

Rancang Bangun Aplikasi Kehadiran Siswa Menggunakan Pengenalan Wajah Berbasis Android Dengan Metode Machine Learning

¹Iskandar, ²Umar Tsani Abdurahman, ³Joko Nursanto

^{1,2,3}Sekolah Tinggi Teknologi Muhammadiyah Cileungsi, Indonesia

iskandar@sttcileungsi.ac.id; umartsani@sttcileungsi.ac.id; jokonursanto@sttcileungsi.ac.id;

Article Info

Article history:

Received, 2022-10-02

Revised, 2022-11-09

Accepted, 2022-11-26

Kata Kunci:

sistem kehadiran
machine learning
tensorflow
android
k-nearest_neighbor

Keywords:

attendance system
machine_learning
tensorflow
android
k-nearest_neighbor

ABSTRAK

Machine learning merupakan suatu pendekatan dalam *Artificial Intelligence* yang banyak diaplikasikan dalam menirukan perilaku manusia untuk penyelesaian masalah secara otomatisasi. Pemanfaatan *machine learning* salah satunya *face recognition* atau pengenalan wajah dalam proses kehadiran siswa yang bisa mengenali kehadiran secara *realtime* menggunakan pengolahan data wajah dari siswa. Sistem kehadiran siswa yang berjalan selama ini masih dilakukan secara manual oleh guru dengan mencatatkan pada daftar hadir siswa, dimana dalam pelaksanaannya kadangkala terjadi kesalahan tulis serta membutuhkan waktu relatif lama dalam proses pencatatannya. Seiring dengan kemajuan jaman pengelolaan sistem kehadiran siswa, manajemen membutuhkan sistem yang fleksibel, tepat guna dan aman terhadap perubahan kebutuhan manajemen sekolah. Dalam penelitian ini dirancang sistem kehadiran siswa berbasis *mobile android* dengan menggunakan *machine learning* untuk sistem kehadiran dengan menggunakan pengenalan wajah menggunakan metode algoritma *K-Nearest Neighbor* dengan *library Python 3.8*, *TensorFlow Lite* dan sistem manajemen *database MySQL*. Dari hasil implementasi dan pengujian model menghasilkan nilai akurasi 94.66%, presisi 94.41%, recall 94.20%, 1 k-fold score 94.19, untuk pengujian dari sisi aplikasi BackEnd dan FrontEnd menunjukkan kecepatan dan keakuratan sudah memadai untuk dapat digunakan pada satu institusi kelas atau sekolah. Kestabilan *server face_api* juga cukup memadai selama testing dapat berjalan tanpa gangguan lebih dari 7x24 jam.

ABSTRACT

Machine learning is a method of artificial intelligence that is frequently used to automatically solve problems by mimicking human behavior. Face identification, also known as facial recognition, is one application of machine learning that is used to track student attendance. It can identify students' presence in real time by processing their facial data. The current student attendance system still requires the teacher to manually enter the information on a student attendance list, which can lead to writing errors and take a considerable amount of time. A adaptable system that is efficient and secure against the shifting demands of school management is required to keep up with the development of student attendance management systems. This research uses an android this study used the K-Nearest Neighbor algorithm, the Python 3.8 library, TensorFlow Lite, and the MySQL database management system to develop an android mobile-based student attendance system that uses facial recognition. According to the model's implementation and testing results, which include accuracy values of 94.66%, 94.41% precision, 94.20% recall, and an 1 k-fold score of 94.19 for testing from the BackEnd and FrontEnd application side, it can be used in one class or educational setting with enough speed and accuracy. As long as testing can continue uninterrupted for more than seven times a day, the face api server's stability is also fairly good.

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



Penulis Korespondensi:

Iskandar,
Program Studi Teknik Informatika,
Sekolah Tinggi Teknologi Muhammadiyah Cileungsi,
Email: iskandar@sttcileungsi.ac.id

1. PENDAHULUAN

Pemanfaatan teknologi komputer saat berkembang sangat pesat dalam hal pengontrolan kedisiplinan siswa selain dalam rangka mengenalkan penerapan teknologi digital pada siswa tentunya juga menjadi ajang promosi bagi sekolah bersangkutan terkait pemanfaatan teknologi komputer berbasis mobile salah satunya tata kelola absensi.

Pada suatu manajemen sekolah xyz pengelolaan absensi siswa masih dengan pencatatan secara manual oleh guru pada daftar hadir siswa, masalah ditemukan saat proses pencatatan yang relatif lama dan kadangkala terjadi kesalahan tulis serta membutuhkan waktu lama dalam rekapitulasi serta membutuhkan ruang untuk penyimpanan datanya. Manajemen sekolah membutuhkan sistem sistem yang mendukung untuk kemajuan institusi.

Metode *K-Nearest Neighbor*[1] merupakan metode paling sederhana yang memiliki basis algoritma pembelajaran dalam menentukan klasifikasi, dimana cara kerja metode ini dapat membandingkan suatu objek data baru yang paling mirip (ketetanggaan) dengan data latih yang sudah tersimpan. Metode KNN ini juga memiliki kelebihan sangat efektif digunakan pada data yang sangat besar, sehingga alasan tersebut metode akan digunakan dalam penelitian.

Machine Learning [2] merupakan suatu metode pembelajaran pada mesin supaya mesin dapat meniru cara kerja otak manusia sehingga dapat memahami dan mengklasifikasikan suatu objek tertentu, dimana hal ini objek yang ditangkap dalam bentuk citra. Dalam penelitian ini dirancang sistem absensi berbasis *mobile* android yang mampu melakukan pengambilan citra wajah, selanjutnya wajah yang terdeteksi diverifikasi dan pengenalan (*recognize*) wajah menggunakan *Machine Learning* menggunakan metode *K-Nearest Neighbor* dengan *library* OpenCV Python 3.8, *face recognition*, *dLib* dan *TensorFlow Lite* dalam proses kehadiran siswa di sekolah.

Dalam implementasi *TensorFlow Lite* berbasis *mobile* Android menggunakan *library* *Mobile FaceNet* dalam proses *real time face recognition* pada aplikasi Android, dimana *Mobile FaceNet* ini dapat mempresentasikan model CNN yang sangat efisien yang dirancang khusus untuk verifikasi wajah *real-time* presisi tinggi di perangkat seluler, *Mobile FaceNet* ini merupakan karya peneliti di Watchdata Inc. di Beijing. Dimana *Mobile FaceNet* ini mencapai performa mencapai kecepatan yang sangat baik dengan akurasi yang sangat tinggi dengan model hanya 4,0 MB. Akurasi yang diperoleh sangat mirip dengan model lain yang lebih berat seperti *FaceNet*[3].

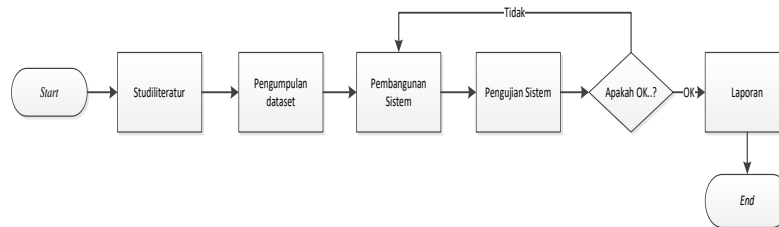
Dalam penelitian ini dirancang sistem absensi berbasis *mobile* android yang mampu melakukan pengambilan citra wajah, selanjutnya wajah yang terdeteksi diverifikasi dan dicocokkan wajah menggunakan *Machine Learning* menggunakan klasifikasi dengan metode *K-Nearest Neighbor*. Dalam implementasi antar muka pada aplikasi dekstop menggunakan *Python* versi 3.8 dengan *library* *opencv-python*, *face recognition*, *dlib*, *pillow*, *imutils*, *numpy*, *flask*, *h5py* dan *sklearn*. Sedangkan pada aplikasi *mobile* menggunakan *Android Studio* versi 3.1, *Java sdk* 1.8, *gradle* versi 6.8.2 dan *TensorFlow Lite (mobile_face_net.tflite)* dalam proses kehadiran siswa di sekolah.

Kami mempelajari beberapa penelitian sebelumnya tentang Sistem kehadiran siswa tingkat lanjut menggunakan Pengenalan wajah berbasis Android, adapun penelitian terkait diantaranya yaitu Penelitian *Advance Attendance System using Face Recognition and Android*[4], Penelitian perancangan Sistem Kehadiran *Online* menggunakan deteksi wajah dan geofencing dengan menggunakan *Deep Learning* Berbasis Android [5], Penelitian terkait kinerja deteksi dan pengenalan wajah menggunakan *Baidu AI* yang menghasilkan, kecepatannya dua kali lebih cepat daripada *Face++* pada kondisi pendeteksian secara langsung dengan nilai akurasi pengenalan lebih tinggi dari *Face++* hal ini dapat mencegah penipuan gambar secara statis[6], Penelitian Sistem kehadiran berbasis AI dengan pengenalan wajah berbasis komputer dengan kecepatan dan akurasi pengenalan wajah dari kamera pengintai[7], Penelitian terkait perancangan absensi berbasis *Internet of Things* (IoT) yang menggunakan Standar IEEE 2413 untuk Kerangka Arsitektur untuk *Internet of Things* (IoT) pada Pengenalan Wajah, Sidik jari dan QR Code[8], Penelitian terkait perancangan sistem kehadiran harian siswa dapat dilacak secara real time dengan pengenalan wajah menggunakan metode pembelajaran mesin untuk menghilangkan kesalahan input secara manual[9].

2. METODE PENELITIAN

2.1. Diagram alir penelitian

Dalam penelitian ini melalui beberapa tahapan yaitu studi Literatur, pembangunan dataset, pembagunan sistem, pengujian sistem jika, berjalan baik maka dilakukan dokumentasi dan pelaporan , jika terjadi kesalahan dilakukan perbaikan sistem kembali sampai selesai.



Gambar 1. Diagram Alur penelitian

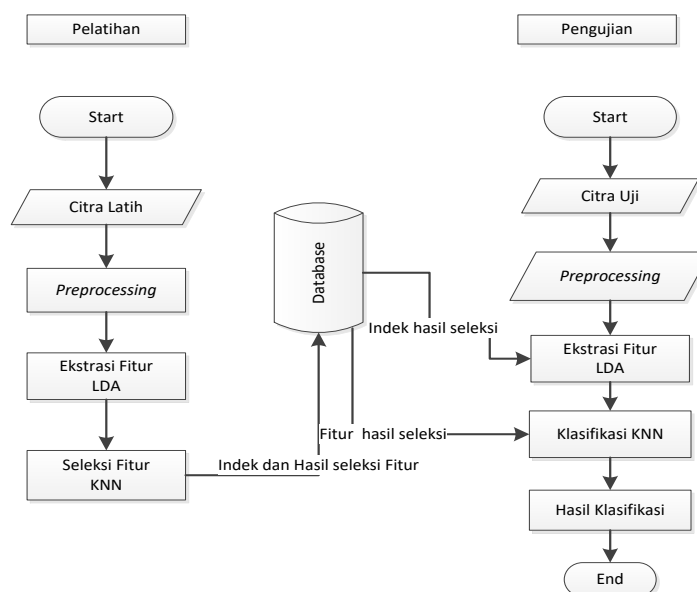
2.2. Pengumpulan dataset

Pada penelitian ini, dataset yang digunakan menggunakan data wajah yang diambil secara langsung oleh aplikasi backend kemudian di crop dengan ukuran 108x108 dan disimpan pada database. Dataset tersebut berjumlah 255 citra wajah yang diambil dari 5 (empat) orang siswa dimana masing-masing siswa diambil otomatis 51 citra wajah dalam berbagai posisi. File di simpan di dalam database dalam format jpg dengan tingkat keabuan 256 piksel.



Gambar 2. Contoh dataset

2.3. Perancangan Sistem



Gambar 3. Perancangan Sistem

Pada Gambar 3 terdapat dua proses pelatihan dan pengujian, proses selanjutnya preprocessing, ekstraksi fitur dan seleksi fitur. Proses selanjutnya hasil seleksi fitur LDA diekstrasi ke fitur KNN di

simpan kedalam database. Pada prose pengujian, mere-load ulang indeks dan fitur dari database di ekstraksi LDA. Proses terakhir melakukan proses klasifikasi menggunakan KNN untuk mendapat akurasi, presisi, recall dan 1 k-fold.

2.4. Preprocessing

Pada proses preprocessing yang dilakukan dengan *splitting*, *resizing*, *men-suffle* dan normalisasi citra. *Splitting* dilakukan untuk membagi data beberapa bagian dengan tujuan untuk sebagai data training dan data testing. *Resizing* untuk mengubah resolusi citra, reduksi citra dilakukan untuk mengubah dimensi citra, normalisasi citra di lakukan karena setiap fitur pada citra memiliki rentang nilai yang sangat signifikan. Semua proses dilakukan dalam sistem.

2.5. Ekstrasi Fitur LDA

Penggunaan metode *Linear Discriminant Analysis* (LDA) merupakan metode dengan sifat diskriminatif yang baik untuk memperkirakan suatu sub ruang secara linear[10]. Metode ini merupakan suatu metode untuk melakukan ekstrasi fitur yang cara kerjanya berdasarkan dari analisa suatu matrik sebaran untuk menemukan proyeksi yang optimal. Sehingga metode LDA ini memaksimalkan penyebaran antar kelas dan meminimalkan penyebaran dalam kelas wajah. Perbedaan antar suatu kelas direpresentasikan oleh scatter between class (S_b) dan untuk perbedaan dalam kelas direpresentasikan oleh scatter within class (S_w).

Adapun tahapan proses ekstrasi fitur dengan menggunakan metode LDA:

- a. Mengubah matrik dua dimensi menjadi matrik satu dimensi atau mengubahnya kedalam bentuk vektor baris dan kolom.
- b. Data training dikelompokkan ke dalam matrik sejumlah kelas (X_i).
- c. Menghitung nilai rata-rata(mean) dari tiap-tiap kelas (μ_i) dengan persamaan (1), jika data dalam bentuk vektor baris maka perhitungan mean dihitung berdasarkan baris, jadi selanjutnya dimensi mean sama dengan satu dimensi satu dengan data training bukan dimensi dataset.

$$\mu_i = \frac{1}{N_i} \sum_{x \in w_i} x \quad (1)$$

- d. Menghitung rata-rata dari semua kelas (μ) dengan persamaan (2)

$$\text{Mean} : \mu_i = \frac{1}{x_i + \dots + x_c} \sum_{x \in w_i} x \quad (2)$$

- e. Menghitung Matrik Sebaran : $S_b = \sum_{i=1}^c N_i (\mu_i - \mu)^T (\mu_i - \mu)$ (3)

$$S_w = \sum_{i=1}^c \sum_{j=1}^{N_i} ((\mu_j - \mu_i)^T (\mu_j - \mu_i)) \quad (4)$$

- f. Menghitung nilai *covariance* matriks (C) dengan persamaan (5)

$$C = S_b * (S_w)^{-1} \quad (5)$$

- g. Menghitung *eigen value* (λ) dan *eigen vector* (v) dengan persamaan (6)

$$C v = \lambda v \quad (6)$$

- h. Memproyeksikan citra asal dengan *eigen vector* terpilih dari nilai *eigen* terbesar ke terkecil sebanyak n-1 *eigen vector*, dimana n jumlah kelas dengan persamaan (7)

$$V = V^T X^i \quad (7)$$

2.6. Klasifikasi KNN

Metode *K-Nearest Neighbor* (KNN)[11][12] merupakan salah satu metode pengambilan keputusan yang digunakan dalam proses klasifikasi terhadap objek berdasarkan dari data pembelajaran yang mempunyai jarak terdekat dengan objek tersebut. Tujuan dari metode KNN ini untuk mengklasifikasi objek baru berdasarkan atribut dan sample dari training data, kemudian disimpan dalam bentuk vektor dari sebuah fitur klasifikasi data. Untuk perhitungan jarak terdekat pada suatu kelas menggunakan metode *euclidean distance*. Adapun langkah-langkah metode KNN :

- a. Menentukan variabel K, dimana K merupakan jumlah tetangga terdekat.

- b. Melakukan perhitungan jarak antara citra yang diujikan dengan citra yang ada di *database* dengan menggunakan rumus persamaan sebagai berikut:

$$euclidean\ distance \quad d(x, y) = \sqrt{\sum_{i=1}^n (X_i - Y_i)^2} \quad (8)$$

- c. Mengurtukan semua objek ke dalam kelompok yang memiliki jarak *euclid* yang terkecil.
- d. Mengumpulkan kategori dari Y yang merupakan klasifikasi dari *nearest neighbor*.
- e. Suatu nilai *query instance* dapat dihitung dalam memprediksikan.

2.7. Teknik Pengujian

Tahap pengujian merupakan tahap untuk menguji apakah sistem sudah berjalan dengan baik atau tidak serta untuk mengetahui kekurangan sistem pada saat terjadi kesalahan pada proses pengujian. Pada proses pengujian digunakan metode *confusion matrix* yang akan menghitung nilai akurasi, recal dan presisi. Perhitungan dengan *confusion matrix* dapat dilakukan dengan persamaan berikut:

$$Akurasi = \frac{Jumlah\ data\ dari\ seluruh\ target}{Total\ keseluruhan\ data} \quad (9)$$

$$Presisi = \frac{Jumlah\ data\ sesuai\ target\ di\ satu\ kelas}{Jumlah\ seluruh\ sesuai\ target} \quad (10)$$

$$Recall = \frac{Jumlah\ data\ sesuai\ target\ di\ satu\ kelas}{Jumlah\ data\ di\ satu\ kelas} \quad (11)$$

2.8. Metode Pengembangan Perangkat Lunak

Supaya proses penelitian berjalan sesuai dengan kaidah pengembangan perangkat lunak[13],[14] dilakukan dengan 5 langkah dimana langkah-langkah tersebut seperti tergambar pada gambar 4, adapun langkah tersebut sebagai berikut :



Gambar 4. Langkah Penelitian

1. Analisis Kebutuhan
Pada tahapan awal penelitian melakukan pencarian literatur sebagai bahan kajian dalam perancangan aplikasi kehadiran siswa di sekolah dengan menggunakan *metode face recognition* berbasis android.
2. Desain dan Perancangan
Pada tahapan ini setelah didapatkan data penelitian dari literatur sebelumnya dijadikan bahan pertimbangan, selanjutnya dilakukan desain dan perancangan dalam pengembangan aplikasi kehadiran menggunakan *face recognition*.
3. Implementasi Sistem
Pada Tahapan ini proses melakukan implementasi dan pengembangan aplikasi sistem dari hasil rancangan aplikasi kehadiran menggunakan *face recognition* menggunakan

metode *Machine Learning*, untuk implementasi pada perangkat desktop menggunakan *library python 3.8* dan untuk perangkat *Mobile android* menggunakan *Android Studio*.

4. Pengujian Sistem

Pada tahapan pengujian sistem ini dimana implementasikan sistem diuji sesuai parameter terkait perkembangan perangkat lunak.

5. Dokumentasi Penelitian

Setelah dilakukan implementasi dan pengujian aplikasi sistem sesuai parameter yang ditentukan maka dilakukan pembuatan dokumentasi baik mulai tahap perancangan sampai selesai pembuatan aplikasi.

2.9. Perancangan Data

Perancangan data pada aplikasi kehadiran *face recognition* ini sebagai berikut :

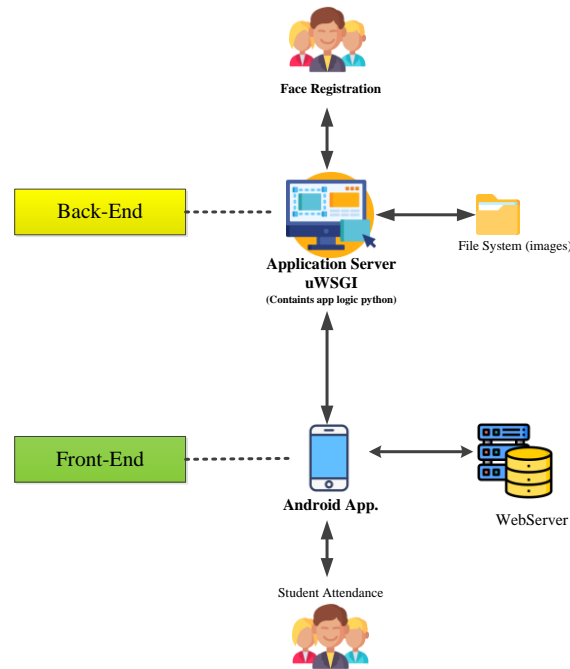
1. Penggunaan data wajah yang dikonversi kedalam bentuk vektor wajah yang disimpan dalam *folder dataset* siswa sesuai dengan nama siswa.
2. Inputan wajah data siswa dengan *cropping* dengan diameter wajah 108x108.
3. File data gambar wajah siswa bertipe file berekstensi *.JPG*.
4. Size dari data wajah tersimpan di *storage* berukuran 5 kb ~ 5.3 kb
5. *Feature* vektor wajah disimpan dalam bentuk file *Json object (.json)* dengan no urut.
6. Jumlah data wajah untuk data training sejumlah 51 data wajah setiap *user*.
7. Pengambilan data *sample* wajah diambil oleh aplikasi *backend* dengan *library python* dengan API Python pada *server environment*.

2.10. Prototype Perancangan Arsitektur Perangkat Lunak

Dalam perancangan sistem kehadiran menggunakan *face recognition* masih dalam bentuk prototype aplikasi, dimana pada sisi *server (Back-End)* menggunakan *library python Server WSGI* dan untuk *client (Front-End)* menggunakan aplikasi android.

2.11. Arsitektur Sistem Kehadiran

Arsitektur dari aplikasi kehadiran siswa menggunakan *face recognition*[15],[16],[17] dimana dari sisi *Back-End*: merupakan aplikasi server *WSGI*[18],[19] yang bertugas sebagai interface dalam menangani penyimpanan dataset wajah user. server menggunakan *server environment (.env)* dengan *library python 3.8, OpenCV Python, Face Recognition, Server Flask, Json* dimana *user* akan melakukan registrasi wajah pada sistem kemudian hasil registrasi akan disimpan pada *database* wajah sebanyak 51 contoh wajah, jumlah wajah tersebut selanjutnya akan di generate dan dilakukan *training* data wajah dengan metode *KNN*[20] (*trained_knn_model.clf*) untuk menentukan nilai kedekatan dari sampel wajah tersebut dimana nilai kedekatan/akurasi nilai tertentu akan menunjukkan akurasi wajah seorang siswa selanjutnya disimpan untuk verifikasi wajah jika ada *request* dari aplikasi android. Dari sisi *Front-End* merupakan *interface* antara siswa dengan *server* dalam melakukan proses kehadiran, jika dalam verifikasi wajah yang terbaca di aplikasi android merupakan data wajah yang sudah didaftarkan dan di training maka siswa tersebut akan di validasi untuk melakukan proses kehadiran, selanjutnya record kehadiran siswa akan dikirim disimpan pada basis data yang mana dalam hal ini menggunakan *database MySQL*. Untuk metode komunikasi antara *Back-End* dan *Front-End* menggunakan *RestFul API web service* sebagai jembatan komunikasi atau pertukaran data antara sistem *client* dengan sistem *server API* yang diolah oleh aplikasi menggunakan tipe *JSON*.



Gambar 5. Arsitektur Sistem Kehadiran *Face Recognition*

3. HASIL DAN PEMBAHASAN

3.1. Pengujian

Pengujian yang dilaksanakan dengan metode split dataset dari 2 bagian sampai 5 bagian, yang bertujuan untuk memproposikan data sampling keseluruhan bagian supaya data lebih merata, selanjutnya dihitung *confusion* matriknya untuk mencari nilai akurasi, presisi, recall dan k-fold *score* F1, adapun hasilnya seperti pada gambar 6 dan 7 :

```

Command Prompt
(A_Rec) E:\A_Rec\Nearst>cclassifier
-----
x_train shape: (75, 4)
y_train shape: (75,)
x_test shape: (75, 4)
y_test shape: (75,)
-----
[[29  0  0]
 [ 0 20  3]
 [ 0  1 22]]
    
```

Gambar 6. Split dataset 5 bagian

```

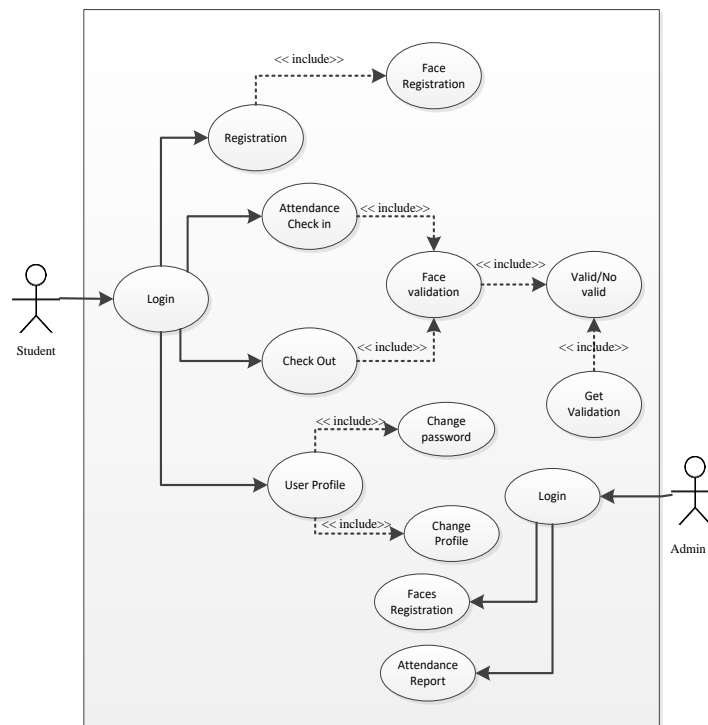
Command Prompt
(A_Rec) E:\A_Rec\Nearst>cclassifier
-----
x_train shape: (75, 4)
y_train shape: (75,)
x_test shape: (75, 4)
y_test shape: (75,)
-----
[[29  0  0]
 [ 0 20  3]
 [ 0  1 22]]
(A_Rec) E:\A_Rec\Nearst>measure
-----
~ Accuracy : 0.9466666666666667
~ Precision : 0.9441269841269841
~ Recall : 0.9420289855072465
~ F1 : 0.9419191919191919
    
```

Gambar 7. Hasil pengujian *confusion* matrik

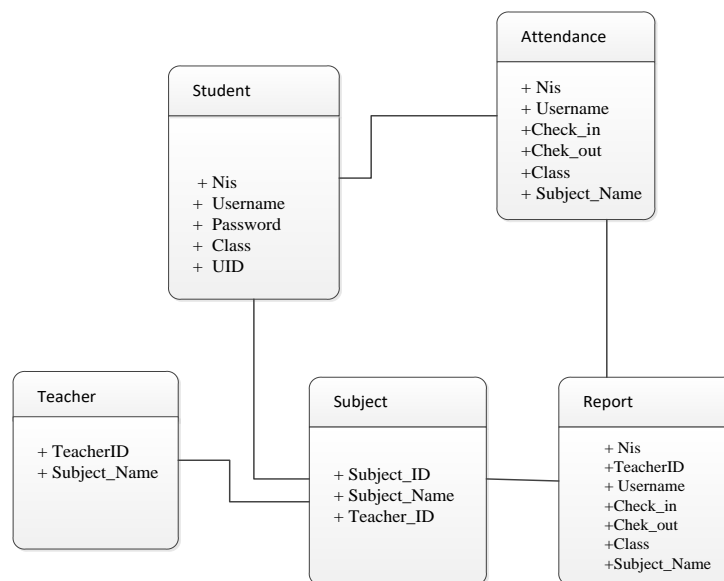
Adapun hasil pengujian *confusion matrix* dengan nilai akurasi 94.66%, presisi 94.41 %, recall 94.20% , k-fold *score* F1 94.19.

3.2. Implementation Kehadiran Siswa Dengan *Face Recognition*

Implementasi pada perancangan sistem kehadiran siswa menggunakan *face recognition* terdiri dari dua aktor yaitu pelajar dan admin, dimana aktor siswa akan melakukan pendaftaran wajah, dimana sistem akan melakukan *cropping* gambar wajah berukuran 108x108 kemudian data wajah akan disimpan dalam folder nama siswa masing-masing, selanjutnya sistem akan *generate* data wajah menggunakan metode KNN data vektor wajah tersebut akan digunakan pada saat proses kehadiran, jika wajah tidak terdeteksi maka siswa tidak akan bisa melakukan kehadiran, untuk aktor admin akan melakukan rekap terkait data kehadiran siswa adapun *use case diagram* seperti pada gambar 8, dan class diagram seperti gambar 9 :



Gambar 8. Use Case Diagram



Gambar 9. Class Diagram

3.3. Implementasi *Training Data Wajah*

Pada posisi tampilan GUI *Back-End* menggunakan *library python* dilakukan *training* data wajah setiap siswa yang melakukan pendaftaran yang disimpan pada dataset masing-masing siswa seperti terlihat pada gambar 10 :

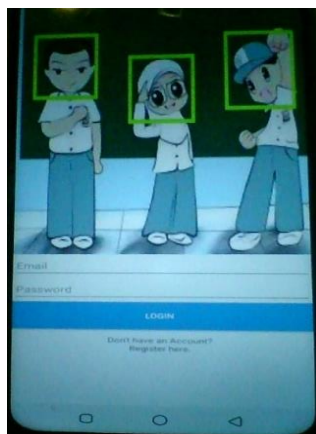
```

C:\>
dataset/Azizah H/2.jpg
<108, 108, 3>
108 108
dataset/Azizah H/20.jpg
<108, 108, 3>
108 108
dataset/Azizah H/21.jpg
<108, 108, 3>
108 108
dataset/Azizah H/22.jpg
<108, 108, 3>
108 108
dataset/Azizah H/23.jpg
<108, 108, 3>
108 108
    
```

Gambar 10. *Training Dataset*

3.4. Implementasi *Front-End Aplikasi*

Dalam pengembangan aplikasi antarmuka untuk posisi *Front-End* menggunakan aplikasi mobile IntelliJ Ide menggunakan Framework Android dibangun dengan menggunakan lingkungan kerja Android Studio versi 3.1, adapun untuk antarmuka yang ditampilkan diantaranya tampilan login siswa (gambar 11), menu utama (gambar 12), kehadiran dengan menggunakan *face recognition* (gambar 13) dan daftar kehadiran siswa (gambar 14) terkait masuk (*check-in*) dan keluar (*check-out*).



Gambar 11. Antarmuka *Login*



Gambar 12. Menu Utama



Gambar. 13. Kehadiran *Face Recognition*



Gambar 14. Daftar Kehadiran Siswa

3.5. Pengujian Aplikasi *Back-End* dan *Front-End*

Terkait pengujian pada sistem aplikasi kehadiran siswa menggunakan *face recognition* seperti Tabel 1 :

Tabel 1. Testing Aplikasi

<i>BACK-END</i>				
No	<i>Modul Item</i>	<i>Feed back</i>	<i>Inference time (sec)</i>	Status
1	Registrasi wajah	Dapat melakukan registrasi	7 ~ 9 detik	<i>accepted</i>
2	Training wajah	Dapat melakukan training wajah	10 ~ 15 sec	<i>accepted</i>
<i>FRONT-END</i>				
No	<i>Modul Item</i>	<i>Feed back</i>	<i>Inference time (sec)</i>	Status
1	Login	User dapat login	Kurang lebih 1 detik	<i>accepted</i>
2	Registrasi	Ussr dapat registrasi	Kurang lebih 1 detik	<i>accepted</i>
3	Kehadiran	User dapat melakukan kehadiran dengan <i>face recognition</i>	Kurang lebih 5 detik	<i>accepted</i>
4	Tampilan Kehadiran	Sistem dapat menampilkan data kehadiran	Kurang lebih 2 detik	<i>accepted</i>
5	Logout	User dapat logout	Kurang lebih 1 detik	<i>accepted</i>

4. KESIMPULAN

Berdasarkan hasil penelitian pada perancangan dan implementasi sistem kehadiran siswa menggunakan *face recognition*, pengujian model dengan metode K-NN menghasilkan nilai pengenalan wajah siswa sesuai dengan *confusion* matrik mendapatkan nilai akurasi 94.66%, presisi 94.41 %, recall 94.20% , k-flod F1 94.19. Pengujian aplikasi secara GUI pada sisi aplikasi BackEnd dapat melakukan registrasi wajah *generate* wajah dengan inferensi waktu 7~9 detik setiap siswa. Untuk GUI sisi *FrontEnd* menggunakan aplikasi *mobile* android untuk melakukan sistem kehadiran siswa dengan inferensi waktu respon perangkat 5 detik untuk mengenali siswa dan mencatat kehadiran pada sistem. Dari implementasi dan trial sistem dapat disimpulkan dari sisi kecepatan dan keakuratan sudah memadai untuk dapat digunakan pada satu institusi kelas atau sekolah. Kestabilan *server face_api* juga cukup memadai selama testing dapat berjalan tanpa gangguan lebih dari 7x24 jam. Sementara penggunaan untuk institusi yang melibatkan ribuan atau jutaan identitas yang

berbeda seperti kependudukan atau pemerintahan mungkin harus melalui testing dan verifikasi yang lebih matang jika menggunakan solusi seperti yang diuraikan.

UCAPAN TERIMA KASIH

Kami atas nama tim mengucapkan terima kasih terkait referensi kode program dalam penelitian yang terkait perancangan untuk membangun aplikasi sistem kehadiran dengan menggunakan pengenalan wajah

REFERENSI

- [1] M. S. Sarma, Y. Srinivas, M. Abhiram, and L. Ullala, "Insider Threat Detection with Face Recognition And KNN User Classification," *2017 IEEE Int. Conf. Cloud Comput. Emerg. Mark. Insid.*, 2017.
- [2] Y. Xin *et al.*, "Machine Learning and Deep Learning Methods for Cybersecurity," *IEEE Access*, vol. 6, pp. 35365–35381, 2018.
- [3] J. Xiao, G. Jiang, and H. Liu, "A Lightweight Face Recognition Model based on MobileFaceNet for Limited Computation Environment," *EAI Endorsed Trans.*, vol. 7, no. 27, pp. 1–9, 2022.
- [4] H. K. Singh, V. Jain, A. Kumar, and V. B. P, "Advance Attendance System using Face Recognition and Android," no. May, pp. 4217–4220, 2021.
- [5] A. P. Putera, P. N. Primandari, S. St, and M. Im, "Rancang Bangun Aplikasi Absensi Online Berbasis Android Menggunakan Metode Deep Learning Pada PT . Pelabuhan Indonesia III (Persero)," 1945.
- [6] P. Chena, X. Gengb, M. Zouc, Q. Xud, and D. Tan, "Development and Optimization of Check-in System Based on Face Recognition Technology," *IOP Conf. Ser. Mater. Sci. Eng.*, 2020.
- [7] K. Kalel, A. Shinde, and P. A. S. Shinde, "AI Based Attendance System Using Faces Recognition," *Int. J. Res. Publ. Rev.*, vol. 3, no. 6, pp. 3299–3304, 2022.
- [8] N. A. El-mawla and M. Ismaiel, "Smart Attendance System Using QR-Code , Finger Print and Face Recognition," *Nile J. Commun. Comput. Sci.*, vol. 2, no. 1, pp. 1–16, 2022.
- [9] G. S. Anisha and G. V. S. S. Reddy, "AUTOMATED FACIAL RECOGNITION FOR CLASS ATTENDANCE," *Int. J. Recent Dev. Sci. Technol.*, vol. 06, no. 06, pp. 168–173, 2022.
- [10] M. H. Ramdani, I. G. P. S. Wijaya, and R. Dwiyanaputra, "OPTIMALISASI PENGENALAN WAJAH BERBASIS LINEAR DISCRIMINANT ANALYSIS DAN K-NEAREST NEIGHBOR MENGGUNAKAN PARTICLE SWARM OPTIMIZATION," *J. Teknol. Informasi, Komput. dan Apl.*, vol. 4, no. 1, pp. 40–51, 2022.
- [11] X. Fu, J. Lu, and X. Zhang, "Intelligent In-vehicle Safety and Security Monitoring System with Face Recognition," *2019 IEEE Int. Conf. Comput. Sci. Eng. IEEE Int. Conf. Embed. Ubiquitous Comput.*, pp. 225–229, 2019.
- [12] M. A. Abuznied and A. Mahmood, "Enhanced Human Face Recognition Using LBPH Descriptor , Multi-KNN , and Back- Propagation Neural Network," *IEEE Access*, vol. 3536, no. c, pp. 1–11, 2018.
- [13] F. Syakti, "METODE PENGEMBANGAN PERANGKAT LUNAK BERBASIS MOBILE : A REVIEW," *Bina Komput.*, vol. 1, no. 2, pp. 82–89, 2019.
- [14] M. Z. Zahid, "Aplikasi Berbasis Android untuk Pembelajaran : Potensi dan Metode Pengembangan," *PRISMA*, vol. 1, pp. 910–918, 2018.
- [15] L. Rai and Z. Wang, "Software Development Framework for Real-Time Face Detection and Recognition in Mobile Devices," *Int. J. Interact. Mob. Technol.*, vol. 14, no. 4, pp. 103–120, 2020.
- [16] F. Cao and L. Zhu, "Research on Campus Attendance System Based on Face Recognition and Trajectory Tracking," *IOP Conf. Ser. Earth Environ. Sci.*, pp. 0–7, 2021.
- [17] A. Rambe, J. P. Tanjung, and Muhathir, "Shafiyatul Amaliyyah School Student Face Absence Using Principal Component Analysis and K-Nearest Neighbor," *JITE (Journal Informatics Telecommun. Eng.*, vol. 5, no. January, 2022.
- [18] G. Manduchi, T. Fredian, and J. Stillerman, "A new web-based tool for data visualization in MDSplus," *Fusion Eng. Des.*, pp. 2–5, 2014.
- [19] R. Rahutomo, A. S. Perbangsa, Y. Lie, T. W. Cenggoro, and B. Pardamean, "Artificial Intelligence

- Model Implementation in Web-Based Application for Pineapple Object Counting,” *2019 Int. Conf. Inf. Manag. Technol.*, vol. 1, no. August, pp. 525–530, 2019.
- [20] D. R. Yulianti, I. I. Triastomoro, and S. Sa’idah, “Identifikasi pengenalan wajah dengan menggunakan metode knn (k-nearest neighbor) dan lbph (local binary pattern histogram) untuk sistem presensi,” *TEKINKOM*, vol. 5, pp. 1–10, 2022.