

APLIKASI PRESENSI MENGGUNAKAN PENGENAL WAJAH BERBASIS OPENCV

PRESENCE APPLICATION USING OPENCV-BASED FACE IDENTIFICATIONS

Neng Yessi Novamonika¹, Sugondo Hadiyoso², Tengku A. Riza³

^{1,2,3}Program Studi Teknologi Telekomunikasi, Fakultas Ilmu Terapan, Universitas Telkom

¹nengy14@gmail.com, ²sugondo@tass.telkomuniversity.ac.id, ³tengkuriza@tass.telkomuniversity.ac.id

Abstrak

Penelitian ini merealisasikan sebuah aplikasi pencatat kehadiran otomatis menggunakan mekanisme pengenalan wajah. Metode pengenalan yang dipakai adalah metode Eigenface dengan algoritma Haar Cascade yang mampu mendeteksi dengan cepat dan *real-time* wajah manusia. Parameter pengujian yang dilakukan meliputi tiga kondisi yakni pengujian terhadap tipe wajah, terhadap jarak ke objek, dan terhadap delay waktu pembacaan objek. Seluruh kondisi tersebut mengacu pada tingkat keakuratan yang dinyatakan dalam TAR (True Acceptance Rate). Hasil pengujian menunjukkan nilai TAR pada tipe wajah berkacamata yakni 100%, sedangkan yang tidak berkacamata memiliki nilai TAR 93,5%. Pada pengujian kedua terhadap jarak antara *smartphone* dan wajah, menunjukkan nilai TAR pada jarak 20 cm yakni 57%, sedangkan pada jarak 30 cm memiliki nilai TAR 56%. Pada pengujian terhadap delay pembacaan wajah, didapatkan hasil rata-rata delay sebesar 3,93 detik.

Kata kunci : kehadiran, wajah, Eigenface, smartphone

Abstract

In this study, an automatic attendance recording application was realized using the face recognition method. The recognition method used is the Eigenface method with the Haar Cascade algorithm which is able to detect quickly and real-time human faces. The parameters of the test include three conditions, namely the test of the face type, the distance to the object, and the time delay of reading the object. All of these conditions refer to the level of accuracy stated in the TAR (True Acceptance Rate). The test results show that the TAR value on the face type with glasses is 100%, while those without glasses have a TAR value of 93.5%. In the second test of the distance between the smartphone and the face, the TAR value showed at a distance of 20 cm is 57%, while at a distance of 30 cm had a TAR value of 56%. In testing the face reading delay, the average delay is 3.93 seconds.

Keywords: attendance, face, Eigenface, smartphone

1. PENDAHULUAN

Face recognition atau pengenalan wajah adalah teknologi komputer yang memungkinkan kita untuk mengidentifikasi atau memverifikasi wajah seseorang melalui sebuah gambar digital [1], [2]. Pengenalan sistem wajah ini sudah banyak digunakan pada aplikasi biometrik selain retina mata, sidik jari, dan iris mata [3], [4].

Seiring perkembangan teknologi yang telah ada, hasil penelitian berkaitan dengan aplikasi pencatat kehadiran otomatis yang telah direalisasikan dan dilaporkan pada beberapa studi. Penelitian aplikasi kehadiran berbasis RFID telah diusulkan pada studi [5]–[7]. Namun demikian, kartu RFID masih memiliki kekurangan antara lain pengaruh logam yang dapat interferensi gelombang radio dan tag mudah rusak sehingga tidak terdeteksi [8]. Kemudian penelitian lain mengusulkan sistem presensi menggunakan sidik jari (*fingerprnt*) yang terintegrasi dengan *web service* [9], [10]. Namun demikian sidik jari dapat dengan mudah dipalsukan [11]. Selain itu sidik jari masih memiliki kekurangan yaitu

kondisi jari harus dalam keadaan bersih, jika tidak biasanya sistem sulit untuk mencocokkan atau membaca data sidik jari. Penelitian aplikasi sistem presensi lainnya diusulkan oleh A. R. Safitri and L. A. Sholikhah [12], membahas sistem presensi dosen menggunakan QR-Code scanner berbasis android. Namun demikian, QR-Code juga cenderung mudah untuk di-copy atau dipalsukan untuk kepentingan tertentu.

Aplikasi presensi yang dijelaskan diatas mempunyai kekurangan masing-masing, untuk mengatasi masalah tersebut maka pada penelitian ini mengembangkan sebuah sistem presensi menggunakan identifikasi pengenalan wajah. Aplikasi presensi menggunakan biometrik wajah telah dikembangkan pada beberapa studi [13]–[15]. Namun aplikasi tersebut berjalan menggunakan personal komputer sehingga memerlukan biaya tinggi untuk implementasi. Oleh karena itu pada penelitian ini diusulkan sebuah sistem presensi berbasis Android mobile yang dikembangkan melalui aplikasi *open source*. Dengan adanya aplikasi ini diharapkan dapat mempermudah sistem presensi agar lebih efisien untuk diterapkan.

2. PERANCANGAN DAN REALISASI SISTEM

2.1 Kebutuhan Perangkat

Perancangan sistem ini membutuhkan beberapa dukungan dari perangkat keras dan perangkat lunak agar sistem dapat berjalan dengan baik. Adapun perangkat keras dan perangkat lunak yang dibutuhkan selama perancangan, yaitu:

1. Laptop
2. Handphone Oppo F5 dengan versi Android minimum Android 7.1.1 ‘Nougat 7.0’
3. Android Studio
4. Java

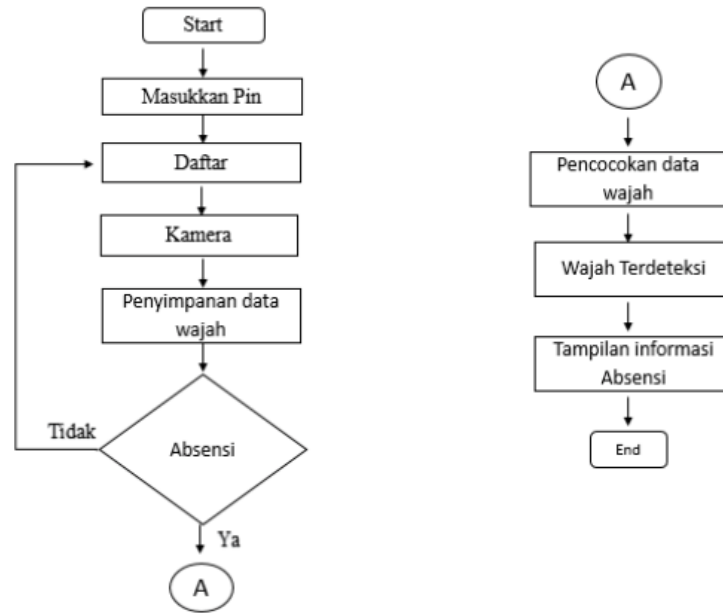
2.2 Alur Kerja Sistem

Ringkasan alur sistem presensi menggunakan pengenalan wajah berbasis opencv dapat dilihat pada Gambar 1 berikut:



Gambar 1. Skema presensi berbasis pengenalan wajah

Detail alur kerja sistem ini disajikan pada Gambar 2 berikut.



Gambar 2. Alur kerja sistem

2.3 Realisasi rancangan desain dan fitur menjadi Aplikasi

Rancangan desain dan fitur yang telah dibuat, diimplementasikan dengan menuliskan syntax-syntax agar menjadi sebuah aplikasi. Beberapa syntax yang ada pada beberapa fitur yg dapat direalisasikan menjadi aplikasi antara lain:

- a) Membuka Aplikasi Cek versi android, jika versi lebih tinggi dari Lollipop akan meminta permission untuk mengakses storage melalui method “requestForSpecificPermission()”, jika permission sudah ada maka dapat langsung menjalankan method “loadOpenCV()” untuk menginisiasi modul openCV. Setelah inisiasi modul opencv, maka selanjutnya menjalankan fungsi “postListener”, pada fungsi ini terdapat mehtod “getFile()” dari module firebase storage untuk mendownload file gambar dari storage firebase dan disimpan pada local storage di android, pada saat fungsi dijalankan, akan muncul dialog tunggu pada aplikasi. Sebelum disimpan, gambar yang didapat dari storage firebase dirubah menjadi warna putih abu-abu dan memotong foto pada area wajah.
- b) Menu daftar wajah, pengguna akan diarahkan ke menu pin, pengguna harus memasukkan pin kemudian menekan tombol dan akan menjalankan fungsi seperti pada gambar dibawah. jika pin benar (111 atau 222) maka akan diarah kan ke laman daftar wajah. Pada laman daftar wajah, ketika laman terbuka, aplikasi akan memuat daftar pengguna yang sudah terdaftar menggunakan fungsi dibawah. Fungsi ini dijalankan menggunakan modul firebase untuk mengambil data pada realtime database menggunakan method “addValueEventListener”, method ini akan terpanggil sekali saat inisiasi fungsi dan akan terpanggil kembali ketika ada perubahan data pada firebase realtime database.

```
btnMasuk.setOnClickListener({view} -- {
    if (Integer.parseInt(mPin.getText().toString()) == 111 || Integer.parseInt(mPin.getText().toString()) == 222 ){
        finish();
    }
    else {
        Toast.makeText( context: LoginActivity.this, text: "Pin salah", Toast.LENGTH_SHORT).show();
    }
});
}

switch (requestCode){
    case 101:
        //progressDialog.setTitle("Sedang memuat data dari database");
        //progressDialog.setCancelable(false);
        //progressDialog.show();
        //mDatabase.addListenerForSingleValueEvent(postListener);
        break;
    case 102:
        Intent intent = new Intent(FaceRecognitionAppActivity.this, DaftarUserActivity.class);
        startActivityForResult(intent, requestCode: 101);
        break;
}
}
```

- c) Register wajah, menu ini membutuhkan permission untuk mengaktifkan kamera, jika permission nya sudah disetujui oleh pengguna, kamera sudah dapat digunakan dan akan menginisiasi modul opencv dengan memanggil method "InitOpenCV()". Saat kamera aktif, aplikasi akan menangkap warna dari obyek yang terekam dan menyimpannya dalam variabel. Saat kamera tidak digunakan atau saat keluar dari laman maka variabel akan dibuang sehingga memori bisa diakses kembali.

```
@Override
protected void onResume() {
    super.onResume();
    int MyVersion = Build.VERSION.SDK_INT;
    if (MyVersion > Build.VERSION_CODES.LOLLIPOP_MR1) {
        if (!checkIfAlreadyhavePermission()) {
            requestForSpecificPermission();
        } else {
            InitOpenCV();
        }
    } else {
        InitOpenCV();
    }
}

private void InitOpenCV(){
    if(OpenCVLoader.initDebug()){
        mBaseLoader.onManagerConnected(LoaderCallbackInterface.SUCCESS);
    } else {
        Toast.makeText( context: this, text: "Failed to load OpenCV", Toast.LENGTH_LONG).show();
        //OpenCVLoader.initAsync(OpenCVLoader.OPENCV_VERSION_3_0_0, this, mBaseLoader);
    }
}
```

- d) Proses pencocokan wajah dengan data awal dilakukan dengan metode eigenface, dengan memanggil fungsi "Nativemethods.trainFacesTask" dari modul library opencv. Jika wajah cocok maka akan muncul dialog yang berisi informasi pengguna berupa nama dan tingkat

kemiripan, jika informasi tersebut sesuai, pengguna dapat menekan tombol “benar” yang kemudian akan disimpan pada database riwayat.

```
if (images.isEmpty())
    return true; // The array might be empty if the method is changed in the OnClickListener

if (mTrainFacesTask != null && mTrainFacesTask.getStatus() != AsyncTask.Status.FINISHED) {
    Log.i(TAG, "mTrainFacesTask is still running");
    return false;
}

Mat imagesMatrix = new Mat((int) images.get(0).total(), images.size(), images.get(0).type());
for (int i = 0; i < images.size(); i++)
    images.get(i).copyTo(imagesMatrix.col(i)); // Create matrix where each image is represented as a column

Log.i(TAG, "Images height: " + imagesMatrix.height() + " Width: " + imagesMatrix.width() + " total: ");

// Train the face recognition algorithms in an asynchronous task, so we do not skip any frames

Log.i(TAG, "Training Eigenfaces");
showToast("Training Eigenfaces");

mTrainFacesTask = new NativeMethods.TrainFacesTask(imagesMatrix, trainFacesTaskCallback);

mTrainFacesTask.execute();

return true;
}
```

3. PEMBAHASAN

3.1 Kondisi Pengujian

Adapun ketentuan kondisi pengujian dalam analisis adalah sebagai berikut:

1. Sampel wajah yang digunakan adalah wajah yang menghadap kamera.
2. Jarak antara wajah dan kamera 20-30cm.
3. Tipe wajah yang di gunakan adalah berkacamata dan tidak berkacamata.

3.2. Parameter Uji

Pengujian dilakukan dengan tiga kondisi, untuk mencari akurasi pengenalan wajah adalah dengan membagi test/sampel yang berhasil dengan jumlah sampel percobaan. Berikut adalah persamaan yang digunakan untuk menghitung akurasi:

$$Akurasi = \frac{Jumlah\ wajah\ dikenali}{Jumlah\ percobaan} \times 100\%$$

Nilai akurasi direpresentasikan dalam True acceptance rate (TAR) adalah jumlah deteksi benar antara gambar masukan dengan identitas yang sesuai. Berikut adalah persamaan yang digunakan untuk menghitung TAR:

$$TAR = \frac{Jumlah\ TAR}{Jumlah\ percobaan} \times 100\%$$

False acceptance rate (FAR) adalah kesalahan dalam identifikasi gambar masukan. FAR juga dapat diartikan ketidaksesuaian antara wajah dengan identitas yang diberikan. Wajah A namun dikenali sebagai wajah B. Berikut adalah berikut yang digunakan untuk menghitung FAR:

$$FAR = \frac{\text{Jumlah FAR}}{\text{Jumlah percobaan}} \times 100\%$$

3.3 Pengujian Tipe Wajah

Pengujian ini dilakukan dengan membandingkan tipe wajah yang berkacamata dan tidak berkacamata, sampel yang diuji terdiri dari 30 sampel wajah yang masing-masing pengujian dilakukan sebanyak 3 kali. Dengan memperhatikan hasil presentase TAR dan FAR. Hasil pengujian skenario ini disajikan pada Tabel 1.

Tabel 1. Akurasi sistem terhadap tipe wajah

Tipe Wajah	Jumlah		%	
	TAR	FAR	TAR	FAR
Tidak Berkacamata (31)	29	2	93,5	6,5
Berkacamata (15)	15	0	100	0

Dari hasil pengujian yang ditunjukkan pada Tabel 1 diketahui bahwa akurasi pada tipe wajah tanpa kacamata dan dengan kacamata masing-masing 93,5% dan 100%. Pada wajah tanpa kacamata dijumpai nilai FAR atau kesalahan deteksi sebesar 6,5%. Oleh karena aplikasi yang dikembangkan belum mendukung untuk mekanisme penolakan, maka semua masukan wajah akan dikenali atau diberikan identitas sesuai dengan vector fitur yang paling dekat. Wajah dengan kacamata mempunyai akurasi hingga 100%, ini dapat dikarenakan oleh perbedaan kacamata dapat memberikan fitur spesifik sehingga memberikan perbaikan akurasi deteksi.

3.4 Pengujian Terhadap Jarak

Pada pengujian ini dilakukan untuk mengetahui jarak terbaik agar aplikasi memberikan akurasi tertinggi. Skenario uji adalah memberikan jarak 20 cm sampai 30 cm antara smartphone dan wajah. Pengujian dilakukan sebanyak 90 kali. Hasil pengujian ini disajikan pada Tabel 2.

Tabel 2. Akurasi sistem terhadap jarak

Jarak (cm)	Jumlah		%	
	TAR	FAR	TAR	FAR
20	52	38	57	43
30	51	39	56	44

Dari hasil pengujian yang ditunjukkan pada Tabel 2, nilai TAR dan FAR tidak berbeda signifikan. Sehingga dapat disimpulkan bahwa sistem ini dapat bekerja dengan baik pada range jarak antara 20-30 cm.

3.5 Pengujian Delay

Pengujian ini dimaksudkan untuk mengetahui delay sistem dalam memproses gambar wajah hingga otentikasi selesai dilakukan. Pengujian dilakukan sebanyak 15 kali dimana waktu dicatat secara otomatis oleh aplikasi dalam rentang waktu mengambil gambar wajah hingga identifikasi berakhir. Hasil pengujian ini disajikan pada Tabel 3. Rata-rata delay proses sistem ini adalah 3,93 detik.

Tabel 3. Delay proses deteksi

Pengujian ke-	Delay (detik)	Rata-rata
1	3,87	3,93 detik
2	3,46	
3	3,36	
4	3,33	
5	3,60	
6	3,47	
7	3,64	
8	3,89	
9	3,33	
10	4,29	
11	5,51	
12	5,59	
13	3,33	
14	4,33	
15	4,05	

4. KESIMPULAN

Pada penelitian ini telah direalisasikan sebuah sistem presensi otomatis melalui pengenalan wajah menggunakan *smartphone*. Aplikasi yang dikembangkan mampu melakukan deteksi dan pengenalan *image* wajah secara *real-time*. Aplikasi bekerja pada *smartphone* Android dengan spesifikasi sistem operasi minimum Android 7.1.1. Akurasi tertinggi sebesar 100%, diperoleh pada kondisi subjek memakai kacamata. Sistem ini mampu bekerja secara *real-time* dengan delay rata-rata yang diberikan sebesar 3,93 detik. Penelitian ini masih terbuka peluang untuk dikembangkan dimasa mendatang khususnya usaha untuk meningkatkan akurasi deteksi. Metode atau *library* lainnya dalam mengenali wajah dapat diaplikasikan dalam tujuan meningkatkan akurasi deteksi. Pengujian terhadap subjek yang belum terdaftar juga perlu dilakukan untuk mengetahui performa sistem dalam menolak subjek yang tidak dikenali.

DAFTAR PUSTAKA

- [1] R. V. Virgil Petrescu, "Face Recognition as a Biometric Application," *J. Mechatronics Robot.*, vol. 3, no. 1, pp. 237–257, 2019.
- [2] S. Arya, N. Pratap, and K. Bhatia, "Future of Face Recognition: A Review," *Procedia Comput. Sci.*, vol. 58, pp. 578–585, 2015.
- [3] T. O. Majekodunmi and F. E. Idachaba, "A review of the fingerprint, speaker recognition, face recognition and iris recognition based biometric identification technologies," *Proc. World Congr. Eng. 2011, WCE 2011*, vol. 2, no. September 2018, pp. 1681–1687, 2011.
- [4] B. O. Omoyiola, "Overview of Biometric and Facial Recognition Techniques," *IOSR J. Comput. Eng.*, vol. 20, no. 4, pp. 1–5, 2018.
- [5] S. Nasution, "Presensi Online Menggunakan RFID pada Kartu Mahasiswa," *INTECOMS J. Inf. Technol. Comput. Sci.*, vol. 1, no. 1, pp. 19–27, 2018.
- [6] N. Natalianto, P. Studi, T. Elektro, F. Sains, and U. S. Dharma, "Sistem Presensi Perkuliahan Berbasis Radio Frequency Identification," *Media Tek. J. Teknol.*, vol. 12, no. 2, pp. 91–103, 2017.
- [7] K. P. Aji, U. Darusalam, and N. D. Nathasia, "Perancangan Sistem Presensi Untuk Pegawai Dengan RFID Berbasis IoT Menggunakan NodeMCU ESP8266," *JOINTECS (Journal Inf. Technol. Comput. Sci.)*, vol. 5, no. 1, p. 25, 2020.
- [8] T. P. Dinh, "Modelling, Computation and Optimization in Information Systems and Management Sciences," in *Modelling, Computation and Optimization in Information Systems and Management Sciences*, 2015, vol. 360, pp. 177–188.
- [9] V. O. Wihana and F. Amrullah, "Penerapan Sistem Informasi Presensi Mahasiswa Menggunakan Fingerprint Berbasis Web," *J. Comput. Sci. Appl. Informatics*, vol. 1, no. 2, pp. 7–10, 2019.
- [10] D. Setiawan Putra and A. Fauzijah, "Perancangan Aplikasi Presensi Dosen Realtime Dengan Metode Rapid Application Development (RAD) Menggunakan Fingerprint Berbasis Web," *J. Inform. J. Pengemb. IT*, vol. 3, no. 2, pp. 167–171, 2018.
- [11] Y. Jiang and X. Liu, "Uniform Local Binary Pattern for Fingerprint Liveness Detection in the Gaussian Pyramid," *J. Electr. Comput. Eng.*, vol. 2018, 2018.
- [12] N. Norhikmah, A. R. Safitri, and L. A. Sholikhan, "Penggunaan QR Code Dalam Presensi Berbasis Android," in *Seminar Nasional Teknologi Informasi dan Multimedia*, 2016, pp. 97–102.
- [13] K. K. A. Viola-jones, D. Kehadiran, K. Vision, and P. Wajah, "Sistem Monitoring Kehadiran Perkuliahan Berbasis," *Maj. Ilm. Teknol. Elektro*, vol. 18, no. 1, pp. 119–126, 2019.
- [14] R. Hartanto and M. N. Adji, "Face recognition for attendance system detection," in *Proceedings of 2018 10th International Conference on Information Technology and Electrical Engineering: Smart Technology for Better Society, ICITEE 2018*, 2018, pp. 376–381.
- [15] Zulfachmi, A. Saputra, and K. O. Saputra, "Lecture Attendance Monitoring System Using Face Detection with Microcontroller," *Int. J. Eng. Emerg. Technol.*, vol. 2, no. 2, pp. 31–36, 2017.