

PERENCANAAN DAN IMPELEMENTASI *IMAGE PROCESSING* UNTUK ABSENSI KEHADIRAN

DESIGN AND IMPLEMENTATION OF IMAGE PROCESSING FOR ABSENCE ATTENDANCE

Zainul Muarifin¹, Denny Darlis, S.Si., M.T.², Atik Novianti, S.ST., M.T.³

^{1,2,3}Prodi D3 Teknologi Telekomunikasi, Universitas Telkom

¹zainulmuarifin@student.telkomuniversity.ac.id, ² enny.darlis@tass.telkomuniversity.ac.id,

³atiknovianti@tass.telkomuniveristv.ac.id

Abstrak

Presensi mahasiswa sekarang dilakukan di Program Studi Teknologi Telekomunikasi Universitas Telkom, masih menggunakan RFID sehingga memberi peluang tindakan yang kurang diinginkan seperti kehilangan RFID, mengisi presensi dititipkan oleh mahasiswa lainnya, dan kurang efisien harus membawa kemana-mana presensi ketika sedang ada perkuliahan. Penelitian sebelumnya sudah ada yang menggunakan sistem pengenalan wajah pada absensi kehadiran yaitu perancangan pengenalan pola wajah untuk aplikasi absensi dengan metode *eigenface*. Tetapi masih ada kekurangan di beberapa faktor yaitu, cahaya dan jarak. Pada proyek akhir ini akan dilakukan perancangan dan implementasi image processing untuk absensi kehadiran. Dari beberapa peneliti menunjukkan bahwa dengan metode LBPH dengan akurasi yang lebih baik dari pada *eigenface*. Maka dari situlah penulis tertarik mencoba sistem deteksi wajah untuk presensi dengan metode LBPH. Simulasi perancangan ini akan dilakukan menggunakan *software PyCharm* dengan menggunakan bahasa pemrograman python. Hasil dari uji coba yang dilakukan dengan menggunakan tiga jenis pengujian yaitu jarak, sudut wajah dan aksesoris. Uji coba dilakukan 10 kali percobaan dengan kondisi di dalam ruangan, dengan pengujian jarak yang digunakan dari 5cm sampai 100 cm, pengujian sudut dari 0 derajat sampai 15 derajat, dan pengujian aksesoris menggunakan kacamata. Hasil dari semua pengujian pengenalan wajah dikenali dengan baik dengan rata-rata keberhasilan 92%. Dan untuk pengiriman data absensi ke-database terkirim sesuai yang diharapkan.

Kata Kunci: *Image Processing, Pengenalan Wajah, LBPH(Local Binary Patterns Histogram).*

Abstract

Presence of students is now carried out in the Telecommunications Technology Study Program Telkom University, still using RFID so as to provide opportunities for undesirable actions such as loss of RFID, fill the presence deposited by other students, and less efficiently have to carry the presence everywhere when there are lectures. Previous studies have used face recognition systems in attendance, namely the design of facial pattern recognition for attendance applications with the eigenface method. But there are still deficiencies in several factors namely, light and distance. In this final project will be carried out the design and implementation of image processing for attendance attendance. From some researchers show that with the LBPH method with better accuracy than eigenface. So that's where the authors are interested in trying a face detection system for presence with the LBPH method. This design simulation will be carried out using PyCharm software using the Python programming language. The results of the trials were carried out using three types of tests, namely: distance, face angle and accessories. The trial was carried out 10 times with indoor conditions, with testing the distance used from 5cm to 100 cm, testing the angle from 0 degrees to 15 degrees, and testing accessories using glasses. The results of all facial recognition tests were well recognized with a success rate of 92%. And for sending absent data to the database sent as expected.

Keywords: *Image Processing, Facial Recognition, LBPH(Local Binary Patterns Histogram).*

1. PENDAHULUAN

Seiring perkembangan zaman, tentunya teknologi juga akan semakin berkembang. Semakin banyak inovasi-inovasi teknologi baru dan juga perkembangan dari teknologi yang sudah ada sebelumnya untuk memberikan dampak positif dan memudahkan kebutuhan manusia. Salah satunya adalah *Image Processing*. *Image Processing* ada banyak implementasi dalam *Image Processing* salah satunya adalah pengenalan wajah. Pengenalan wajah adalah teknologi pengenalan wajah yang saat ini tengah dalam masa pengembangan. Teknologi ini sudah masuk ke bidang Pendidikan yaitu sebagai absensi kehadiran. Untuk saat ini absensi menggunakan pengenalan wajah masih jarang terutama di Indonesia. Di Indonesia masih banyak yang menggunakan absensi manual atau masih menggunakan RFID.

Pada penelitian sebelumnya sudah digunakan sistem RFID pada Absensi Kehadiran yaitu sistem absensi berbasis RFID [10]. Namun masih ada kelemahan yaitu masih banyak yang melakukan kecurangan saat absensi dan membutuhkan waktu cukup lama saat proses absensi. Penelitian sebelumnya sudah ada yang menggunakan sistem pengenalan wajah pada absensi kehadiran yaitu perancangan pengenalan pola wajah untuk aplikasi absensi dengan metode eigenface [3]. Namun masih ada kelemahan saat menggunakan metode eigenface, karena dalam perangkat lunak ini dipengaruhi beberapa factor seperti cahaya, dan jarak. Pada Penelitian ini menggunakan metode LBPH (*Local Binary Patterns Histogram*) dengan akurasi yang lebih baik di banding dengan metode eigenface [7].

Pada proyek akhir ini akan dilakukan perancangan dan implementasi *image processing* untuk absensi kehadiran. Metode pengenalan wajah yang akan digunakan adalah algoritma LBPH (*Local Binary Patterns Histogram*). Dalam proses pembuatan penelitian ini menggunakan bahasa python dan pemrosesan dengan opencv. Proses pengenalan citra pada identifikasi wajah dilakukan dengan adanya masukkan berapa citra untuk di cocokkan dengan citra yang disimpan dalam sistem memori. Sebelum citra dilakukan deteksi wajah menggunakan algoritma viola-jones. Setelah itu citra di konversi ke bentuk *grayscale* dan di simpan di memori [11]. Kemudian dilakukan perhitungan untuk setiap citra yang menjadi data latihan yang ada dalam memori yang akan digunakan untuk pengenalan wajah yang akan melakukan absensi.

2. Dasar Teori

2.1 Image Processing

Image Processing atau pengolahan citra merupakan pengolahan citra agar citra lebih baik dari pada sebelumnya dengan menggunakan komputer. Pengolahan citra bertujuan untuk memperbaiki kualitas citra agar mudah diinterpretasikan oleh manusia atau mesin. Contohnya operasi pengolahan citra lainnya adalah penghilang derau (*noise*) pada citra, pengubahan kontras citra, penajaman dan lain-lain [13]. Dalam pengolahan citra terdapat dua jenis pengolahan, yaitu pengolahan citra analog dan pengolahan citra digital [2].

2.2 Computer Vision

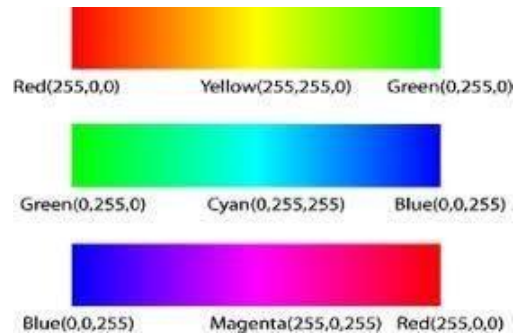
Computer vision merupakan ilmu yang mempelajari bagaimana komputer dapat mengenali objek yang diamati dengan cara mengekstrak informasi dari objek tersebut [4]. Cabang ilmu ini berkaitan dengan Artificial Intelligence yang mampu menghasilkan Visual Intelligence System. Computer Vision adalah kombinasi antara :

1. Pengolahan Citra Merupakan bidang yang berhubungan dengan proses memperbaiki citra/gambar yang bertujuan untuk mendapatkan kualitas citra yang lebih baik.
2. Pengenalan Pola Merupakan bidang yang berhubungan dengan proses identifikasi obyek pada citra atau interpretasi citra untuk mengekstrak informasi dari citra tersebut.

2.3 Warna

2.4.1 RGB

RGB (Red Green Blue) adalah model warna yang terdiri dari merah, hijau, dan biru. Ketiga warna jika digabungkan akan membentuk warna lainnya. Setiap warna dasar, misalnya merah, dapat diberi rentang-nilai. Untuk nilai rentangnya dari 0 sampai 255. Pilihan skala 256 ini didasarkan pada cara mengungkap 8 digit bilangan biner yang digunakan oleh mesin komputer. Dengan cara ini, akan diperoleh warna campuran sebanyak $256 \times 256 \times 256 = 1677726$ jenis warna [10]. Jenis warna tersebut seperti sebuah vektor di ruang 3 dimensi yang memiliki titik koordinat yang dinyatakan dalam bentuk 3 bilangan yaitu, komponen-x, komponen-y, komponen-z. Contoh jika ada sebuah vektor dituliskan sebagai $r=(x,y,z)$, maka x-y-z tersebut diganti dengan komponen R(ed), G(reen), B(lue).



Gambar 2.1 Warna RGB

2.4.2 Grayscale

Grayscale yaitu citra yang memiliki tingkatan warna dominan abu-abu, dimana warna pada ruang warna RGB(*Red, Green, Blue*) dengan komponen warna merah, hijau, dan biru dengan intensitas yang sama, untuk citra keabuan disimpan sebagai 8 bit sehingga memberikan $2^8 = 256$ keabuan dari warna hitam sampai warna putih

2.4.3 Warna Biner

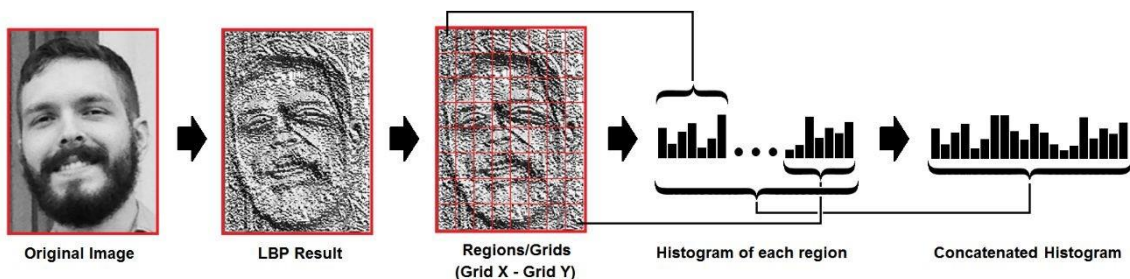
Sebuah citra biner dapat dianggap sebagai tipe khusus dari citra intensitas yang hanya berisi hitam (0) dan putih (1). Selain itu, kita juga dapat menyatakan citra biner sebagai citra ber-indeks dengan hanya dua warna. Algoritma untuk pengolahan citra biner saat ini telah berkembang dengan baik dan dengan mudah dimengerti oleh manusia sehingga masih banyak digunakan dalam pengolahan citra digital. Selain itu dengan jumlah bit untuk tiap pikselnya yang lebih sedikit membuat proses pengolahannya juga lebih cepat.

2.4 Metode Viola-Jones

Metode Viola-Jones merupakan sebuah metode yang digunakan untuk mendeteksi objek seperti manusia, mobil, roda mobil, wajah, dan lain sebagainya. Metode ini banyak digunakan untuk mendeteksi objek karena metode ini memiliki algoritma yang efisien sehingga tidak memerlukan waktu yang banyak untuk melakukan proses pendeteksian objek [12]. Proses pendeteksian objek dilakukan dengan membandingkan citra terhadap sebuah pengklasifikasi yang dibentuk dari data pelatihan.

2.5 Local Binary Pattern Histogram (LBPH)

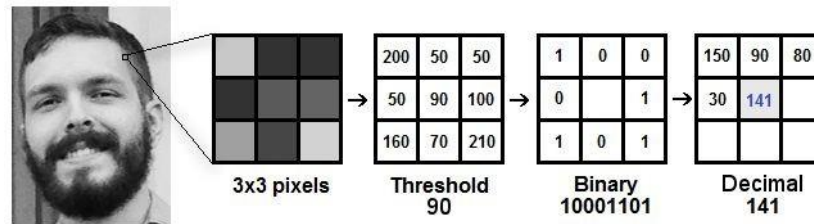
Local Binary Pattern Histogram (LBPH) adalah algoritma untuk mengklasifikasi yang dikombinasikan dengan histogram dan merupakan teknik baru dari metode LBP (*Local Binary Patterns*) untuk mengubah performa hasil pengenalan wajah. LBP (*Local Binary Pattern*) pada umumnya didesain untuk pengenalan tekstur. LBPH (*Local Binary Patterns Histogram*) adalah metode yang paling cocok untuk dilakukan pengenalan citra [1].



Gambar 2.2 Metode LBPH

Pada Metode LBPH, citra dibagi menjadi beberapa bagian, kemudian dilakukan proses ekstraksi ciri untuk mendapatkan nilai histogram dengan mengubah setiap pixel dari citra menjadi thresholding dengan ukuran 8×8 dengan diambil satu pixel tengah untuk menjadi pembanding. Jika nilai tetangga lebih besar atau sama besar dari pixel yang menjadi pembanding dalam nilai grayscale maka akan diubah nilainya menjadi 1 dan sebaliknya jika nilai tetangga lebih kecil dari pembanding dalam nilai grayscale maka akan diubah nilai pixelnya menjadi 0.

Setelah diubah menjadi nilai biner maka akan menghasilkan deret biner yang kemudian jika diubah menjadi nilai decimal akan menghasilkan nilai baru untuk nilai tengah pixel selanjutnya, deret biner tersebut yang disebut dengan kode LBP (*Local Binary Patterns*).



Gambar 2.3 Proses LBPH

Tahapan LBPH dan proses mendapatkan nilai decimal dengan mengubah pixel 3x3 dapat dilakukan konversi:

1. Pixel yang didapat dengan cara sistem pencuplikan 3x3 pixels
2. Setiap sel memiliki nilai BWthreshold yang berbeda dengan diatur sesuai tingkat ke-abuan masing-masing sel dengan nilai 0-255
3. Menggunakan nilai tengah sebagai titik tengah (*middle point*) sebagai parameter penentuan nilai bit 0 dan 1
4. Bernilai 0 ketika nilai tengah memiliki nilai yang lebih besar dibanding nilai tetangga (*neighbors*) dan bernilai 1 ketika nilai tengah memiliki nilai tengah sama dengan tetangga (*neighbors*) atau lebih kecil
5. Setelah mendapatkan *Binary dari threshold* dan pixel yang diambil adalah 10001101 kemudian dilakukan konversi ke decimal menggunakan pengalihan
6. Tuliskan angka binernya. Cara ini tidak menggunakan kuadrat. Sehingga, lebih mudah untuk mengubah angka besar dan hanya perlu mengingat jumlahnya saja
7. Mulailah dari kiri, kalikan dua total penjumlahan sebelumnya dan tambahkan digitnya.
8. Kalikan dua total penjumlahan Anda sekarang dan tambahkan digit selanjutnya dan ulangi kemudian jumlahkan hasilnya untuk mendapatkan nilai decimal
9. Konversi biner ke desimal dimulai dari $0x2+1=1$, $1x2+0=2$, $2x2+0=4$, $4x2+0=8$, $8x2+1=17$, $17x2+1=35$, $35x2+0=70$, $70x2+1=141$
10. Hasil akhir Konversi biner 10001101 menjadi desimal adalah 141

2.6 Python

Python merupakan bahasa pemrograman yang mudah dibaca dan dimengerti. Selain itu, python juga memiliki library yang cukup lengkap sehingga memudahkan *programmer* untuk membuat aplikasi yang memiliki banyak fitur dengan menggunakan *source code* yang tampak sederhana [9].

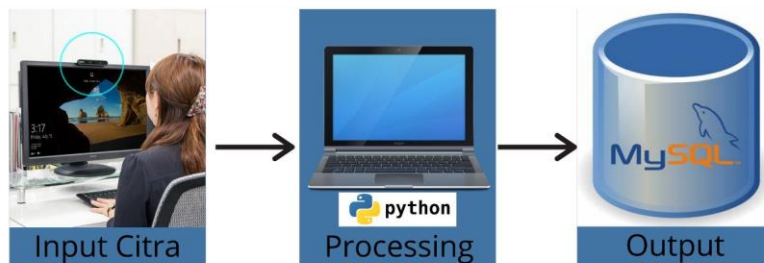
2.7 OpenCV

OpenCV (Open Computer Vision) adalah sebuah *API (Application Programming Interface) Library* yang sudah sangat populer untuk digunakan pada Pengolahan Citra dan *Computer Vision*. *Computer Vision* itu sendiri adalah salah satu cabang dari Pengolahan Citra (*Image processing*) yang dapat membuat komputer bisa melihat seperti manusia. Dengan *vision* tersebut komputer dapat mengambil keputusan, melakukan aksi, dan mengenali terhadap suatu objek. Beberapa pengimplementasian dari *Computer Vision* adalah *Face Recognition*, *Face Detection*, *Face/Object Tracking*, *Road Tracking*, dll. *OpenCV* adalah *library Open Source* untuk *Computer Vision* dan memiliki fungsi akuisisi yang baik untuk image/video.

3. Perancangan Sistem Informasi Absensi

3.1 Deskripsi Proyek Akhir

Pada Proyek akhir ini akan dilakukan perancangan dan implementasi image processing untuk absensi kehadiran. Pada perancangan ini dipilih teknologi pengenalan wajah untuk digunakan absensi kehadiran dengan metode *Local Binary Patterns Histogram (LBPH)*.



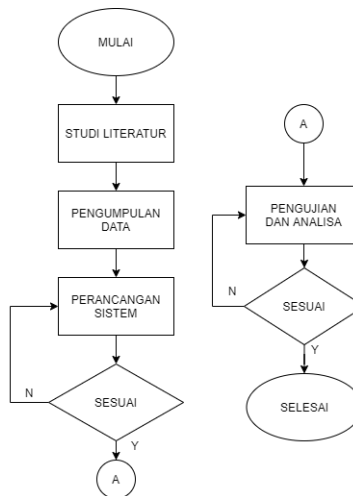
Gambar 3. 1 Model Sistem

Pada Gambar 3.1 merupakan model sistem absensi kehadiran yang akan dijelaskan tahapan atau alur dari input citra pendaftaran sampai mendapatkan output.

Pada tahap input citra dibagi menjadi dua yaitu input untuk data yang akan digunakan untuk diuji dan input yang digunakan untuk absensi kehadiran. Tahap selanjutnya adalah *prosesing*, semua proses pemrograman yang dilakukan dengan menggunakan bahasa python, dari menghidupkan kamera untuk input citra hingga proses absensi sehingga mendapatkan output yang diinginkan. Output disini menggunakan *database mysql*.

3.2 Proses Pengerjaan Sistem

Pada proyek akhir ini akan dilakukan proses pengerjaan sistem absensi kehadiran. Diagram alir tahapan yang akan dilakukan, bisa dilihat pada Gambar 3.2



Gambar 3.2 Diagram Proses Pengerjaan PA

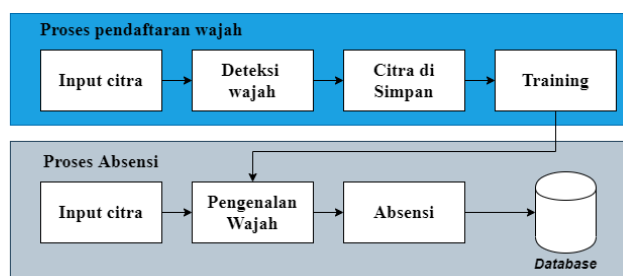
Pada bagian ini akan dijelaskan tahapan dan alur proses pengerjaan sistem absensi kehadiran yang dilakukan melalui beberapa tahap. Tahap pertama, studi literatur dimana dilakukan dengan mengumpulkan literatur-literatur dan kajian-kajian yang berkaitan dengan permasalahan yang ada pada penelitian proyek akhir ini, baik berupa buku referensi, artikel, maupun *e-journal* yang berhubungan dengan perancangan sistem absensi kehadiran.

Tahap kedua, melakukan pengumpulan data berupa data-data yang akan digunakan untuk proses pembuatan data training maupun data uji. Tahap ketiga, perancangan sistem dimulai dari tahap pendaftaran wajah untuk digunakan sebagai uji pengenalan wajah, sampai melakukan proses pengenalan wajah untuk absensi kehadiran. Jika, belum sesuai maka akan dirancang ulang hingga sesuai, dan jika sudah sesuai akan dilanjutkan ke tahap selanjutnya.

Tahap keempat, melakukan pengujian sistem absensi dari pendaftaran wajah hingga melakukan absensi dan menemukan output yang sesuai. Jika tidak sesuai maka akan dilakukan pengujian lagi hingga sesuai. Setelah pengujian sesuai perlu dilakukan Analisa dengan parameter-parameter yang sudah ada.

3.3 Perancangan Sistem Absensi Kehadiran

Dalam melakukan perancangan sebuah sistem absensi kehadiran ini dilakukan menggunakan perangkat lunak pycharm. Berikut blok diagram sistem secara umum Gambar 3.3 dan flowchart alur perancangan sistem ini dapat dilihat pada Gambar 3.4



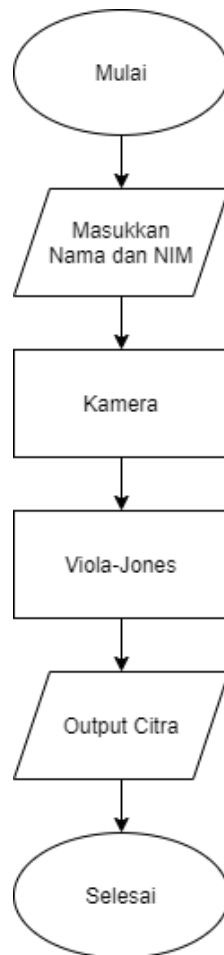
Gambar 3.3 Blok Diagram Sistem Secara Umum

Berikut ini tahap-tahap yang dilakukan untuk mendeteksi wajah manusia, jalanya model blok diagram sistem secara umum adalah sebagai berikut :

Pada proses pendaftaran wajah langkah pertama adalah memasukkan data wajah dengan cara mengaktifkan kamera untuk mengambil wajah lalu wajah akan di deteksi dengan algoritma viola-jones dan wajah akan di simpan di penyimpanan sistem dan akan dilatih untuk digunakan saat pengenalan wajah.

Pada proses absensi kita harus memasukkan wajah dengan cara mengaktifkan kamera lalu objek akan dikenali dengan menggunakan metode LBPH yang telah dicocokkan dengan data wajah yang sudah dilatih sebelumnya. Saat wajah dikenali maka objek akan di anggap melakukan absensi kehadiran dan data absensinya akan di kirim ke *database*.

3.3.1 Perancangan Pendaftaran wajah



Gambar 3.4 Flowchart Deteksi Wajah

Pada Gambar 3.4 merupakan tahap pendaftaran wajah yang dirancang dalam beberapa tahap. Tahap pertama memasukkan nama dan nim yang akan didaftarkan.

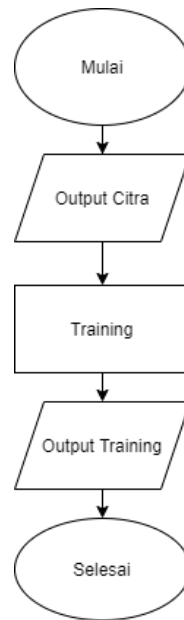
Pada tahap kedua kamera akan memproses dan menampilkan citra yang akan di deteksi wajahnya. Di tahap ketiga akan diproses pendeteksian wajah dengan algoritma viola-jones. Prosesnya bisa dilihat di Gambar 3.5.

```
harcascadePath = "haarcascade_frontalface_default.xml"
detector = cv2.CascadeClassifier(harcascadePath)
sampleNum = 0
while (True):
    ret, img = cam.read()
    gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
    faces = detector.detectMultiScale(gray, 1.3, 5)
    for (x, y, w, h) in faces:
        cv2.rectangle(img, (x, y), (x + w, y + h), (255, 0, 0), 2)
        sampleNum = sampleNum + 1
        cv2.imwrite("DataWajah" + os.sep + name + "." + NIM + '.' +
                    str(sampleNum) + ".jpg", gray[y:y + h, x:x + w])
        cv2.imshow('frame', img)
    if cv2.waitKey(100) & 0xFF == ord('q'):
        break
    elif sampleNum > 30:
        break
```

Gambar 3.5 Proses Viola Jones

Pada tahap selanjutnya output yang dihasilkan dari metode viola-jones akan di simpan dimemori sistem. Data output digunakan untuk bahan training yang akan digunakan.

3.3.2 Perancangan Pelatihan Wajah

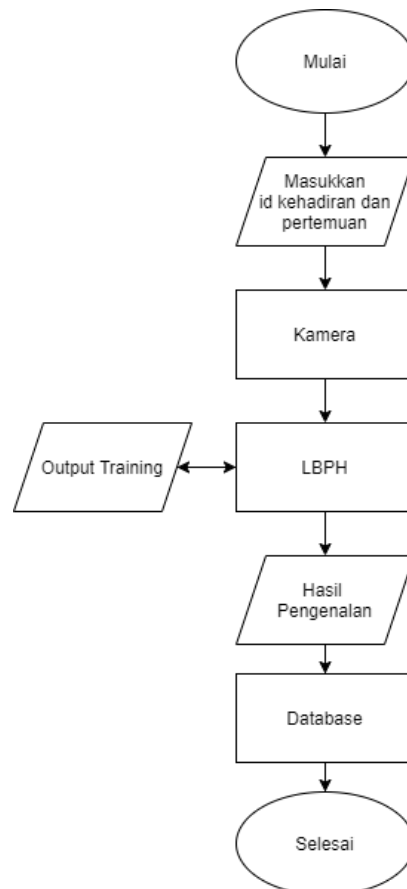


Gambar 3.6 Flowchart Training

Pada Gambar 3.6 menjelaskan tentang training data yang digunakan untuk mencocokkan data citra diproses pengenalan wajah, berikut tahap tahap yang dilakukan.

Pada tahap pertama output citra yang tersimpan dimemori sistem akan diambil datanya. Pada tahap kedua data yang diambil dari memori sistem akan ditraining. Tahap terakhir data yang sudah ditraining akan disimpan dalam format *.xml dan disimpan di memori sistem.

3.3.3 Perancangan Pengenalan Wajah



Gambar 3.7 Flowchart Pengenalan Wajah

Pada Gambar 3.7 menjelaskan tentang tahap pengenalan wajah untuk absensi. Tahap pertama yaitu memasukkan id kehadiran dan pertemuan. Tahap kedua kamera akan memproses untuk pengenalan wajah.

Tahap ketiga proses pengenalan wajah menggunakan metode LBPH, dan akan dicocokkan dengan data training. Proses LBPH bisa dilihat di Gambar 3.8

```

ret, im = cam.read()
gray = cv2.cvtColor(im, cv2.COLOR_BGR2GRAY)
faces = faceCascade.detectMultiScale(gray, 1.2, 5)
for(x, y, w, h) in faces:
    cv2.rectangle(im, (x, y), (x+w, y+h), (225, 0, 0), 2)
    Id, conf = recognizer.predict(gray[y:y+h, x:x+w])
    if(conf < 50):
        Kehadiran = 'H'
        aa = df.loc[df['NIM'] == Id]['Name'].values
        tt = aa
        attendance.loc[len(attendance)] = [matkul, Id, Kehadiran, pertemuan]
    else:
        Id = 'Unknown'
        tt = str(Id)
    if(conf > 75):
        noOfFile = len(os.listdir("GambarTidakDiketahui"))+1
        cv2.imwrite("GambarTidakDiketahui"+os.sep+"Image"+str(noOfFile) +
            ".jpg", im[y:y+h, x:x+w])
        cv2.putText(im, str(tt), (x, y+h), font, 1, (255, 255, 255), 2)
    attendance = attendance.drop_duplicates(subset=['nim'], keep='first')
    cv2.imshow('im', im)
    if (cv2.waitKey(1) == ord('q')):
        break
  
```

Gambar 3.8 Proses LBPH

Tahap selanjutnya output dari pengenalan wajah akan mendapatkan hasil seperti berikut : id_matakuliah, NIM, Status_Kehadiran, dan pertemuan. Tahap terakhir dari pengenalan wajah adalah data output akan dikirim ke database.

3.4 Pembuatan Layout Sistem Absensi Kehadiran

Pada bab pembuatan *layout* sistem kehadiran ini menggunakan bahasa python dan menggunakan *library* tkinter. Tkinter adalah library untuk pembuatan gui untuk python. Berikut gambar *layout* sistem, dan pemrograman *layout*.

1. AppAdmin

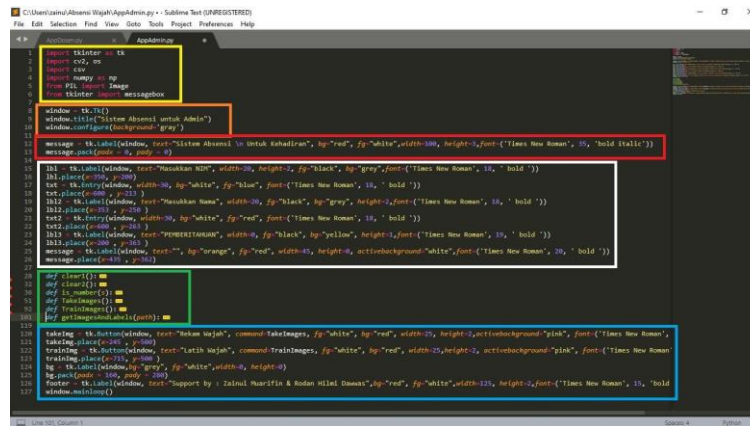
Berikut ini adalah tampilan untuk *layout* yang sudah di rancang sebelumnya.



Gambar 3.9 *Layout Aplikasi Admin*

Pada Gambar 3.5 merupakan tampilan *layout* appadmin. Pada tampilan tersebut terdapat tiga label yang digunakan untuk menampilkan informasi untuk membantu dalam menggunakan aplikasi, lalu terdapat dua masukkan yang digunakan untuk memasukkan data yang akan digunakan memberikan kode unik agar data tertata. Lalu ada pemberitahuan fungsinya untuk menampilkan pemberitahuan saat aplikasi dijalankan. Dan terdapat pula dua tombol yang berfungsi untuk menjalankan pemrograman yang sudah diatur sebelumnya.

Selanjutnya ada tahap pemrograman sistem, berikut gambar pemrograman sistem absensi kehadiran.



Gambar 3.10 *Pemrograman Aplikasi Admin*

Pada Gambar 3.6 penjelasan tentang pemrogramannya. Pada kotak warna kuning adalah library yang digunakan untuk memproses pemrograman. Pada kotak warna oranye adalah pemrograman untuk pembuatan gui. Pada kotak warna merah adalah pemrograman untuk nama jendela aplikasi yang dibuat. Pada kotak warna putih adalah pemrograman untuk *layout* yang ada di dalam aplikasi yaitu label, masukan, dan notifikasi. Pada kotak warna hijau adalah pemrograman untuk menjalankan perekaman wajah. Dan pada kotak warna biru adalah pemrograman tombol yang digunakan untuk mengakses pemrograman pada kotak warna hijau.

2. AppDosen

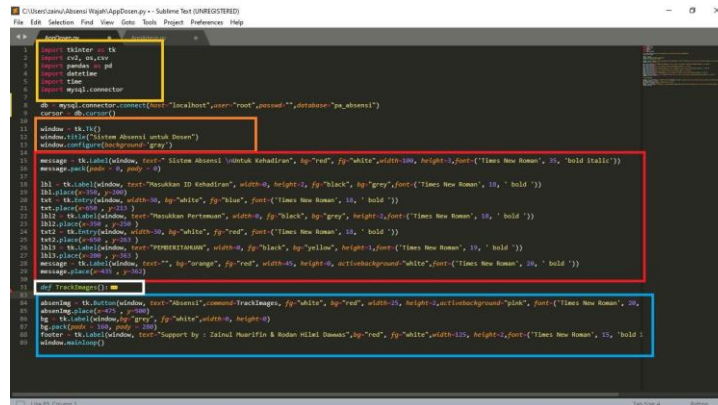
Berikut ini adalah tampilan untuk *layout* yang sudah di rancang sebelumnya.



Gambar 3.11 Layout Aplikasi Dosen

Pada Gambar 3.8 merupakan tampilan *layout* appadmin. Pada tampilan tersebut terdapat tiga label yang digunakan untuk menampilkan informasi untuk membantu dalam menggunakan aplikasi, lalu terdapat dua masukan yang digunakan untuk memasukkan data yang akan digunakan memberikan kode unik agar data tertata. Lalu ada pemberitahuan fungsinya untuk menampilkan pemberitahuan saat aplikasi dijalankan. Dan terdapat pula satu tombol yang berfungsi untuk menjalankan pemrograman yang sudah diatur sebelumnya.

Selanjutnya ada tahap pemrograman sistem, berikut gambar pemrograman sistem absensi kehadiran.



Gambar 3.12 Pemrograman Layout AppDosen

Pada Gambar 3.6 penjelasan tentang pemrogramannya. Pada kotak warna kuning adalah library yang digunakan untuk memproses pemrograman. Pada kotak warna oranye adalah pemrograman untuk pembuatan gui dan pemrograman untuk nama jendela aplikasi yang dibuat. Pada kotak warna merah adalah pemrograman untuk *layout* yang ada di dalam aplikasi yaitu label, masukan, dan notifikasi. Pada kotak warna putih adalah pemrograman untuk menjalankan absensi wajah. Dan pada kotak warna biru adalah pemrograman tombol yang digunakan untuk mengakses pemrograman pada kotak warna hijau.

3.5 Kebutuhan Sistem

Pada bab ini adalah membahas tentang kebutuhan sistem pada proyek akhir ini, maka dibutuhkan adanya perangkat keras dan perangkat lunak yang dibutuhkan agar berjalan dengan baik.

3.5.1 Perangkat Keras

Perangkat keras yang dibutuhkan pada proyek akhir ini adalah.

1. Laptop, laptop yang digunakan untuk perancangan sistem ini memiliki spesifikasi sebagai berikut.
 - Merk : Asus A456U
 - Ram : 4 GB
 - Os : Windows 10
2. Kamera Kamera yang digunakan adalah kamera webcam dari laptop yang memiliki spesifikasi sebagai berikut :
 - Resolusi : 640 x 480

3.5.2 Perangkat Lunak

Perangkat Lunak yang dibutuhkan pada proyek akhir ini adalah.

1. PyCharm

Pycharm adalah aplikasi yang digunakan untuk membuat dan menjalankan pemrograman python. Pycharm sangat mudah digunakan untuk pemula, mudah dalam pemasangan *library*.

2. Xampp

Xampp adalah sebuah software web server apache yang didalamnya sudah tersedia database server mysql dan support php programming. Xampp merupakan software yang mudah digunakan, gratis dan mendukung instalasi di windows.

4. Hasil Simulasi dan Pengujian Sistem Absensi

4.1 Skenario Pengujian Sistem Absensi Kehadiran

Pada bab ini akan dilakukan skenario pengujian, diharapkan dapat memastikan apakah sistem sudah sesuai dengan tujuan awal dari pembuatan sistem absensi kehadiran ini. Serta memastikan apakah semua fungsionalitas yang terdapat pada sistem ini telah berfungsi sesuai yang di harapkan. Dengan menggunakan parameter-parameter yang sudah di jelaskan di bab sebelumnya, yaitu.

1. Jarak
2. Sudut Wajah
3. Aksesoris

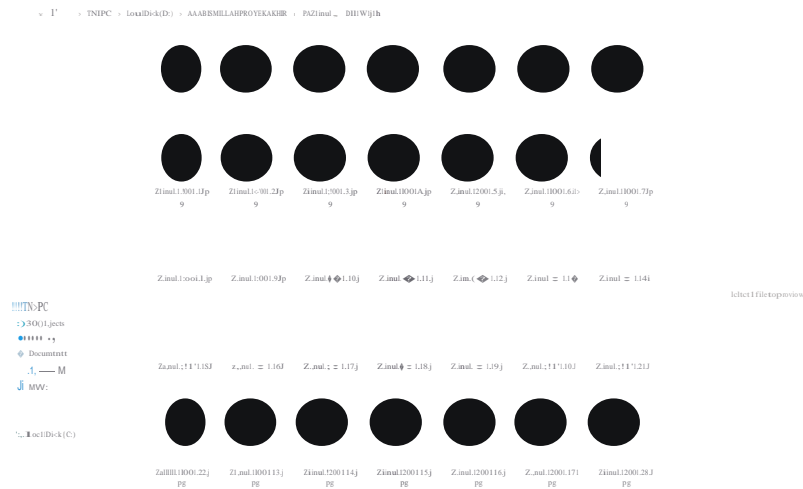
4.2 Data Pengujian Sistem Absensi Kehadiran

Pada tahap kali ini adalah melakukan pendaftaran sebagai data untuk melakukan pengujian. Pendaftaran dilakukan di dalam ruangan dengan cahaya yang cukup. Disini kita mendaftarkan tiga mahasiswa yaitu, zainul muarifin, rodan hilmi dawwas, dan risky aditya. Berikut pengujian pendaftaran wajah.



Gambar 4.1 Proses Pendaftaran

Pada Gambar 4.1 saat melakukan pendaftaran wajah, pertama memasukkan nama dan nim yang akan di daftarkan wajahnya. Lalu tekan rekam wajah untuk memulai perekaman wajah. Data informasi dan data wajah akan direkam lalu disimpan di penyimpanan sistem bisa dilihat pada Gambar 4.2. Selanjutnya tekan latih wajah agar data yang didaftarkan akan digunakan untuk pengenalan wajah, data latih tersebut di simpan dalam format *.yml. gambar bisa dilihat pada Gambar 4.3



Gambar 4.2 Data Gambar Rekam Wajah

Name	Date modified	Size
D Trainer.yrnl	13/07/2020 10:15	YML File 13.900 KB

Gambar 4.3 Data Latih

4.3 Hasil Pengujian Sistem Absensi Kehadiran

Pada tahap pengujian sistem absensi kehadiran ini untuk mengetahui tingkat keberhasilan sistem absensi kehadiran ini berdasarkan jarak dan cahaya saat melakukan absensi.

4.3.1 Hasil Pengujian Jarak

Pada pengujian jarak ini dilakukan oleh tiga orang, jarak yang digunakan adalah dari jarak 5 cm sampai dengan 100 cm. Pengujian dilakukan di dalam ruangan dengan 10 kali percobaan oleh setiap orang dan dilakukan secara bergantian. Berikut merupakan table dari hasil pengujian jarak bisa dilihat pada Tabel 4.1.

Tabel 4.1 Pengujian berdasarkan jarak dari 5 cm sampai 100 cm

No	Pengujian	Jarak	Nama yang Terdeteksi			Total Terdeteksi	Total Tidak Terdeteksi
			Zainul Muarifin	Rodan Hilmi D.	Rizky Aditya		
1	1	5 cm	√	√	√	Tiga Hadir	-
2	2	10 cm	√	√	√	Tiga Hadir	-
3	3	20 cm	√	√	√	Tiga Hadir	-
4	4	30 cm	√	√	√	Tiga Hadir	-
5	5	40 cm	√	√	√	Tiga Hadir	-
6	6	50 cm	√	√	√	Tiga Hadir	-

7	7	60 cm	√	√	√	Tiga Hadir	-
8	8	70 cm	√	√	√	Tiga Hadir	-
9	9	80 cm	√	√	√	Tiga Hadir	-
10	10	100 cm	√	√	√	Tiga Hadir	-

4.3.2 Hasil Pengujian Sudut Wajah

Pada pengujian sudut wajah dilakukan oleh tiga orang, jarak yang digunakan adalah 50 cm dan percobaan dilakukan di dalam ruangan. Pengujian ini dilakukan dengan 10 kali percobaan oleh setiap orang dan dilakukan secara bergantian. Berikut merupakan hasil tabel pengujian bisa dilihat pada Tabel 4.2.

Tabel 4.2 Pengujian berdasarkan sudut wajah

No	Peng-ujian	Sudut Wajah	Nama yang Terdeteksi			Total Terdeteksi	Total Tidak Terdeteksi
			Zainul Muarifin	Rodan Hilmi D.	Rizky Aditya		
1	1	0 derajat	√	√	√	Tiga Hadir	-
2	2	5 derajat kekanan	√	√	√	Tiga Hadir	-
3	3	10 derajat kekanan	√	√	√	Tiga Hadir	-
4	4	15 derajat kekanan	√	x	x	Satu Hadir	Dua Tidak Hadir
5	5	5 derajat kekiri	√	√	√	Tiga Hadir	-
6	6	10 derajat kekiri	√	x	x	Satu Hadir	Dua Tidak Hadir
7	7	15 derajat kekiri	√	x	x	Satu Hadir	Dua Tidak Hadir
8	8	10 derajat keatas	√	√	√	Tiga Hadir	-
9	9	10 derajat kebawah	√	√	√	Tiga Hadir	-
10	10	15 derajat kebawah	√	√	x	Dua Hadir	Satu Tidak Hadir

4.3.3 Hasil Pengujian Aksesoris

Pada pengujian aksesoris dilakukan oleh tiga orang, jarak yang digunakan adalah 50 cm dan percobaan dilakukan di dalam ruangan. Pengujian ini dilakukan dengan 10 kali percobaan oleh setiap orang dan dilakukan secara bergantian. Berikut merupakan hasil tabel pengujian bisa dilihat pada Tabel 4.3.

Tabel 4.3 Pengujian berdasarkan menggunakan aksesoris

No	Peng-ujian	Aksesoris	Nama yang Terdeteksi			Total Terdeteksi	Total Tidak Terdeteksi
			Zainul Muarifin	Rodan Hilmi D.	Rizky Aditya		
1	1	Kacamata	√	√	√	Tiga Hadir	-
2	2	Kacamata	√	√	√	Tiga Hadir	-
3	3	Kacamata	√	√	√	Tiga Hadir	-
4	4	Kacamata	√	√	√	Tiga Hadir	-
5	5	Kacamata	√	√	√	Tiga Hadir	-
6	6	Kacamata	√	√	√	Tiga Hadir	-
7	7	Kacamata	√	√	√	Tiga Hadir	-
8	8	Kacamata	√	√	√	Tiga Hadir	-
9	9	Kacamata	√	√	√	Tiga Hadir	-
10	10	Kacamata	√	√	√	Tiga Hadir	-

5. Kesimpulan dan Saran

5.1 Kesimpulan

Berdasarkan hasil perancangan, dan pengujian yang telah dilakukan maka dapat diambil beberapa kesimpulan sebagai berikut :

1. Berdasarkan hasil perancangan dan pengujian sistem absensi kehadiran. Yang dibangun menggunakan bahasa python dan metode pengenalan wajah menggunakan metode LBPH dapat membedakan citra wajah setiap individu sesuai yang diharapkan.
2. Berdasarkan hasil pengujian jarak pada sistem absensi kehadiran. Dapat mengenali wajah sesuai yang di harapkan dengan rata-rata keberhasilan 100%. Tetapi pada jarak 100 cm terkadang sistem tidak mengenali wajah dengan baik.
3. Berdasarkan hasil pengujian sudut wajah pada sistem absensi kehadiran. Jarak yang digunakan adalah 50 cm, sistem dapat mengenali wajah sesuai yang di harapkan dengan rata-rata keberhasilan 76,67%, yang diperoleh dari rata-rata keberhasilan tiap individu.
4. Berdasarkan hasil pengujian aksesoris pada sistem absensi kehadiran. Aksesoris yang digunakan adalah kacamata, hasil yang didapat untuk mengenali wajah rata-rata keberhasilan 100%, yang diperoleh dari pengenalan menggunakan aksesoris saja tanpa melakukan pengukuran jarak dan sudut wajah.
5. Berdasarkan hasil dari semua pengujian. Sistem absensi kehadiran dapat mengenali wajah dengan rata-rata keberhasilan 92%, yang di dapat dari hasil rata-rata tiap pengujian.

5.2 Saran

Berdasarkan hasil kesimpulan proyek akhir ini, dapat disampaikan beberapa saran untuk pengembangan selanjutnya yaitu :

1. Meningkatkan Keamanan dari sistem absensi kehadiran tersebut, bisa menambah login untuk mengakses sistem.
2. Meningkatkan akurasi dari pengenalan wajah dengan mengunkana metode yang lebih akurat.

DAFTAR PUSTAKA

- [1]. Fauzan, A. (2018). *PERANCANGAN SISTEM DETEKSI WAJAH UNTUK PRESENSI KEHADIRAN MENGGUNAKAN METODE LBPH (Local Binary Pattern Histogram) BERBASIS ANDROID*. Bandung: Telkom University.
- [2]. Jatikusumo, D. (2016). *Panduan Pengolahan Citra*. Jakarta: Universitas Mercu Buana.
- [3]. Koswara, D., & Kurniawan, H. (2016). *PERANCANGAN PENGENALAN POLA WAJAH UNTUK APLIKASI ABSENSI DENGAN METODE EIGENFACE*. Bandung: Media Informatika.
- [4]. Munir, R. (2004). *Pengolahan Citra Diital dengan Pendekatan Algoritmik*. Bandung: Informatika.
- [5]. Ranjan, H. (2019, November 3). <https://github.com/HimanshuRanjan/Face-Recognition-Based-Attendance-System>. Retrieved from <https://github.com>.
- [6]. Shashankch292. (2018, Agustus 13). <https://github.com/shashankch292/Face-Detection-Recognition-Based-Attendance-System>. Retrieved from <https://github.com>.
- [7]. Simaremare, H., & Kurniawan, A. (2016). *Perbandingan Akurasi Pengenalan Wajah Menggunakan Metode LBPH dan Eigenface dalam Mengenali Tiga Wajah Sekaligus secara Real-Time*. Riau: Jurnal Sains, Teknologi dan Industri.
- [8]. SmartCisitu. (2018). *apa-itu-python-python-adalah*. Retrieved Juni 28, 2020, from <https://www.smartcisitu.com/apa-itu-python-python-adalah/>
- [9]. Subkhi, M. B. (2018). *SISTEM PENGENALAN WAJAH UNTUK PRESENSI KULIAH DENGAN METODE EIGENFACE PCA (PRINCIPAL COMPONENT ANALYSIS) DAN CITY BLOCK*. Kediri: Simki_Techsain.
- [10]. Susanto, R., Ananta, A., Susanto, A., & Trianto, M. (2009). *SISTEM ABSENSI BERBASIS RFID*. Jakarta: Jurnal Teknik Komputer.
- [11]. Syahrizal, M. R., Harianto, & Triwidyastuti, Y. (2017). *PERANCANG BANGUN PEMINJAMAN LOKER MENGGUNAKAN RFID DAN METODE FACE RECOGNITION*. Surabaya: JCONES.
- [12]. Wiryadinata, R., Sagita, R., & Priswanto. (2016). *PENGENALAN WAJAH PADA SISTEM PRESENSI MENGGUNAKAN METODE DYNAMIC TIMES WRAPPING, PRINCIPAL COMPONENT ANALYSIS DAN GABOR WAVELET*. Purwokerto: Dinamika Rekayasa.
- [13]. Zulkifli. (2016). *Teknik Pengolahan Citra Menggunakan Metode Kecerahan Citra Kontras Dan Penajaman Citra Dalam Menghasilkan Kualitas Gambar*. Aceh: Universitas Almuslim.