



Penerapan Face Recognition Pada Sistem Presensi

Andri Nugraha Ramdhon¹, Fadly Febriya²

^{1,2}Teknik Informatika, Sekolah Tinggi Teknologi Bandung

¹andri.nugraha.r@gmail.com, ²fadlyfebriya@gmail.com

Abstract

The development of digital image technology is increasing nowadays. However, the use of image technology on surveillance cameras has not been optimally utilized. On the other hand, the various presence data monitoring systems that currently exist have their respective advantages and disadvantages, and need to be continuously developed so as to facilitate the data processing. The student attendance system at STT Bandung is basically good but it is still not optimal. The process of collecting student attendance data is still quite time-consuming and still allows human errors to occur in the data input process. Therefore, the author intends to help overcome this by utilizing face recognition technology in an integrated presence process. LBPH (Local Binary Pattern Histogram) is currently the best method in face recognition technology because the detection and recognition process is relatively fast and has the highest level of accuracy when compared to other methods. After testing the resilience of the system from the results of the prototyping that was built, the results obtained with a success rate of 86.85%.

Keywords: Attendance System, Face Detection, Face Recognition, IP CCTV, Python.

Abstrak

Perkembangan teknologi citra digital kian meningkat saat ini. Namun, pemanfaatan teknologi citra pada kamera pengawas tersebut masih belum dimanfaatkan secara optimal. Pada sisi lain berbagai sistem monitoring data presensi yang telah ada saat ini memiliki kekurangan dan kelebihan masing-masing, dan perlu untuk terus dikembangkan sehingga memudahkan dalam proses pengolahan datanya. Sistem presensi mahasiswa di STT Bandung pada dasarnya sudah bagus namun masih dirasa kurang optimal. Proses pendataan presensi mahasiswa masih cukup memakan waktu dan masih memungkinkan terjadi *human error* dalam proses input datanya. Oleh karena itu penulis bermaksud untuk membantu mengatasi hal tersebut dengan memanfaatkan teknologi *face recognition* dalam proses presensi yang terintegrasi. *LBPH (Local Binary Pattern Histogram)* merupakan metode terbaik saat ini dalam teknologi *face recognition* dikarenakan proses pendeteksian dan pengenalan yang relatif cepat dan tingkat akurasi yang tertinggi jika dibandingkan dengan metode yang lain. Setelah dilakukan pengujian ketahanan sistem dari hasil *prototyping* yang dibangun didapatkan hasil dengan tingkat keberhasilan sebesar 86.85%.

Kata kunci: Sistem Presensi, Pendeteksi Wajah, Pengenalan Wajah, Kamera Pengawas, Python.

1. Pendahuluan

Perkembangan teknologi citra digital kian meningkat saat ini. Namun pemanfaatan teknologi citra pada kamera pengawas tersebut masih belum dimanfaatkan secara optimal [1]. Dari segi perangkat lunak, saat ini penggunaan teknologi citra digital *face recognition* sedang marak bermunculan. Seperti penggunaan *face recognition* untuk membuka *smartphone*.

Pada sisi lain berbagai sistem monitoring data presensi yang telah ada saat ini memiliki kekurangan dan kelebihan masing-masing, dan perlu untuk terus dikembangkan sehingga memudahkan dalam proses pengolahan datanya [2]. Mekanisme pertama masih berupa kertas fisik yang rentan rusak, membutuhkan waktu yang relatif lama dan sulit dalam mengkalkulasi datanya. Mekanisme kedua berkembang menjadi sistem yang tersimpan dalam *database*. Dalam hal kalkulasi

data memang jauh lebih cepat tapi proses input datanya harus teliti dan masih memakan waktu karena diinput satu-persatu. Mekanisme terakhir yaitu menggunakan sidik jari. Sistem ini juga masih mempunyai suatu kelemahan, yaitu pemindaian sidik jari orang hadir harus dilakukan secara bergiliran, tidak bisa dilakukan secara bersamaan sekaligus, sehingga seringkali menimbulkan antrian yang panjang. Oleh karena itu sistem ini dianggap kurang efisien [2]. Selain itu *output* data sidik jari juga biasanya terpisah dengan *database* sehingga dibutuhkan sinkronisasi data terlebih dahulu untuk mendapatkan data yang *real time*. Maka dari itu diperlukannya pengembangan sistem presensi yang lebih baik lagi.

Di sisi lain sistem presensi mahasiswa di Sekolah Tinggi Teknologi (STT) Bandung sudah berbasis elektronik, data tersimpan di *database* pada sistem berbasis *website*. Dosen melakukan pengecekan kehadiran dengan cara



Lisensi

Lisensi Internasional Creative Commons Attribution-ShareAlike 4.0.

memanggil nama mahasiswa satu persatu lalu memberikan status presensi di sistem. Hal ini sudah cukup baik namun tidak menutup kemungkinan sering kali muncul beberapa permasalahan. Proses pendataan presensi mahasiswa cukup memakan waktu karena mahasiswa dipanggil satu persatu dan dibutuhkan ketelitian dalam mengisi status presensinya sering kali ada yang terlewat atau bahkan terjadi *human-error* pada pengisiannya. Berdasarkan hasil observasi mandiri dari 5 kelas dengan rata-rata 30 mahasiswa di setiap kelasnya, bisa terjadi 5-7 komplain terhadap data presensi tersebut dalam kurun waktu 1 semester. Selain itu ada keterbatasan manusia yang dimiliki dosen untuk mengingat keseluruhan nama dan wajah mahasiswa khususnya pada lingkup kelas yang baru diampu.

Oleh karena itu berdasarkan pemaparan di atas maka peneliti bermaksud membangun sebuah sistem yang memanfaatkan penerapan teknologi *face recognition* pada proses presensi mahasiswa di STT Bandung guna mengefektifkan waktu serta mengurangi *human error*. Pada sisi *frontend*, input data gambar atau video bisa diambil dari kamera pengawas yang terintegrasi dengan *database* di sisi server. Sisi *backend* bisa memonitoring dan mengelola data presensi mahasiswa dari hasil olah citra. Metode yang digunakan untuk *face recognition* yaitu *LBPH (Local Binary Pattern Histogram)*.

Sistem presensi dengan *face recognition* memang bukan hal yang baru, namun dari hasil studi literatur yang penulis lakukan, kebanyakan masih bersifat *stand alone* dan masih menggunakan kamera lokal (*webcam*). Keterbaruan pada penelitian ini adalah sistem presensi *face recognition* yang terintegrasi secara *real-time* yang menggunakan kamera eksternal (kamera pengawas / *IP CCTV*) sebagai input data dengan penggunaan metode *LBPH* dengan proses yang relatif cepat dan akurasi yang tinggi.

2. Metode Penelitian

Bagian ini merupakan pemaparan mengenai tahapan-tahapan metode penelitian yang digunakan penulis di penelitian ini.



Gambar 1. Metode Penelitian

2.1. Studi Literatur

Proses studi literatur dimulai dengan mencari referensi data pustaka yang berkaitan dengan *face recognition*, teknologi yang digunakan, perkembangannya serta pemanfaatannya. Kegiatan ini berisi membaca, mencatat

dan mengolah data penelitian dari beberapa sumber referensi terkait.

Face detection merupakan teknologi komputer yang digunakan beberapa sistem dan atau aplikasi untuk mendeteksi adanya wajah [3]. Keberhasilan dari *face detection* memiliki tingkat pengaruh yang tinggi dalam performa dan kegunaan dari suatu sistem pengenalan wajah [4].

Face recognition merupakan pengembangan dari teknologi *face detection* dimana teknologi ini dapat menghasilkan/generate wajah dari hasil tangkapan kamera dan melakukan deteksi persamaan wajah dengan data wajah yang diketahui komputer, sehingga komputer dapat mengenali dan atau mengetahui keberadaan seseorang [3]. *Face recognition* dikategorikan menjadi 3 kategori, yaitu *verification*, *identification*, dan *watch* [5].

OpenCV (Open Computer Vision Library) merupakan *library* perangkat lunak *open source* yang memiliki lisensi *BSD-licensed product*. *OpenCV* memiliki lebih dari 2500 algoritma yang telah di optimasi disediakan untuk menangani hal mengenai computer vision dan machine learning. Algoritma yang ada dapat digunakan untuk mendeteksi wajah, mengenali wajah, mengidentifikasi objek, dan lain-lain. Algoritma *OpenCV* untuk pengenalan wajah sampai pada saat penelitian ditulis, telah tersedia tiga algoritma pengenalan wajah, diantaranya *Eigenface*, *Fisherface*, dan *Local Binary Pattern Histogram (LBPH)* [6].

Visar Shehu dan Agni Dika pernah mencoba mengimplementasikan konsep yang hampir mirip dengan penelitian ini, kamera dipasang di ruangan kelas sebagai inputan data citra lalu terhubung dengan *web-server* [7]. Akan tetapi belum adanya bagian yang memonitoring dan mengolah data presensi tersebut, masih berupa verifikasi data pengenalan wajah saja. Selain itu ditemukan kekurangan dengan penggunaan metode *Eigen Face* jika terdapat sampel wajah yang mirip di dalam kelas.

Metode lain yang digunakan selain itu adalah penggunaan gabungan algoritma *Eigen Face* dan *Principal Component Analysis (PCA)* [8]. Namun ternyata ditemukan bahwa *PCA* tergolong lambat untuk mengenali wajah serta kerap menghasilkan data yang kurang akurat.

Pendekatan yang lain yaitu dengan menggabungkan metode *HAAR Cascade* dengan *Eigen Face* [9]. *HAAR Cascade* digunakan sebagai pendeteksi wajah (*face detection*) sedangkan *Eigen Face* digunakan sebagai pengenalan wajahnya (*face recognition*). Berdasarkan hasil akhir ternyata masih ditemukan kurang optimalnya sistem tersebut untuk mengidentifikasi wajah.

Selain itu terdapat beberapa metode lain yang saat ini sedang berkembang seperti *Speeded Up Robust Feature (SURF)*, *Histogram of Oriented Gradients (HOG)* dan

Local Binary Pattern (LBP) [10]. Kortli, Yasin., dkk melakukan penelitian membandingkan beberapa metode seperti *CFs*, *LBP*, *HOG*, *SIFT*, *SURF* dan *BRIEF* didapatkan hasil bahwa *LBP* merupakan metode terbaik saat ini dikarenakan proses pendeteksian dan pengenalan yang relatif cepat dan tingkat akurasi yang tertinggi jika dibandingkan dengan metode yang lain [11].

Local Binary Pattern (LBP) diperkenalkan pertama kali pada tahun 1992 oleh Timo Ojala dan David Harwood di Universitas Maryland. *Local Binary Pattern* didefinisikan sebagai perbandingan nilai biner piksel pada pusat gambar dengan 8 nilai piksel disekelilingnya [12]. *LBP* merupakan metode yang terkenal dalam mengenali sebuah objek. Dalam hal ini, cara yang digunakan adalah membedakan objek dengan *background* [13].

Local Binary Pattern Histogram (LBPH) adalah sebuah kombinasi algoritma antara *LBP* dengan *Histogram of Oriented Gradients(HOG)* [14]. *LBPH* merupakan penyempurnaan performa dari *LBP* dalam proses *face recognition*.

Berikut tahapan-tahapan pada algoritma *Local Binary Pattern Histogram (LBPH)* [15]:

1. Parameters

LBPH memiliki 4 parameter yaitu *radius*, *neighbors*, *grid x* dan *grid y*. Radius digunakan untuk membentuk *circular LBP* biasanya direpresentasikan dengan area lingkaran dengan jari-jari yang terdapat *pixel* inti di bagian tengahnya, besar radius ini secara *default* bernilai 1. *Neighbors* adalah jumlah titik sampel *pixel* yang dibutuhkan untuk membuat *circular LBP*, biasanya bernilai *default* 8. *grid X* adalah jumlah sel *pixel* horizontal sedangkan *grid Y* untuk yang vertikal.

2. Training The Algorithm

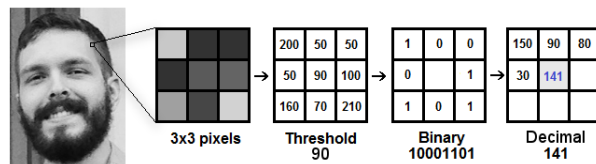
Algoritma hari di *training* tujuannya agar proses *face recognition* kian terus membaik. Proses *training* dilakukan dengan cara membaca *dataset* dan dibuatkan model *training*. Pada proses ini perlu dilakukan pemberian *Id* untuk masing-masing *dataset* agar memudahkan pada proses pengenalan wajah.

3. Applying The LBP Operation

Langkah komputasi dari *LBPH* dengan membuat gambar menengah yang akan menggambarkan gambar asli dengan cara yang lebih baik dalam menangani karakteristik wajah. Proses ini dilakukan dengan menggunakan konsep *sliding window*, berdasarkan parameter *radius* dan *neighbors*.

Pada gambar 2, dapat dilihat proses pembuatan nilai *LBP* yang diambil di setiap *pixel* berukuran 3x3 pada gambar yang sudah dilakukan *grayscale* terlebih dahulu. Nilai yang tertulis di *pixel* merepresentasikan intensitas cahaya dengan rentang 0 – 255. Kemudian nilai pada piksel yang di tengah akan menjadi ambang batas.

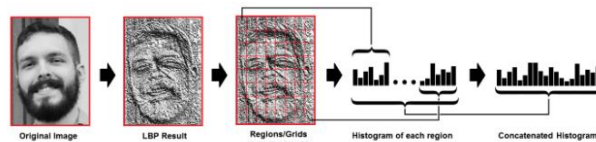
Ambang batas ini akan menentukan 8 nilai piksel lain disekelilingnya. Jika nilai piksel di sekitarnya lebih dari nilai ambang batas, akan diubah menjadi satu, sedangkan jika kurang dari nilai ambang batas akan diubah menjadi 0. Setelah proses selesai maka semua nilai pada window 3x3 itu adalah nilai biner. Nilai biner ini akan disusun searah jarum jam untuk dikonversikan menjadi nilai desimal. Nilai desimal akan menjadi ambang batas pada gambar baru yang akan dibentuk. Proses ini akan dilakukan secara berulang hingga semua piksel pada gambar asli dikonversi dan membentuk gambar baru yang merepresentasikan karakteristik wajah yang lebih baik dari pada gambar asli.



Gambar 2. *LBP Operations*[15]

4. Extracting The Histograms

Pada tahapan ini, gambar yang sudah di buat *generated* akan dibuatkan menjadi beberapa *grid* yaitu *grid x* dan *grid y*. Gambar 3 menunjukkan proses perubahan gambar *grayscale* diproses *LBP* hingga menjadi *histogram*. Karena gambar adalah hasil *grayscale*, maka *histogram* hanya memiliki 256 posisi (0 – 255). Setiap *histogram dataset* akan digabungkan untuk membentuk *histogram* baru yang lebih besar, dimana *histogram* ini akan mewakili karakteristik gambar asli.



Gambar 3. Proses Ekstrak Histogram[15]

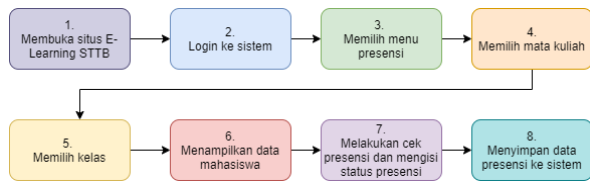
5. Performing The Face Recognition

Pada tahapan ini setelah proses *training* algoritma selesai dan setiap *histogram* yang dibuat sudah mewakili setiap gambar di *dataset*. Langkah berikutnya adalah memberikan *input* data gambar uji yang harus melalui semua proses di tahapan sebelumnya lalu dibandingkan nilai kedekatannya dengan setiap *histogram* dari *dataset* hasil *training*. Dari hasil perbandingan tersebut akan menghasilkan *Id* dari *histogram* yang paling mendekati dengan gambar uji. Selain itu algoritma ini juga akan mengembalikan nilai *confidence* untuk menentukan nilai ambang batas kedekatan *histogram*. Dengan adanya nilai *confidence* tersebut maka jika ada gambar uji yang tidak dikenali maka algoritma akan mengembalikan nilai *Id* yang paling mendekati gambar uji.

2.2. Analisis Sistem Yang Berjalan

Tahapan ini merupakan tahapan untuk menganalisis proses presensi yang sedang berjalan saat ini di STT Bandung. Mengidentifikasi permasalahannya, alur

sistemnya dan keterlibatan pihak didalamnya. Hasil dari analisis tersebut akan dijadikan bahan acuan kebutuhan sistem yang akan dibangun.



Gambar 4. Alur Presensi Yang Sedang Berjalan

Pada sistem yang sedang berjalan ini pada dasarnya sudah berjalan dengan baik namun tahapan yang harus dilakukan masih terbilang cukup banyak dan tidak otomatis sesuai dengan jadwal yang sudah ditentukan. Penyimpanan data presensi ke sistem dalam alur normal hanya bisa dilakukan sekali dan harus menunggu semua mahasiswa masuk ke kelas. Biasanya presensi dilakukan di akhir jam pelajaran dan rata-rata memakan waktu kurang lebih 10-15 menit. Bila terjadi kasus mahasiswa yang surat keterangan izin dan sakit nya menyusul di kemudian hari, maka akses untuk mengubah status mahasiswa tersebut dilakukan oleh pihak akademik dengan melihat bukti fisik yang dilampirkan.

2.3. Analisis Kebutuhan Sistem

Tahapan ini terbagi menjadi dua bagian yaitu analisis kebutuhan perangkat lunak dan analisis kebutuhan perangkat keras. Analisis kebutuhan perangkat lunak merupakan kebutuhan perangkat lunak yang diperlukan untuk dijadikan acuan sistem yang dibangun agar berjalan.

Berikut ini adalah spesifikasi minimum perangkat lunak yang digunakan dalam membangun sistem presensi yang diajukan.

Tabel 1. Tabel Kebutuhan Perangkat Lunak

Device	Software
Server	Python Ver.3 Firebase Storage Firebase Database OpenCV
Client	OS Windows/Linux/Mac

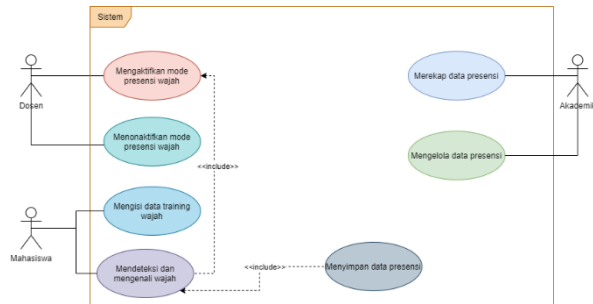
Berikut ini adalah spesifikasi minimum perangkat keras yang digunakan dalam membangun sistem presensi yang diajukan.

Tabel 2. Tabel Kebutuhan Perangkat Keras

Device	Software
IP CCTV	Wireless: WiFi 802.11 b/g/n Connection: Wireless Network Protocol: DHCP, HTTP, P2P, UDP, RTSP Video Resolution: 1080P
Client	RAM 4GB Hardisk Space 1GB

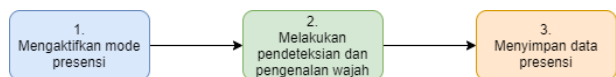
2.4. Perancangan Sistem

Tahapan ini merupakan proses perancangan sistem presensi dengan *face recognition*.



Gambar 5. Use Case Sistem Presensi Usulan

Fungsional sistem yang diusulkan meliputi beberapa aktor yang terlibat seperti dosen, mahasiswa dan akademik. Berikut pemaparan fungsionalnya : Dosen memiliki akses untuk mengaktifkan dan menonaktifkan mode presensi wajah yang terkoneksi dengan IP CCTV di dalam ruangan kelas. Tentu saja diawali dengan login dan akses credential pada umumnya. Jika mode diaktifkan maka pendeteksian dan pengenalan wajah sudah berlangsung. Mahasiswa memiliki akses untuk mengisi data training wajah di sistem. Mahasiswa akan mengisi data berupa beberapa sampel foto wajah secara langsung yang ditangan oleh sistem sebagai bahan uji pengenalan wajah. Hal ini bisa diperbaharui jika diperlukan. Ketika mode presensi diaktifkan maka mahasiswa akan terlibat dalam proses pengecekan dan pengenalan wajah di sistem. Wajah yang sudah dicek dan dikenali akan tersimpan otomatis satu kali setiap pertemuan di database presensi. Akademik memiliki akses untuk melakukan rekap data presensi sesuai yang dibutuhkan, baik per-kelas maupun per-mata kuliah. Selain itu akademik juga memiliki pengelolaan manual data presensi untuk mengubah status presensi mahasiswa jika terjadi kasus surat keterangan izin/sakit yang menyusul.



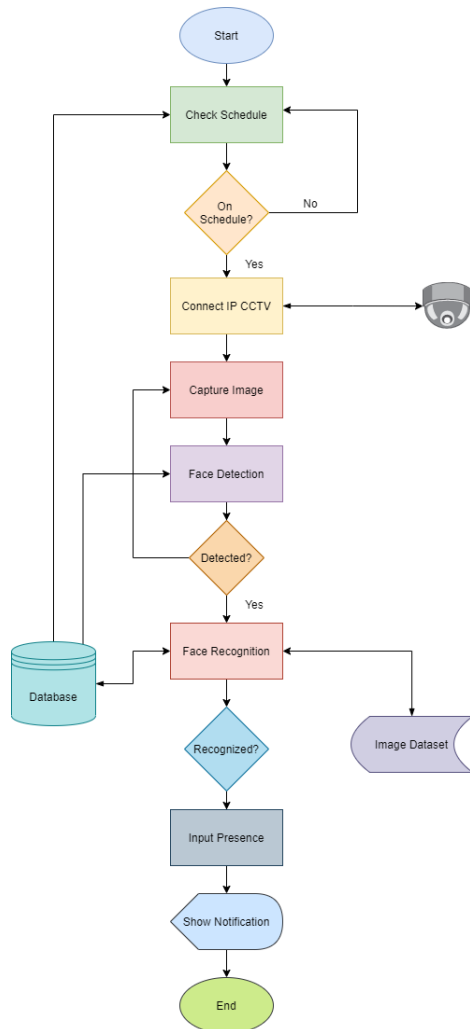
Gambar 6. Proses Presensi Usulan

Dosen mengaktifkan mode presensi kelas yang terkoneksi dengan IP CCTV ruangan kelas. Sistem otomatis membaca kesesuaian jadwal dan kelas sambil melakukan pendeteksian dan pengenalan presensi wajah. Wajah yang berhasil terdeteksi akan disimpan otomatis pada sistem sesuai KBM yang berlangsung

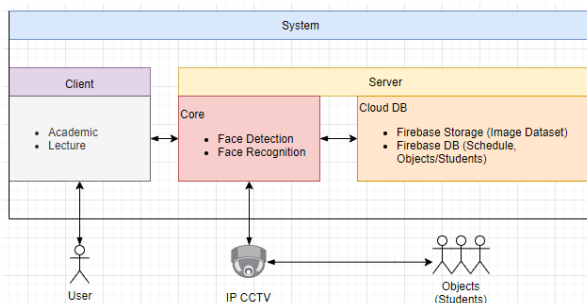
2.5. Implementasi Dan Pengujian

Tahapan ini merupakan tahapan implementasi rancangan ke dalam bentuk code dengan Bahasa pemrograman Python. Setelah tahapan implementasi rampung, maka tahapan berikutnya adalah pengujian kerentanan sistem. Hasil uji akan dijadikan dasar evaluasi keberhasilan sistem yang dikembangkan.

Pada gambar 7, algoritma sistem yg dibangun meliputi 4 tahapan utama yaitu : Pertama pengecekan jadwal KBM, jika sesuai dilanjutkan ke tahapan kedua yaitu pendeteksian wajah dari IP CCTV. Tahapan ketiga adalah tahapan mengenali objeknya dan tahapan terakhir adalah proses input data presensi. Proses ini akan terus dilakukan selama KBM berlangsung.



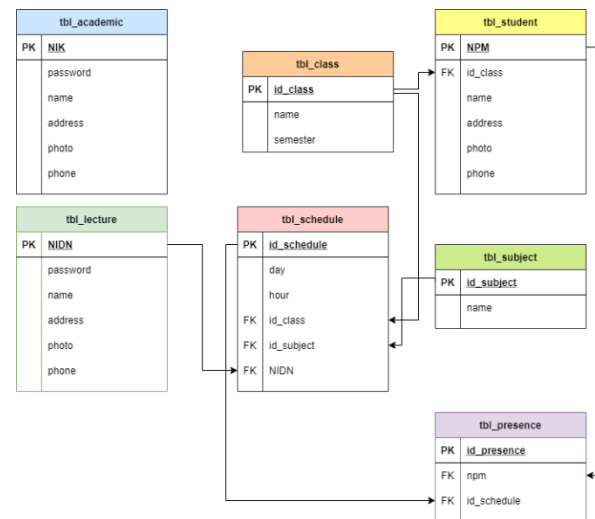
Gambar 7. Algoritma Sistem Presensi



Gambar 8. Arsitektur Sistem

Pada gambar 8, merupakan pemaparan arsitektur sistem yang dibagi menjadi dua bagian yaitu bagian *client/frontend* dengan bagian *server/backend*.

Pada gambar 9, menjelaskan mengenai struktur *database* dan relasi tabel dari sistem presensi yang dikembangkan.



Gambar 9. Skema Database

2.6. Kesimpulan

Penarikan kesimpulan diambil berdasarkan tujuan penelitian yang dikorelasikan dengan hasil capaian penelitian. Keberhasilan tujuan yang dicapai akan dijadikan tolak ukur nilai penerapan sistem yang dikembangkan.

3. Hasil dan Pembahasan

Pada bagian ini akan dipaparkan mengenai hasil pengujian sistem dan pembahasannya. Pengujian sistem dibagi menjadi dua bagian, pertama pengujian dengan data *real* dan kedua berasal dari *Google Facial Expression Comparison Datasheet (FEC Datasheet)* [16].

Pengujian dilakukan atas 30 responden yang terdiri dari 18 laki-laki dan 12 perempuan. Tabel 3 adalah hasil dari pengujian yang dilakukan. Percobaan ini dilakukan sebanyak 5 iterasi di setiap jenis pengujian.

Tabel 3. Tabel Pengujian *Face Recognition (Real)*

Jenis Pengujian	Dikenali	Tidak Dikenali	Tingkat Sukses	Tingkat Gagal
Cahaya				
- Terang	138	12	90%	10%
- Redup	132	18		
Posisi				
- Miring	120	30	83%	17%
- Lurus	129	21		
Ekspresi				
- Tersenyum	129	21	87%	13%
- Datar	132	18		
Jarak				
- 100-250cm	136	14	87%	13%
- 300-500cm	125	25		
Atribut				

- Penutup Kepala	128	22	85%	15%
- Kacamata	127	23		
Validitas			86.4%	13.6%

Didapatkan hasil validitas data dengan tingkat keberhasilan sebesar 86.4%.

Pengujian kedua menggunakan data (*FEC Datasheet*). Total wajah yang dimiliki oleh *FEC Datasheet* ini sebanyak 155,943. Namun pada pengujian kali ini yang digunakan hanya 32, 64 dan 128 dengan jumlah *data training* 100.

Tabel 4. Tabel Pengujian *Face Recognition (FEC Datasheet)*

Jumlah Image	Dikenali	Tidak Dikenali	Error Debug	Tingkat Sukses	Tingkat Gagal
32	28	3	1	90%	10%
64	55	8	1	87%	13%
128	108	19	1	85%	15%
Validitas				87.3 %	12.7%

Didapatkan hasil presentase tingkat keberhasilan sebesar 87.3% dengan 3 iterasi pengambilan data dengan jumlah *image* yang berbeda.

Jika tingkat keberhasilan dari kedua pengujian tersebut diambil rata-ratanya maka akan didapatkan hasil dengan tingkat keberhasilan sebesar 86.85%. Sedangkan tingkat kegagalannya sebesar 13.15%.

4. Kesimpulan

Berdasarkan dari hasil pengujian diatas dapat disimpulkan dengan hasil 86.85% proses pengenalan wajah untuk sistem presensi ini dapat dinyatakan sangat baik.

Namun jika dilihat dari tingkat kegagalannya yang masih menyentuh data 13.15% maka dari hasil evaluasi didapatkan penyebab terbesar adalah data *training/sample* yang dijadikan *datasheet* yang kurang jelas gambarnya. Pengambilan data tersebut harus dilakukan dengan baik agar proses pengenalan dapat berfungsi optimal. Pengambilan *data training* harus jelas dalam berbagai posisi, ekspresi maupun cahaya dengan pengambilan data training maksimum 100 cm. Selain itu, disarankan akan lebih baik jika dapat melakukan koneksi otomatis ke *IP CCTV* terdekat untuk [17]

menghindari salah sambung koneksi dan memudahkan proses presensi.

Daftar Rujukan

- [1] Firdaus, M.I., Ridho, I.I., 2016. Aplikasi Pendeteksi Wajah Menggunakan Metode Haar. *Media Sains*, 9 (1). pp.27-31.
- [2] Suheri, C., Ruslianto, I., 2017. Identifikasi Wajah Manusia untuk Sistem Monitoring Kehadiran Perkuliahan menggunakan Ekstraksi Fitur Principal Component Analysis (PCA). *Jurnal Edukasi dan Penelitian Informatika (JEPIN)*, 3 (1). pp.9-15.
- [3] Luisan W.A., Steven R.S., Alwin M.S., 2017. Implementasi Algoritma Pengenalan Wajah Untuk Mendeteksi Visual Hacking. *E-Journal Teknik Informatika*. 11 (1).
- [4] Chi Ho Chan, 2008. Multi-scale Local Binary Pattern Histogram for Face Recognition, *School of Electronics and Physical Sciences, University of Surrey, Guildford, Surrey*.
- [5] Chakka M., Venugopal P., 2016. Face Detection and Recognition Using LBPH. *International Journal of Engineering Sciences & Research Technology (IJERST)*, 5 (3). pp.10-16.
- [6] OpenCV, 2020. *About*. [Online] (Updated 15 Jun 2020). Tersedia di : <http://opencv.org/about.html>. [Accessed 18 Sep 2020].
- [7] Shehu, V., Dika, Agni., 2010. Using Real Time Computer Vision Algorithms in Automatic Attendance Management Systems. *32nd International Conference on Information Technology Interfaces (ITI)*. Cavtat, Croatia, 21-24 Jun 2010. IEEE : Croatia.
- [8] Wagh, P., Thakare, R., Chaudhari, J., Patil, S., 2015. Attendance System Based On Face Recognition Using Eigen Face And PCA Algorithms. *International Conference on Green Computing and Internet of Things (ICGCIoT)*. 1. pp.303-308.
- [9] Fuzail, M., Muhammad, H., Nouman, F., 2014. Face Detection System for Attendance of Class Students. *International Journal of Multidisciplinary Science and Engineering*, 5 (4). pp. 6-10.
- [10] Shafiqul, M., Mahmud, A., Papeya, A.A., Onny, I.S., 2017. Real Time Classroom Attendance Management System. *Department of Computer Science and Engineering. BRAC University*.
- [11] Kortli, Yassin., Jridi, Maher., Al Falou, Ayman., Atri, Mohamed., 2018. A Comparative Study of CFs, LBP, HOG, SIFT, SURF, and BRIEF Techniques for Face Recognition. *Proc. SPIE 10649, Pattern Recognition and Tracking XXIX*. 10649.
- [12] M.P. Inen., M. Pietikäinen., A. Hadid., G. Zhao., T. Ahonen., 2011. *Computer Vision Using Local Binary Patterns*. Springer.
- [13] M. S. Karis., N. R. A. Razif., N. M. Ali., M. A. Rosli., M. S. M. Aras., M. M. Ghazaly., 2016. Local Binary Pattern (LBP) with application to variant object detection: A survey and method. *IEEE 12th Int. Colloq. Signal Process. its Appl. CSPA 2016, March*. pp. 221–226.
- [14] F. Deeba., A. Ahmed., H. Memon., F. A. Dharejo., A. Ghaffar., 2019. LBPH-based enhanced real-time face recognition. *Int. J. Adv. Comput. Sci. Appl.*, 10 (5). pp.274–280.
- [15] Hanuebi, A., Sompie, S., Kambey, F., 2019. Aplikasi Pengenalan Wajah Untuk Membuka Pintu Berbasis Raspberry Pi. *Jurnal Teknik Informatika.*, 14 (2). pp.243-252.
- [16] R Vemulapalli, A Agarwala., 2018. A Compact Embedding for Facial Expression Similarity. *CoRR*, abs/1811