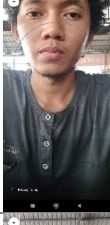



Gambar 6. Hasil Percobaan Deteksi Wajah Metode Haar Cascade

Pengujian fungsi sistem deteksi dilakukan untuk memastikan fungsi sistem deteksi wajah berjalan sesuai pada area deteksi yang sudah ditentukan, dapat dilihat pada gambar 5. Pengujian dilakukan dengan menguji lokasi dan ukuran wajah pada antarmuka. Berikut tabel hasil pengujian *Blackbox* untuk sistem deteksi:

Tabel 1. Pengujian Blackbox Untuk Sistem Area Deteksi

No	Lokasi	Ukuran	Gambar	Hasil Diharapkan	Kesimpulan
1	Tepat di tengah	Menyerupai lingkaran deteksi		Mendeteksi	Berhasil

2	Tepat di tengah	\leq dari Setengah lingkaran deteksi		Tidak mendeteksi	Berhasil
3	Tepat di tengah	Memenuhi layar antarmuka		Tidak mendeteksi	Berhasil
4	Di luar area deteksi	Menyerupai lingkaran deteksi		Tidak mendeteksi	Berhasil
5	Di luar area deteksi	\leq dari Setengah lingkaran deteksi		Tidak mendeteksi	Berhasil

Berdasarkan hasil pengujian sistem area deteksi yang dilakukan, didapat kesimpulan bahwa sistem yang telah dibuat berjalan sesuai dengan yang diharapkan. Sistem yang dibuat hanya dapat melakukan deteksi wajah pada area deteksi. Sistem ini digunakan untuk meminimalisir hasil deteksi wajah yang tidak layak atau tidak diharapkan.

4. KESIMPULAN

Berdasarkan hasil implementasi yang dilakukan pada halaman *verification* menggunakan metode Haar Cascade, dapat ditarik kesimpulan bahwa sistem berhasil dikembangkan pada aplikasi android yang berbasis *framework* React Native. Sistem dapat mendeteksi wajah dan melakukan pemotongan area wajah pada citra sebelum dikirimkan ke server, serta dapat menampilkan respon dari server pada antarmuka halaman verifikasi. Saran penulis untuk pengembangan selanjutnya, dapat ditambahkan sistem *yaw*, *pitch*, dan *roll* untuk memperketat sistem deteksi wajah, sehingga akan lebih sulit lagi untuk dimanipulasi. Serta menambahkan informasi panduan pada halaman deteksi wajah untuk mempermudah pengguna yang masih awam.

DAFTAR PUSTAKA

Aditya. (2021). *7 Aplikasi Dompot Digital Terbaik di Indonesia dan Cara Transfer yang Mudah*. Flip.Id. <https://flip.id/blog/aplikasi-dompot-digital-terbaik>

Anarki, G. A., Auliasari, K., & Orisa, M. (2021). Penerapan Metode Haar Cascade Pada Aplikasi Deteksi Masker. *JATI (Jurnal Mahasiswa Teknik Informatika)*, 5(1), 179–186. <https://doi.org/10.36040/jati.v5i1.3214>

Bupu, A. G., Medan, K. K., & Amalo, H. (2024). Analisis Yuridis Cyber Crime Pembobolan Dana Nasabah pada Aplikasi Mobile Banking dengan Modus Pembobolan Jalur Undangan Pernikahan Palsu Karolus Kopong Medan Lahirnya Undang-undang Nomor 11 Tahun 2008 Tentang Informasi dan Transaksi. *Hakim: Jurnal Ilmu Hukum Dan Sosial*, 2(2). <https://doi.org/https://doi.org/10.51903/hakim.v2i2.1829>

Data Basecamp. (2022). *What is an API (Application Programming Interface)*. Databasecamp.De.

- <https://databasecamp.de/en/data/api-en>
- Diahwahyuningtyas, A., & Nugroho, R. S. (2023, January 29). Ramai soal Sniffing, Modus Penipuan Resi hingga Undangan yang Bisa Curi Saldo Rekening. *Kompas.Com*. <https://www.kompas.com/tren/read/2023/01/29/183000165/ramai-soal-sniffing-modus-penipuan-resi-hingga-undangan-yang-bisa-curi?page=all>
- Eisenman, B. (2015). *Learning React Native* (1st ed.). O'Reilly Media, Inc.
- Fitrah, G. F. R. (2020). *RANCANG BANGUN APLIKASI REKOMENDASI PENYAMPAIAN MATERI PEMBELAJARAN BERDASARKAN DETEKSI EMOSI WAJAH SKRIPSI* [UNIVERSITAS KOMPUTER INDONESIA]. <http://elibrary.unikom.ac.id/id/eprint/2957>
- Handayani, D., Fardiyanti Rosianah, S., Imam Sobari, D., Afriandi Putra, D., Muchlis Zulhija, H., Maulida, I., Arkhab Purnama Agdana, M., Khusni Mubarakh, M., Wulansari, W., & Faizul Khanafi, W. (2023). Ragam Modus Cyber Crime Di Era Digital 4.0. *Abdi Jurnal Publikasi*, 1(4), 422–427. <https://jurnal.portalpublikasi.id/index.php/AJP/index422>
- Malahella, A. H., Arwani, I., & Tibyani. (2020). Pemanfaatan Framework React Native dalam Pengembangan Aplikasi Pemesanan Minuman Kopi pada Kedai Bycoffee. *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer*, 4(9), 3178–3184. <https://j-ptiik.ub.ac.id/index.php/j-ptiik/article/view/7898>
- Nicco, & Fahrudi, I. (2015). Rancang Bangun Sistem Biometrik Pengenalan Wajah Menggunakan Principal Component Analysis. *Jurnal Integrasi*, 6(1), 64–71. <https://jurnal.polibatam.ac.id/index.php/JI/article/view/92>
- OpenCV. (n.d.-a). *Face Detection using Haar Cascades*. Opencv.Org. Retrieved April 3, 2024, from https://docs.opencv.org/3.4/d2/d99/tutorial_js_face_detection.html
- OpenCV. (n.d.-b). *OpenCV About*. Opencv.Org. Retrieved May 8, 2024, from <https://opencv.org/about/>
- Piarsa, I., & Hisamuddin, R. (2010). Sistem Verifikasi Online Menggunakan Biometrika Wajah. *Teknologi Elektro*, 9(1), 88–94. <https://ojs.unud.ac.id/index.php/JTE/article/download/1772/1258>
- Priambodo, B. W., & Wahjoeono, D. (2023). Perlindungan Hukum terhadap Penggunaan Teknologi Biometrik dalam Transaksi Perbankan untuk Meningkatkan Keamanan. *Madani : Jurnal Ilmiah Multidisipliner*, 1(11), 322–327. <https://doi.org/https://doi.org/10.5281/zenodo.10276789>
- Radityatama, R. G. (2017). *Rancang Bangun Aplikasi Mobile Android Sistem Kehadiran Mahasiswa Melalui Pencocokan Wajah Dengan Menggunakan Library Android Face Recognition With Deep Learning Studi* [Institut Teknologi Sepuluh Nopember]. <http://repository.its.ac.id/id/eprint/42762>
- Santoso, B., & Kristianto, R. P. (2020). Implementasi Penggunaan Opencv Pada Face Recognition Untuk Sistem Presensi Perkuliahan Mahasiswa. *Sistemasi*, 9(2), 352–361. <https://doi.org/10.32520/stmsi.v9i2.822>
- Saputri, D. A. E., Ernawati, I. A., Adjeng Nutfa, N., Rabbaanii, & Satriani, A. D. (2023). Penerapan AAA Security Dalam Aplikasi BNI Mobile Banking. *Indonesian Journal of Innovation Multidisipliner Research*, 1(2), 63–73. <https://multidisipliner.org/index.php/ijim/article/view/9>
- Saputri, R. R. (2022). *STRATEGI MINIMUM VIABLE PRODUCT DALAM PENGEMBANGAN PRODUK STARTUP DI APLIKASI PEMBAYARAN DIGITAL BANGBELI (PT. DOA ANAK DIGITAL) KABUPATEN MALANG JAWA TIMUR SKRIPSI Oleh*. [http://digilib.uinkhas.ac.id/17153/1/Rizka Ramadhina Saputri_E20182088.pdf](http://digilib.uinkhas.ac.id/17153/1/Rizka%20Ramadhina%20Saputri_E20182088.pdf)
- Simbolon, M. M., Kesuma, I. G. K. W., & Wibowo, A. E. (2021). Kejahatan Siber pada Penyelenggaraan Perdagangan Berbasis Sistem Elektronik Dalam langkah Pengamanan Pertumbuhan Ekonomi Digital Indonesia. *Defendonesia*, 5(1), 1–12. <https://doi.org/https://doi.org/10.54755/defendonesia.v5i1.98>
- Suharso, A. (2016). Pengenalan Wajah Menggunakan Metode Viola-Jones dan Eigenface Dengan Variasi Posisi Wajah Berbasis Webcam. *Techno Xplore : Jurnal Ilmu Komputer Dan Teknologi Informasi*, 1(2), 19–30. <https://doi.org/10.36805/technoxplore.v1i2.107>
- Susim, T., & Darujati, C. (2021). PENGOLAHAN CITRA UNTUK PENGENALAN WAJAH (FACE RECOGNITION) MENGGUNAKAN OPENCV. *Jurnal Syntax Admiration*, 2(3), 534–545. <https://doi.org/https://doi.org/10.46799/jsa.v2i3.202>
- Viola, P., & Jones, M. (2001). Rapid object detection using a boosted cascade of simple features. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1. <https://doi.org/10.1109/cvpr.2001.990517>

Yulina, S. (2021). Implementation of Haar Cascade Classifier for Face Detection and Grayscale Image Transformation Using OpenCV. *Jurnal Komputer Terapan*, 7(1), 100–109. <https://doi.org/10.35143/jkt.v7i1.3411>