

Verifikasi Wajah untuk Menghitung Jumlah Transaksi Pengunjung Menggunakan *Deep Metric Learning*

Rifqi Affan Maulana¹, Riyanto Sigit², Setiawardhana³

^{1,2,3}Politeknik Elektronika Negeri Surabaya, Jl. Raya ITS - Kampus PENS, Surabaya, 60111, Indonesia

Info Artikel

Riwayat Artikel:

Received 2025-06-08

Revised 2025-06-24

Accepted 2025-06-28

Corresponding Author:

Rifqi Affan Maulana

Email: affanrifqi555@gmail.com



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

Abstract – This research carries the theme of facial recognition to detect visitors' faces by counting the number of times visitors make transactions. The objective of this research is to develop and implement a face verification system for public purposes, such as commercial purposes. One potential application of this system is in the realm of promotions, where it could be utilized to track the number of transactions conducted by visitors. The method employed utilizes deep metric learning (DML) to generate a model capable of verifying various facial images through the Convolutional Neural Network (CNN) architecture, which is designed to train human face image data. The triplet loss method is employed in training data due to its recognition as a more flexible approach in utilizing labels (in the form of face images) to facilitate comparison with the detected face images. The model employed for face recognition applications is facenet, a system that has been demonstrated to achieve a high degree of accuracy. The research's output is an application capable of swiftly and precisely verifying facial images of visitors and calculating the number of visitor transactions. The number of visitor transactions can subsequently be utilized as a promotional or discount strategy in commercial services.

Keywords: Deep Metric Learning, Face Verification, CNN, Facenet

Abstrak – Penelitian ini mengungkap tema pengenalan wajah untuk mendeteksi wajah pengunjung dengan cara menghitung berapa kali pengunjung yang melakukan transaksi. Urgensi penelitian ini adalah untuk memanfaatkan teknologi pengenalan wajah untuk verifikasi citra wajah. Tujuan dari penelitian ini adalah mengembangkan dan mengimplementasikan sistem verifikasi wajah untuk kepentingan umum seperti tujuan komersial misalnya untuk kepentingan promosi dengan cara memberikan bonus untuk seberapa banyak transaksi pengunjung. Metode yang digunakan menggunakan deep metric learning (DML) untuk membuat model yang bisa memverifikasi wajah yang berbeda melalui arsitektur Convolutional Neural Network (CNN) yang berfungsi untuk training data citra wajah manusia. Training data menggunakan metode triplet loss karena dinilai lebih fleksibel dalam penggunaan label (disini berupa citra wajah) untuk membandingkan dengan citra wajah yang dideteksi. Model yang digunakan untuk aplikasi pengenalan wajah adalah facenet karena dinilai memiliki akurasi yang cukup tinggi. Hasil luaran penelitian berupa aplikasi yang dapat memverifikasi citra wajah secara cepat dan akurat serta menghitung jumlah transaksi pengunjung. Jumlah transaksi pengunjung nantinya dapat digunakan sebagai promo atau diskon dalam layanan komersil.

Kata Kunci: Deep Metric Learning, Face Recognition, CNN, dan Facenet

I. PENDAHULUAN

Pengenalan wajah atau biasa dikenal dengan *face recognition* adalah sistem pengenalan dengan menerapkan pendekatan biometrik yang menggunakan metode verifikasi atau mengenali identitas citra wajah seseorang sehingga sistem dapat membedakan citra wajah yang berbeda[1]. Penggunaan teknologi *face recognition* banyak digunakan dalam fitur pengenalan wajah smartphone, contohnya digunakan dalam sistem absensi menggunakan pengenalan wajah dan lain-lain yang berkaitan dengan *face recognition*. Keunggulan dalam *face recognition* cenderung menjadi efisien, akurat, dan cerdas karena pengaruh otomatisasi komputer[2]. Manfaat dari teknologi ini yaitu peningkatan keamanan dalam suatu sistem, Penggunaan yang efisien dan cepat, dan implementasi yang fleksibel dengan menerapkan pada sistem yang berbasis verifikasi wajah. Algoritma pendeteksian wajah yaitu dengan menemukan koordinat fitur pada wajah seperti letak mata, hidung, dan mulut, lalu mencocokkan fitur tersebut dengan citra yang ingin diidentifikasi. Penggunaan Teknologi *face recognition* dapat diimplementasikan sebagai sistem verifikasi wajah untuk mengenali wajah dengan presentase keakuratan sebesar 95% untuk mengenali wajah yang sudah tersimpan dalam sistem[3]. Kegunaan teknologi *face recognition* pada zaman sekarang banyak dibutuhkan salah satunya untuk kepentingan verifikasi wajah pada bidang komersial. Beberapa contoh yaitu pemanfaatan *face recognition* dalam memudahkan dan mempercepat *boarding* penumpang pada stasiun kereta api dan penilaian *rating* pada restoran dengan memanfaatkan ekspresi wajah pengunjung dengan memberi nilai berdasarkan ekspresi oleh pengunjung lalu hasil tersebut diunggah pada halaman restoran di *Google Maps* [4][5].

Penelitian ini menggunakan metode *Deep Metric Learning* (DML) untuk pengenalan wajah. Metode ini merupakan gabungan dari teknologi *deep learning* dan *metric learning*[6] yang bertujuan mengembangkan metode yang untuk mengukur kesamaan antara sampel data dengan mempelajari fungsi representasi yang memetakan sampel data ini ke dalam ruang *embedding* yang representatif [7]. Dalam DML terdapat fungsi *loss* (*loss function*) yang memungkinkan memungkinkan generalisasi untuk kelas yang tidak terlihat dengan menyematkan *embedding* fitur yang serupa secara berdekatan dan *embedding* fitur yang berbeda secara berjauhan, daripada secara langsung memetakan ke label kelas. Salah satu fungsi *loss* dalam DML yaitu fungsi *triplet loss* yang mempunyai dengan mendorong fitur *embedded* yang sama dan menjauhkan fitur *embedded* yang berbeda dengan menggunakan triplet data, yaitu *Anchor* sebagai referensi, *Positive* sebagai referensi yang sama, dan *Negative* sebagai referensi yang berbeda. Fungsi *triplet loss* dinilai secara langsung mencerminkan apa yang ingin dicapai dalam verifikasi, pengenalan, dan pengelompokan wajah diakrenakan fungsi *triplet loss* mencoba memaksakan *margin* antara setiap pasangan wajah dari satu orang dengan semua wajah lainnya. Hal ini memungkinkan wajah-wajah untuk satu identitas hidup dalam sebuah jenis, sementara masih menerapkan jarak dan dengan demikian diskriminasi terhadap identitas lainnya[8]. Penerapan DML dalam verifikasi wajah menghasilkan beberapa inovasi. Penelitian pada verifikasi wajah bermasker dinilai sulit karena fitur pada wajah tidak terlihat sehingga menyebabkan sistem *face recognition* sederhana tidak dapat mengenali. Penelitian ini menggunakan model yang menerapkan metode membandingkan fitur *embedded* wajah yang bermasker dengan wajah yang tidak bermasker yang sudah tersimpan di dalam database. Hasil menunjukkan akurasi dalam memprediksi wajah sebesar 88,92%. Selain itu waktu yang dibutuhkan untuk mengenali wajah sebanyak kurang dari 10ms membuat model tersebut bisa digunakan dalam rekaman video langsung (*live footage*)[9]. Penelitian lain yaitu mengenai sistem absensi dengan menerapkan DML untuk klasifikasi citra seseorang[10]. Penerapan sistem berupa sistem absensi pada siswa otomatis dimana sistem akan mengambil fitur wajah siswa. Fitur yang diambil kemudian dibandingkan dengan fitur yang telah disimpan kedalam database. Apabila fitur wajah yang diambil sama dengan salah satu fitur di dalam database maka sistem akan mencatat kehadiran siswa yang sesuai dengan pemilik fitur yang diambil. Hasil pengambilan data absensi dapat diambil berupa data *excel*.

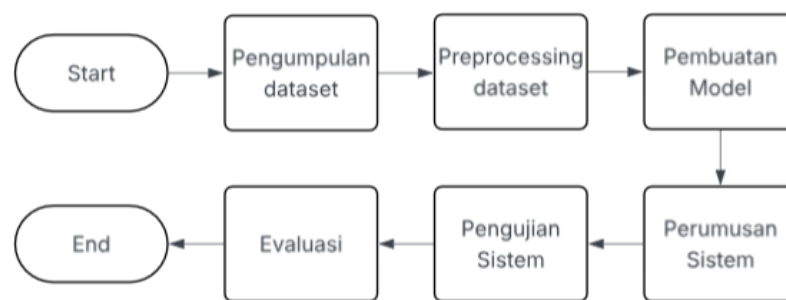
Penerapan DML kedalam sebuah sistem verifikasi wajah dibutuhkan sebuah model yang dapat melakukan *data learning* pada citra wajah manusia. *Convolutional Neural Network* atau CNN adalah algoritma *neural network* yang digunakan untuk memproses input berupa data citra wajah, menentukan kepentingan (bobot dan bias yang dipelajari) ke berbagai aspek dalam citra wajah dan berfungsi untuk membedakan objek satu dengan objek lainnya[11]. Salah satu metode *learning* yaitu menggunakan *transfer learning*. Metode ini bertujuan untuk meningkatkan *learning* pada model dengan cara transfer pengetahuan *learning* yang berbeda pada bidang yang sama [12]. Metode ini memungkinkan model yang digunakan tidak melakukan *training* secara manual karena proses *transfer learning* Dalam penelitian tentang perbandingan *transfer learning* dengan *training* dari nol, hasil ketepatan *transfer learning* jauh diunggulkan karena model bisa mengekstraksi secara lebih cepat dan tepat [13]. *Facenet* merupakan model *deep learning* yang bekerja dengan ekstraksi fitur sebuah citra wajah lalu membandingkan dengan kemiripan jarak antara wajah, jadi tidak klasifikasi secara langsung tetapi memetakan ke dalam suatu ruang[8]. Cara kerja *facenet* yaitu menerapkan *triplet loss* dengan melatih model untuk bisa membedakan jarak suatu fitur, apabila jarak antara fitur berdekatan maka akan dianggap orang yang sama dan jika jarak antara fitur berjauhan maka akan dianggap orang yang berbeda. Tingkat akurasi *facenet* pada dataset LFW dibandingkan model lain sebanyak 99,63% dengan kesalahan sebanyak 22 dari 6000 pasang dataset citra[14]. Model yang telah ditraining akan diimplementasikan kedalam sebuah aplikasi untuk mengenali citra wajah. *OpenCV* merupakan *library* pemrograman *open source* yang biasanya digunakan untuk memproses citra. *OpenCV* memiliki beberapa fitur yaitu *image filtering* yang bertujuan untuk memodifikasi atau memperkuat citra, *image transformation* untuk membuat gambar baru dari dua atau lebih sumber yang bertujuan menyorot suatu fitur tertentu, *object tracking* yang bertujuan mencari objek dalam citra, dan *feature detection* untuk mencari ciri khas dari suatu citra[15]. Dalam pengolahan citra wajah digunakan metode untuk mendeteksi bagian wajah untuk mengurangi kesalahan sistem untuk mengenali fitur suatu citra. MTCNN atau Multi-Task Cascaded Convolutional Network adalah perombakan dari gabungan model CNN dengan 3 lapisan jaringan yang mengelompokkan kandidat dan klasifikasi untuk mendapatkan hasil deteksi wajah yang efisien dan cepat[16]. MTCNN dinilai secara konsisten memberikan hasil deteksi wajah yang bagus dalam menyelesaikan perbedaan permasalahan deteksi citra wajah seperti pengujian kondisi wajah melihat kedepan, posisi jauh dari kamera sepanjang 1,5 meter, dan posisi menghadap kebawah[17].

Penelitian ini bertujuan untuk mengembangkan sistem yang mengimplementasikan metode DML dalam pengenalan wajah untuk kepentingan komersil seperti menghitung jumlah transaksi. Pengenalan wajah dalam sistem ini akan menerapkan fungsi *triplet loss* dalam DML untuk mengambil fitur khas wajah seperti mulut, mata, hidung, dan lain-lain yang berkaitan dengan citra wajah seseorang. fitur-fitur khas citra wajah yang diambil tadi akan akan disimpan dalam *embedding file* yang nantinya akan digunakan untuk model deteksi wajah. Proses deteksi wajah dengan cara membandingkan jarak nilai *embedding*. Jika dalam proses deteksi citra wajah menunjukkan nilai *embedding* yang berdekatan atau hampir sama maka citra tersebut akan dianggap sama

sedangkan apabila nilai *embedding* jauh maka akan dianggap citra yang berbeda. Model sistem ini akan di-*training* menggunakan metode *transfer learning* yang bertujuan untuk mempercepat proses dan menghasilkan hasil yang akurat. Kelebihan lain dari metode ini adalah kemampuan model yang bisa digunakan tanpa di-*training* secara manual. Sistem akan dibuat menggunakan *library OpenCV* untuk mendukung pengolahan citra sistem. Algoritma MTCNN akan digunakan untuk deteksi citra wajah manusia secara akurat dan *FaceNet* digunakan untuk ekstraksi fitur khas wajah menjadi nilai *embedding* dan melakukan perbandingan nilai *embedding* tersebut. Dalam penelitian ini, sistem ini akan diterapkan pada bidang komersial seperti pada kasir restoran cepat saji yang dimana pengunjung antri secara berbaris. Cara kerja sistem ini yaitu dengan menghitung wajah pengunjung pada setiap pengunjung saat akan melakukan transaksi. Wajah pengunjung nantinya akan diverifikasi lalu menampilkan jumlah kunjungan berupa seberapa banyak transaksi pengunjung. Jumlah kunjungan tersebut nantinya bisa digunakan untuk tujuan komersil contohnya seperti promo atau diskon makanan yang dibeli. Penerapan sistem ini memungkinkan inovasi penerapan teknologi pengenalan wajah dalam sektor komersial. Kebaruan dalam penelitian pengenalan wajah ini dapat membantu dalam penerapan tujuan komersil seperti diskon atau promosi dengan metode sederhana seperti verifikasi wajah pengunjung dan menghitung jumlah berapa kali bertransaksi. Diharapkan juga dari penelitian ini menjadi terobosan penerapan metode DML dalam verifikasi wajah pada bidang komersial.

II. METODE

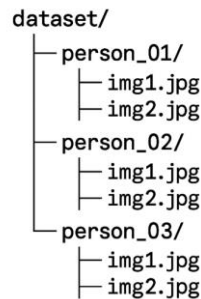
Untuk menyelesaikan penelitian ini dilakukan beberapa tahapan penelitian yang bertujuan memudahkan penelitian dan pembuatan sistem. Tahapan penelitian ini meliputi pengumpulan dataset, *preprocessing* dataset, pembuatan model, pembuatan sistem, implementasi sistem, evaluasi. Tahapan penelitian dapat dilihat pada Gambar 1.



Gambar 1. Tahapan metode penelitian

A. Pengumpulan Dataset

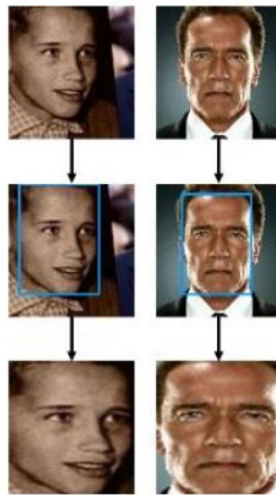
Pada tahapan pengumpulan dataset untuk penelitian berupa citra wajah manusia. Untuk mendapatkan kualitas citra wajah yang bagus, pengambilan citra wajah dilakukan dengan memperhatikan aspek-aspek pengambilan seperti cahaya yang cukup, sudut pengambilan citra, dan resolusi kamera yang tinggi. Citra yang diambil merupakan bagian wajah tampak depan karena memiliki banyak fitur khas pada citra wajah seseorang. Citra wajah yang digunakan pada penelitian ini berjumlah sebanyak sembilan kelas. Kelas tersebut berupa label atau folder yang berisi citra wajah seseorang. Jumlah label yang digunakan sebanyak sembilan buah dengan isi sebanyak 10 buah citra seseorang dengan total sebanyak 90 citra wajah. Citra diambil dengan menggunakan *webcam* dan disimpan dalam format gambar *.jpeg* dan resolusi gambar dibiarkan seperti saat pengambilan citra. Hal ini bertujuan untuk menjaga kualitas dataset agar sistem mudah mengenali wajah. Pengambilan dataset dilakukan dalam ruangan dengan cukup cahaya dan sudut kamera yang merekam semua bagian depan wajah. Struktur dalam pengumpulan dataset bisa dilihat dalam gambar 2



Gambar 2. struktur pada dataset

B. Preprocessing Dataset

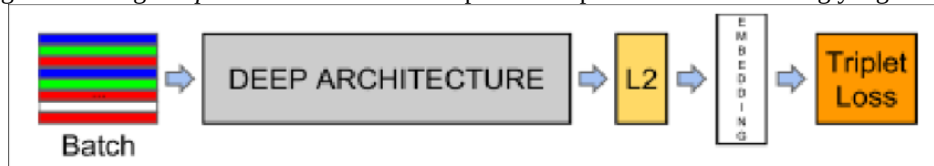
Tahapan ini bertujuan menerapkan metode *preprocessing* untuk dataset yang bertujuan untuk memfilter dataset dari elemen citra yang tidak dibutuhkan seperti elemen selain citra wajah. Proses ini memengaruhi kinerja model dengan meningkatkan akurasi, konvergensi cepat, dan pengelompokan kelas lebih baik [18]. Proses ini menerapkan algoritma MTCNN untuk mendeteksi bagian wajah untuk digunakan di dalam model dan sistem. MTCNN dapat mendeteksi dan menandai bagian wajah yang dideteksi secara akurat seperti bagian hidung, mulut dan mata. Pada gambar 3 merupakan proses *preprocessing* pada dataset.



Gambar 3. Proses alur tahapan preprocessing

C. Pembuatan Model

Model dibuat dari hasil *training data* pada dataset yang telah melalui tahap *preprocessing*. Dataset tersebut akan dilatih menggunakan jaringan syaraf tiruan dengan menerapkan algoritma *deep metric learning* yang menggunakan fungsi *triplet loss*. Gambar 4 merupakan tahapan dalam data training yang dilakukan.



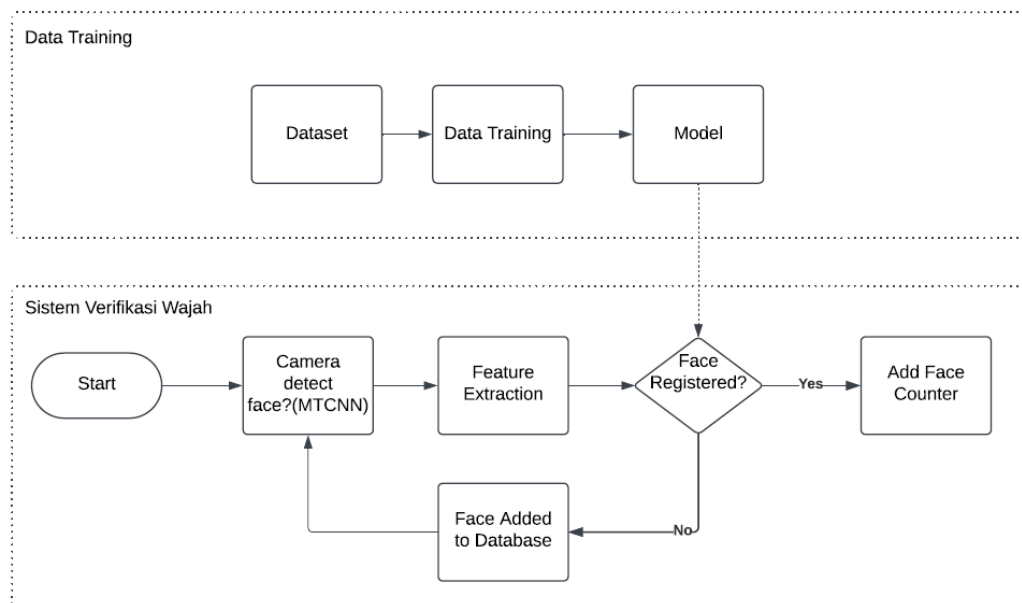
Gambar 4. Tahapan data training

Tahap ini menerapkan *pretrained* model *facenet* sebagai basis model untuk proses *training*. Dataset yang telah melalui proses tahap *embedding* akan menghasilkan berupa vektor 128-d. *Embedding* tersebut berisi data-data ciri khas citra wajah pada dataset. Data yang dihasilkan dari *embedding* akan disimpan sebagai Model. Model digunakan untuk membandingkan dengan *embedding* lain pada sistem dengan cara membandingkan menghitung jarak kedekatan antara vektor positif (jarak terdekat) dan negatif (jarak terjauh).

D. Pembuatan Sistem

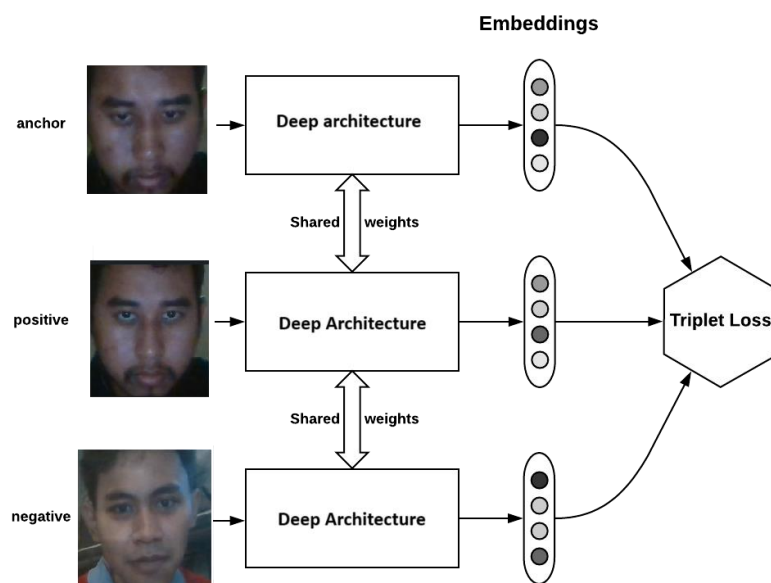
Sistem ini bekerja dengan cara verifikasi wajah dan menambahkan berapa kali jumlah terdeteksi Sistem dibuat dengan bahasa pemrograman *python* yang dilengkapi dengan *library OpenCV* dan *PyTorch* yang

membantu dalam fitur pengenalan wajah. Sistem ini dijalankan pada komputer yang telah diintegrasikan pada kamera *webcam* dengan resolusi 1080p. Model yang telah dibuat akan diterpkan sebagai basis database untuk verifikasi wajah. Proses cara kerja sistem verifikasi wajah dapat dilihat pada gambar 5.



Gambar 5. flowchart cara kerja sistem

Pada proses awal yaitu adalah tahapan mendeteksi wajah. Proses deteksi wajah menerapkan algoritma MTCNN yang berfungsi untuk memetakan daerah citra yang akan dideteksi. Daerah yang dideteksi berupa bagian depan wajah seseorang. Setelah mendeteksi wajah, sistem akan melanjutkan ke tahap *face extraction* yaitu proses pengambilan nilai fitur khas wajah dengan cara *embedding* fitur wajah sama seperti yang dilakukan pada tahapan pembuatan model dengan bantuan *facenet*. Setelah nilai *embedding* tercatat, sistem akan melanjutkan ke tahap verifikasi wajah. Pada tahap ini nilai *embedding* dari wajah yang terdeteksi akan dibandingkan dengan nilai yang ada pada model *database*. Jika jarak pada *embedding* wajah tersebut dekat dengan *embedding* database, maka wajah tersebut akan dianggap *possitive* atau mirip, sedangkan apabila jarak *embedding* jauh dari dataset maka akan dianggap *negative* atau tidak mirip. Gambar 6 merupakan cara verifikasi wajah dengan membandingkan jarak nilai *embedding*.



Gambar 6. ilustrasi perbandingan nilai embedding

Jika hasil verifikasi wajah dinyatakan mirip maka sistem akan menambahkan jumlah seberapa kali wajah terdeteksi pada sistem berupa *counter* kunjungan. Apabila wajah dinyatakan tidak sama, sistem bisa menambahkan citra wajah tersebut ke dalam sistem lalu melakukan tahapan deteksi wajah ulang. *Counter* kunjungan akan dianggap sebagai berapa banyak transaksi seseorang lakukan. Luaran sistem berupa hasil deteksi wajah dan jumlah transaksi

E. Pengujian Sistem

Sistem ini akan diuji dengan contoh kasus pada antrian berbaris di restoran cepat saji. Peralatan yang digunakan yaitu komputer yang telah terpasang kamera di depan tempat transaksi dan monitor komputer untuk menampilkan hasil kunjungan berupa label pengunjung dan jumlah transaksi. Setiap transaksi dilakukan oleh satu pengunjung. Oleh karena itu sistem hanya mendeteksi antrian paling depan yang sedang bertransaksi. Sistem diatur dengan cara hanya deteksi wajah kurang dari satu meter yang bertujuan agar sistem tidak mendeteksi dan mencatat wajah diluar yang bertransaksi. Hasil luaran sistem akan ditampilkan pada layar berupa label dari deteksi wajah dan jumlah transaksi. Jumlah transaksi akan dijadikan untuk kepentingan promosi komersial seperti pemberian potongan harga.

F. Evaluasi

Tahapan evaluasi bertujuan untuk mendapatkan nilai seberapa akurat model dalam memverifikasi citra wajah dan hasil pada pengujian sistem. Evaluasi model berupa prediksi pada satu label dengan membandingkan dengan label lain. Pendekatan standar dalam pengenalan wajah yaitu dengan menghitung evaluasi berdasarkan deteksi positif dan negatif. Pada gambar 7 merupakan tampilan *confusion matrix*

Ground truth\detection	Detected (Positive)	Rejected (Negative)
Relevant	True positive (TP)	False negative (FN)
Nonrelevant	False positive (FP)	True negative (TN)

Gambar 7. tampilan confusion matrix.

Tabel tersebut berupa perbandingan untuk membedakan pasangan citra yang sama dengan tidak sama. TP (*true positive*) berisi perbandingan pasangan citra yang sama yang didapatkan dengan perbandingan citra dalam satu kelas sedangkan *false positive*(FP) merupakan perbandingan suatu kelas citra dengan kelas lain. FN(*false negative*) menyatakan kegagalan prediksi label berbeda untuk label sama sedangkan TN(*true negative*) yaitu prediksi label yang harusnya sama untuk label berbeda. Beberapa cara evaluasi untuk menentukan kualitas model. Akurasi merupakan evaluasi utama untuk mengukur tingkat ketepatan dalam verifikasi citra, Formula 1 untuk menghitung akurasi:

$$\text{Akurasi} = \frac{(TP+TN)}{(TP + TN + FP + FN)}, \quad (1)$$

Precision merupakan cara untuk menghitung seberapa besar prediksi positif dengan mengukur prediksi TP dengan total semua prediksi positif sehingga model tidak salah dalam mengenali wajah asing yang tidak terdaftar, seperti pada formula 2

$$\text{Precision} = \frac{TP}{(TP + FN)}, \quad (2)$$

Recall banyak pasangan wajah yang seharusnya dikenali (positif) berhasil terdeteksi dengan benar oleh model. Apabila *recall* tinggi maka model jarang melewatkan wajah yang dikenali, seperti pada formula 3

$$\text{Recall} = \frac{TP}{(TP + FP)}, \quad (3)$$

F1 score digunakan untuk menyeimbangkan nilai *precision* dan *recall*. Jika F1 tinggi maka sistem, seperti pada formula 4

$$\text{F1 - Score} = \frac{TP}{(TP + FN)}. \quad (4)$$

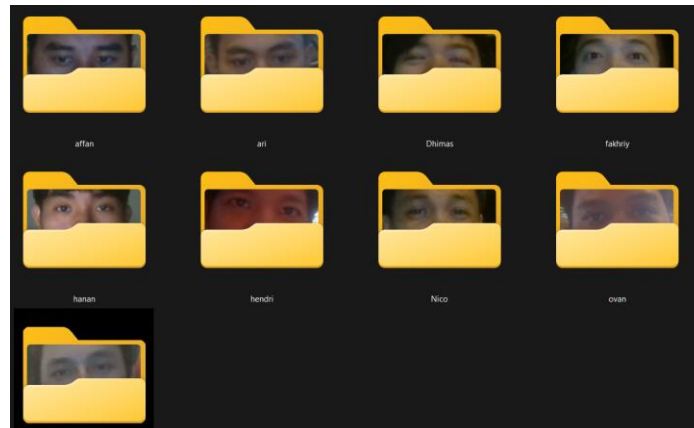
Selain evaluasi pada model, evaluasi juga dilakukan pada pengujian sistem. Evaluasi pada sistem ini untuk analisis fitur pada sistem seperti pendeteksian wajah, verifikasi wajah, dan penambahan counter pada citra wajah seseorang. Pengujian diharapkan sistem dapat diterapkan dalam keperluan komersial.

III. HASIL DAN PEMBAHASAN

Hasil dari penelitian ini adalah tahapan yang dilakukan dalam penelitian serta model pengenalan wajah dan sistem untuk verifikasi wajah untuk menghitung banyak transaksi.

A. Dataset dan Preprocessing,

Dataset yang diambil berupa kumpulan dari wajah manusia sebanyak 9 orang lalu dikelompokkan sesuai label masing-masing. Setiap label memiliki jumlah foto sebanyak 10 buah. Dataset diambil menggunakan *webcam*. Dataset kemudian memasuki proses *preprocessing* untuk menghasilkan citra berupa bagian wajah seseorang saja. Hasil tersebut akan disimpan dalam folder *preprocessing*. Pada gambar 8 merupakan kumpulan dataset dan gambar 9 merupakan hasil beberapa contoh hasil *preprocessing*



Gambar 8. Kumpulan dataset yang diambil



Gambar 9. beberapa hasil *preprocessing* dari dataset

B. Hasil Pembuatan Model

Dataset pada yang telah melalui tahapan *preprocessing* akan dijadikan dalam bentuk vektor *embedding* menggunakan *pretrained* model yaitu *facenet*. Embedding tersebut disimpan dalam file *.pt* untuk dilanjutkan ke proses evaluasi model. Gambar 10 merupakan hasil dari *embedding*


```

1. Dhimas ➤ tensor([ 0.0900, -0.0012,  0.0405,  0.0395,  0.0121])...
2. Dhimas ➤ tensor([ 0.0801, -0.0108,  0.0093,  0.0325,  0.0198])...
3. Dhimas ➤ tensor([0.0897, 0.0142, 0.0503, 0.0274, 0.0356])...
4. Dhimas ➤ tensor([0.0903, 0.0017, 0.0253, 0.0117, 0.0331])...
5. Dhimas ➤ tensor([ 0.0707, -0.0025,  0.0287,  0.0243,  0.0501])...
6. Dhimas ➤ tensor([0.0735, 0.0082, 0.0400, 0.0272, 0.0396])...
7. Dhimas ➤ tensor([0.0752, 0.0073, 0.0391, 0.0289, 0.0265])...
8. Dhimas ➤ tensor([0.0821, 0.0011, 0.0307, 0.0214, 0.0451])...
9. Dhimas ➤ tensor([0.0890, 0.0014, 0.0372, 0.0178, 0.0172])...
10. Dhimas ➤ tensor([0.0646, 0.0159, 0.0174, 0.0391, 0.0460])...
11. Nico ➤ tensor([-0.0031,  0.0525, -0.0459,  0.0232,  0.0415])...
12. Nico ➤ tensor([ 0.0154,  0.0488, -0.0251,  0.0163,  0.0301])...
13. Nico ➤ tensor([ 0.0111,  0.0454, -0.0227,  0.0272,  0.0333])...
14. Nico ➤ tensor([ 0.0140,  0.0355, -0.0386,  0.0279,  0.0472])...
15. Nico ➤ tensor([ 0.0103,  0.0446, -0.0318,  0.0285,  0.0598])...
16. Nico ➤ tensor([-0.0038,  0.0394, -0.0510,  0.0163,  0.0624])...
17. Nico ➤ tensor([ 0.0022,  0.0481, -0.0314,  0.0175,  0.0532])...
18. Nico ➤ tensor([ 0.0030,  0.0454, -0.0263,  0.0259,  0.0375])...
19. Nico ➤ tensor([ 0.0203,  0.0555, -0.0056,  0.0222,  0.0157])...
20. Nico ➤ tensor([ 0.0253,  0.0458, -0.0272,  0.0356,  0.0419])...
21. affan ➤ tensor([ 0.0808, -0.0064,  0.0013,  0.0329,  0.0122])...
22. affan ➤ tensor([ 0.0537,  0.0201, -0.0149,  0.0560,  0.0116])...
23. affan ➤ tensor([ 0.0599,  0.0294, -0.0061,  0.0554, -0.0069])...
24. affan ➤ tensor([0.0700, 0.0151, 0.0055, 0.0546, 0.0170])...
25. affan ➤ tensor([ 0.0759,  0.0291,  0.0065,  0.0626, -0.0136])...
26. affan ➤ tensor([ 0.0626,  0.0138, -0.0044,  0.0742, -0.0140])...
27. affan ➤ tensor([ 0.0683,  0.0293,  0.0128,  0.0561, -0.0036])...
28. affan ➤ tensor([0.0655, 0.0298, 0.0031, 0.0634, 0.0038])...
29. affan ➤ tensor([ 0.0595,  0.0108, -0.0094,  0.0555,  0.0210])...
30. affan ➤ tensor([ 0.0606,  0.0124, -0.0020,  0.0563,  0.0110])...
31. affan ➤ tensor([ 0.0815,  0.0196, -0.0063,  0.0751,  0.0148])...

```

Gambar 10. Hasil pembuatan *embedding*

C. Hasil Pembuatan Sistem

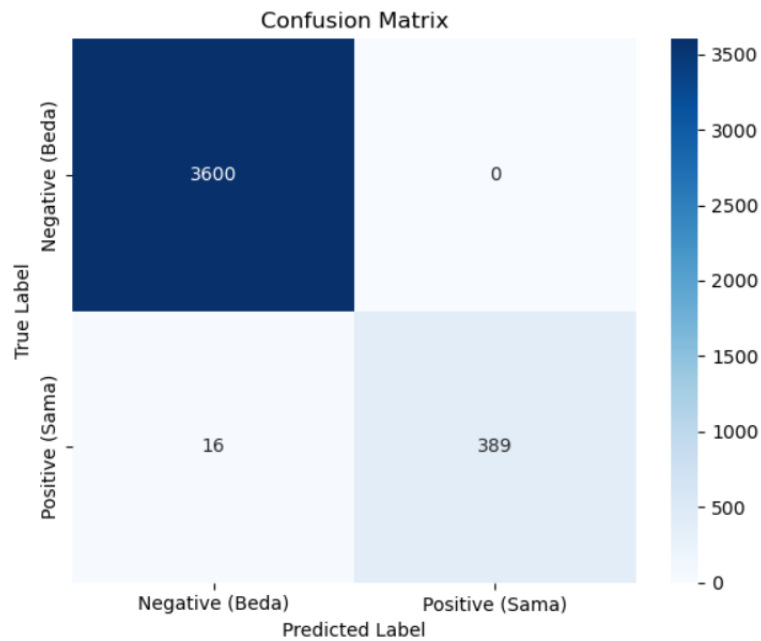
Sistem akan dijalankan dalam komputer yang telah diintegrasikan dengan *webcam*. Input dalam sistem yaitu wajah yang telah direkam secara langsung oleh *webcam*. Input tersebut akan diterima oleh sistem untuk diolah untuk proses verifikasi wajah dan menambahkan *counter* transaksi. Output sistem akan ditampilkan pada display berupa tampilan tangkapan kamera beserta keterangan pemilik wajah dan jumlah *counter* transaksi. Spesifikasi hardware dan software dalam menjalankan sistem ini dapat dilihat pada tabel 1

TABEL 1
SPESIFIKASI HARDWARE DAN SOFTWARE YANG DIGUNAKAN PADA SISTEM

Peralatan	Spesifikasi
<i>Hardware</i>	Laptop -AMD Ryzen 5600U with radeon graphic -16GB RAM -512 GB SSD Kamera Webcam 1080p
<i>Software</i>	Anaconda Jupiter Notebook OpenCV Pytorch

D. Evaluasi Model

Hasil *Embedding* yang digunakan dalam model memiliki label dengan banyak 9 buah berisi 10 wajah. Dengan total 90 wajah didapatkan kemungkinan 4005 pasangan fitur wajah yang berisi 405 pasangan citra wajah sama dan 3600 pasangan citra berbeda. Setelah model ditraining menghasilkan evaluasi berupa *confusion matrix* pada gambar 7.



Gambar 11. confusion matrix training model

Hasil tersebut menunjukkan bahwa model dapat memprediksi sebanyak 389 pasang citra wajah dalam label sama dan 3600 pasang label wajah berbeda. Kesalahan prediksi terdapat pada prediksi yang seharusnya pasangan sama dianggap berbeda sebanyak 16 pasang. Untuk evaluasi tingkat akurasi model dalam prediksi ada pada label 2:

TABEL 2
HASIL EVALUASI MODEL

	Precision	Recall	F1 Score	Support
Pasangan citra berbeda	1.00	1.00	1.00	3600
Pasangan citra sama	1.00	0.96	0.98	405
Akurasi			0.99	4005
Macro Avg	1.00	0.98	0.99	4005
Weighted Avg	1.00	1.00	1.00	4005

Hasil evaluasi model menghasilkan nilai precision sebanyak 100%, recall sebanyak 98%, F1 score sebanyak 99% dan akurasi sebanyak 99%. Dari hasil evaluasi akurasi model menunjukkan bahwa model dapat mengidentifikasi citra wajah pada label yang sama dan wajah label berbeda dengan akurat dan sangat sedikit kesalahan. Selain itu model *facenet* juga dibandingkan dengan *pretrained* model lain seperti *OpenFace* dan *SphereFace* dalam hal akurasi dengan menggunakan data yang sama. Hasil perbandingan akurasi dapat dilihat pada tabel 3

TABEL 3
PERBANDINGAN AKURASI PRETRAINED MODEL

	Support
Facenet	99,60%
Openface	96,93%
Sphereface	89,99%

Dari hasil perbandingan pada 90 citra wajah dengan dua model lain dengan metode yang masing-masing memiliki metode berbeda, *facenet* memiliki akurasi paling tinggi sebesar 99,60%. Hal ini membuktikan bahwa *facenet* dapat menghasilkan model yang cukup bagus dengan dataset yang terbatas.

E. Evaluasi pengujian sistem

Sistem yang telah diuji dapat menampilkan sebuah jendela baru sebagai tampilan luaran deteksi wajah. Waktu yang dibutuhkan sistem untuk mendeteksi suatu citra wajah rata-rata sebanyak 0,2 – 0,3 ms sehingga membuat transaksi tidak lama. Wajah yang tidak terdaftar dalam database akan diberi tanda berupa tulisan

“unknown”. Pengunjung dapat mendaftarkan wajah pada sistem jika diinginkan. Untuk wajah yang sudah terdaftar dalam *database* sistem akan terdeteksi dan diberikan tanda nama label untuk citra yang sedang transaksi, sedangkan untuk yang belum transaksi akan diberikan tanda “too far”. Mekanisme transaksi ini yaitu untuk citra wajah yang berada dalam jarak deteksi kurang dari satu meter dan menghadap depan. Mekanisme ini diterapkan untuk memudahkan sistem untuk mencatat *counter* jumlah transaksi agar tidak menambahkan *counter* pada wajah dibelakang seseorang yang sedang transaksi. Pada proses penghitungan *counter*, wajah yang ditandai dengan “too far” tidak akan ditambahkan nilai *counternya* karena dianggap belum bertransaksi. Hasil dari *counter* transaksi bisa dibuat untuk kepentingan komersil seperti memberikan diskon atau promo berdasarkan jumlah *counter* berkunjung. Hasil deteksi dan verifikasi dapat dilihat pada gambar 12.



Gambar 12. pengujian sistem dengan deteksi foto wajah (a) belum terdaftar (b) terdaftar dan diluar batas deteksi (c) terdaftar dan dalam batas deteksi , dan (d) pengujian pada dua wajah dengan jarak berbeda

Dalam pengujian didapatkan beberapa kekurangan dalam sistem. Kekurangan tersebut yaitu tidak mendeteksi wajah jika tidak menghadap depan. Permasalahan ini ditemukan ketika citra yang bertransaksi mengalihkan pandangan dari kamera seperti menoleh kekanan atau kekiri, sistem tidak dapat mengenali karena *database* yang dimasukkan hanya berupa data wajah bagian depan sehingga menyebabkan sistem mendeteksi kembali dan menambahkan *counter* baru. Kekurangan ini dapat diatasi dengan penambahan dataset baru yang meliputi semua bagian wajah

IV. SIMPULAN

Penelitian ini bertujuan untuk menerapkan teknologi pengenalan wajah pada layanan komersil. Sistem ini mengimplementasikan algoritma *Deep Metric Learning* dalam pembuatan model. Tingkat akurasi yang dihasilkan untuk mendeteksi 4005 pasangan wajah sebesar 99,6% memungkinkan model untuk mendeteksi label yang berbeda secara akurat. Implementasi sistem menggunakan algoritma MTCNN yang dinilai mampu mendeteksi wajah secara akurat dan *facenet* untuk mengambil fitur khas dari citra wajah seseorang. Sistem yang dihasilkan dapat mendeteksi wajah secara akurat dan cepat. Sistem dapat merekam wajah seseorang secara langsung dan mencatat *counter* transaksi. *Counter* ini bisa digunakan untuk keperluan komersil seperti pemberian promo atau diskon. Permasalahan dalam deteksi wajah yaitu tidak bisa mendeteksi bagian wajah yang tidak terdaftar sehingga harus menambahkan dataset baru seperti dataset wajah yang direkam dari semua sisi. Penelitian ini dapat dikembangkan dengan mengaplikasikan pada web untuk mengatur sistem seperti penambahan sesi kunjungan dan melihat semua *counter* pengunjung, IoT juga bisa diintegrasikan kedalam sistem ini untuk membuat suatu alat yang bisa memverifikasi wajah secara otomatis dan menghasilkan suara untuk memberitahu pengunjung mengenai transaksi yang dilakukan.

UCAPAN TERIMA KASIH

Ucapan terima kasih kepada program studi S2 Teknik Informatika, Politeknik Elektronika Negeri Surabaya atas segala bantuan yang telah diberikan selama proses penelitian ini serta kedua pembimbing yang selalu membantu dalam bimbingan penelitian ini. Ucapan terima kasih kepada pihak redaksi jurnal yang telah memberikan ruang untuk mempublikasi hasil penelitian saya.

DAFTAR PUSTAKA

- [1] Mu, Xiaohui & Li, Siying & Haipeng, Peng. (2020). A Review of Face Recognition Technology. IEEE Access. PP. 1-1. 10.1109/ACCESS.2020.3011028.
- [2] Wu, Zehao. (2024). Overview of Mainstream Face Recognition Methods. Transactions on Computer Science and Intelligent Systems Research. 7. 390-397. 10.62051/e37s5n84.
- [3] M. Arsal, B. Agus Wardijono, dan D. Anggraini, "Face Recognition Untuk Akses Pegawai Bank Menggunakan Deep Learning Dengan Metode CNN", *TEKNOSI*, vol. 6, no. 1, hlm. 55–63, Jun 2020.
- [4] C. G. Da Rato dan A. R. Adriyanto, "Efektivitas Sistem Face Recognition pada Boarding Gate Stasiun Kota Bandung," *Action Research Literate*, vol. 8, no. 6, pp. 1–7, Jun. 2024, doi: 10.46799/ar.v8i6.392.
- [5] W. -J. Chang *et al.*, "A Deep Learning Facial Expression Recognition based Scoring System for Restaurants," *2019 International Conference on Artificial Intelligence in Information and Communication (ICAIIIC)*, Okinawa, Japan, 2019, pp. 251-254, doi: 10.1109/ICAIIIC.2019.8668998.
- [6] J. Lu, J. Hu and J. Zhou, "Deep Metric Learning for Visual Understanding: An Overview of Recent Advances," in *IEEE Signal Processing Magazine*, vol. 34, no. 6, pp. 76-84, Nov. 2017, doi: 10.1109/MSP.2017.2732900.
- [7] D. D. Mohan, B. Jawade, S. Setlur, and V. Govindaraj, Deep Metric Learning for Computer Vision: A Brief Overview. 2023. [Online]. Available: <https://arxiv.org/abs/2312.10046>
- [8] Florian Schroff, Dmitry Kalenichenko, James Philbin: "FaceNet: A Unified Embedding for Face Recognition and Clustering", 2015; arXiv:1503.03832. DOI: 10.1109/CVPR.2015.7298682.
- [9] Golwalkar, Rucha & Mehendale, Ninad. (2022). Masked-face recognition using deep metric learning and FaceMaskNet-21. Applied Intelligence. 52. 13268-13279. 10.1007/s10489-021-03150-3.
- [10] Bankar, Rutuja & Bargat, Nikita & Hanmante, Isha & Dakore, Prof. (2022). Face Recognition Using Facenet Deep Learning Network for Attendance System. International Journal of Scientific Research in Computer Science, Engineering and Information Technology. 458-463. 10.32628/CSEIT228630.
- [11] Azmi, Khairul & Defit, Sarjon & Sumijan, Sumijan. (2023). Implementasi Convolutional Neural Network (CNN) Untuk Klasifikasi Batik Tanah Liat Sumatera Barat. JURNAL UNITEK. 16. 28-40. 10.52072/unitek.v16i1.504.
- [12] F. Zhuang *et al.*, A Comprehensive Survey on Transfer Learning. 2020. [Online]. Available: <https://arxiv.org/abs/1911.02685>
- [13] Oztel, Ismail & Yolcu, Gözde & Oz, Cemil. (2019). Performance Comparison of Transfer Learning and Training from Scratch Approaches for Deep Facial Expression Recognition. 1-6. 10.1109/UBMK.2019.8907203.
- [14] Learned-Miller, Erik & Huang, Gary & RoyChowdhury, Aruni & Li, Haoxiang & Hua, Gang. (2016). Labeled Faces in the Wild: A Survey. 10.1007/978-3-319-25958-1_8.
- [15] Sinha, Ekta & Kumar, Ashwani & Tyagi, Abhinav. (2025). OpenCV for Computer Vision Applications. International Journal For Multidisciplinary Research. 7. 10.36948/ijfmr.2025.v07i03.44280.
- [16] Anggara, K. D., Kartikasari, D. P., & Bakhtiar, F. A. (2023). Implementasi Algoritma MTCNN dalam Mekanisme Autentikasi berbasis Pengenalan Wajah. *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer*, 7(8), 3613–3621. Diambil dari <https://j-ptiik.ub.ac.id/index.php/j-ptiik/article/view/13044>
- [17] Ryando, Catoer & Sigit, Riyanto & Setiawardhana, Setiawardhana & Dewantara, Bima. (2025). Face Recognition for Logging in Using Deep Learning for Liveness Detection on Healthcare Kiosks. JOIV : International Journal on Informatics Visualization. 9. 295. 10.62527/joiv.9.1.2759.
- [18] D. Tribuana, Hazriani, and A. L. Arda, "Image Preprocessing Approaches Toward Better Learning Performance with CNN", *J. RESTI (Rekayasa Sist. Teknol. Inf.)*, vol. 8, no. 1, pp. 1 - 9, Jan. 2024.