Jurnal Teknik Elektro, Vol. 13, No. 1, Maret 2020, 25-31
ISSN 1411-870X
DOI: 10.9744/jte.11.1.1-6

INTEGRASI SMART DOOR LOCK DENGAN FACE RECOGNITION BERBASIS RASPBERRY PI 3 DILENGKAPI FITUR GOOGLE ASSISTANT

Ivan Surya Hutomo, Handy Wicaksono
Program Studi Teknik Elektro,Universitas Kristen Petra
Jl.Siwalankerto 121-131, Surabaya 60236, Indonesia
E-Mail: hutomoivan@gmail.com ; handy@ petra.ac.id ;

Abstrak **- Penelitian ini bertujuan mengintegrasikan smart door dengan pengenalan wajah dan Google Assistant. Apabila wajah dikenali maka smart door akan melakukan unlock, apabila wajah tidak dikenali maka smart door tidak melakukan *unlock*.Sedangkan Google Assistant dapat digunakan untuk memantau dan mengontrol smart door menggunakan smartphone atau perangkat lain yang menggunakan Google Assistant.**

Sistem smart door lock ini akan menggunakan Raspberry Pi 3 sebagai microcontroller utama dan menggunakan servo sebagai aktuator pengunci. Program akan menggunakan platform Node-RED, Blynk dan MQTT yang sangat membantu dalam pengembangan Internet of Things. Keseluruhan program akan ditulis menggunakan bahasa Python yang umum digunakan pada perangkat Raspberry Pi 3 yang menggunakan OS Raspbian. Pengenalan wajah akan menggunakan Haar Cascade dan Local Binary Pattern Histogram. Integrasi dengan Google Assistant akan menggunakan dialogflow dan firebase.

Pengujian menunjukkan bahwa integrasi face recognition dengan smart door berhasil, apabila wajah dikenali dengan average confidence lebih dari 60% maka smart door akan melakukan *unlock*. Jika wajah tidak dikenali, notifikasi email berisi gambar wajah berhasil dikirimkan ke pemilik rumah. Google Assistant juga berhasil diintegrasikan untuk memantau dan mengontrol *smart door* dengan tingkat keberhasilan mencapai 92.8%.

Kata kunci**: Face Recognition, Google Assistant, Digital Assistant, Smart Door Lock, Raspberry Pi 3..**

I. PENDAHULUAN

Penerapan IoT untuk smarthome kini juga berkonsentrasi pada sistem keamanan dan aksesibilitas suatu rumah. Sebelumnya untuk keamanan rumah, hanya digunakan pintu yang menggunakan kunci mekanik. Kekurangan dari sistem tersebut adalah pintu harus dibuka menggunakan kunci aslinya. Sistem ini dinilai kurang aman dan efisien karena sistem mekanik dari pintu dapat dimodifikasi untuk membuka paksa pintu tersebut, selain itu manusia juga harus membawa kunci fisik untuk membuka pintu.

Selanjutnya pengembangan *smartdoor* mengarah pada penambahan fitur autentikasi elektronis serperti NFC dan sebagainya. Autentifikasi elektronis selanjutnya berkembang menjadi autentifikasi biometrik. Metode biometric merupakan penggunaan ciri – ciri biologis makhluk hidup yang unik dan tidak identik satu dengan yang lain untuk kepentingan keamanan. Metode biometric yang banyak diterapkan di smartdoor yaitu fingerprint dan voice recognition.

Pengintegrasian Face Recognition dengan smart door dinilai lebih efisien dibandingkan menggunakan NFC dan metode biometric lainnya. Dengan menggunakan pengenalan wajah berbasis kamera, pemilik rumah tidak perlu melakukan kontak fisik untuk membuka pintu karena kunci pintu akan terbuka otomatis ketika wajah dengan otoritas tertentu terdeteksi. Penggunaan face recognition merupakan teknologi keamanan yang mutakhir karena teknologi ini akan bertindak seperti manusia dalam mengenali wajah tertentu sehingga dapat dikatakan teknologi face recognition merupakan teknologi yang semakin mendekati kecerdasan manusia [1].

Penelitian sebelumnya dengan judul "Sistem Kendali Akses Ruangan menggunakan Citra Pengenalan Wajah" oleh Frans Rotinsulu belum mengintegrasikan face recognition pada suatu aktuator dan juga masih menggunakan laptop dan webcam sebagai interfacenya, oleh karena itu melalui penelitian ini penulis ingin menerapkan sistem face detection haar cascade dan Local Binary Pattern Histogram pada suatu aktuator untuk membuka dan mengunci smartdoor.

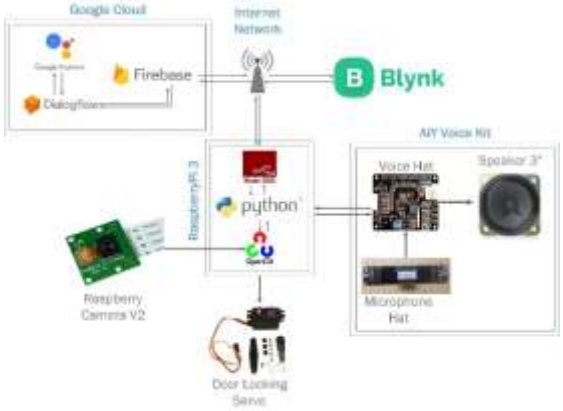
Penelitian sebelumnya juga belum mengintegrasikan smartdoor dengan user experience yang baik dan mutakhir serta belum mengintegrasikan smartdoor dengan sistem notifikasi yang baik. Oleh karena itu selain mengintegrasikan smartdoor dengan pengenalan wajah, pada penelitian ini penulis juga merancang dan mengintegrasikan smartdoor dengan natural user interface dimana nantinya pengguna dapat berkomunikasi dengan pintu seperti berkomunikasi dengan manusia lainnya. Penulis juga akan merancang sistem notifikasi yang baik untuk sistem smartdoor. Diharapkan dengan penggunaan Natural Digital Interface dan sistem notifikasi yang mutakhir, smartdoor dapat meningkatkan pengalaman pengguna (user experience) khususnya mempermudah penggunaannya untuk orang awam.

Natural User Interface akan menggunakan layanan Google Assistant yang dapat membantu memberikan informasi yang lebih jelas mengenai kondisi pintu saat itu termasuk meningkatkan user experience perangkat IoT pengguna. Dengan mengintegrasikan smartdoor lock dengan face recognition dan Google Assistant, diharapkan

pengguna memiliki cara yang lebih efisien, aman, dan interaktif dalam membuka pintu.

II. PERENCANAAN DESAIN ALAT

Berikut merupakan rancangan sistem secara umum :

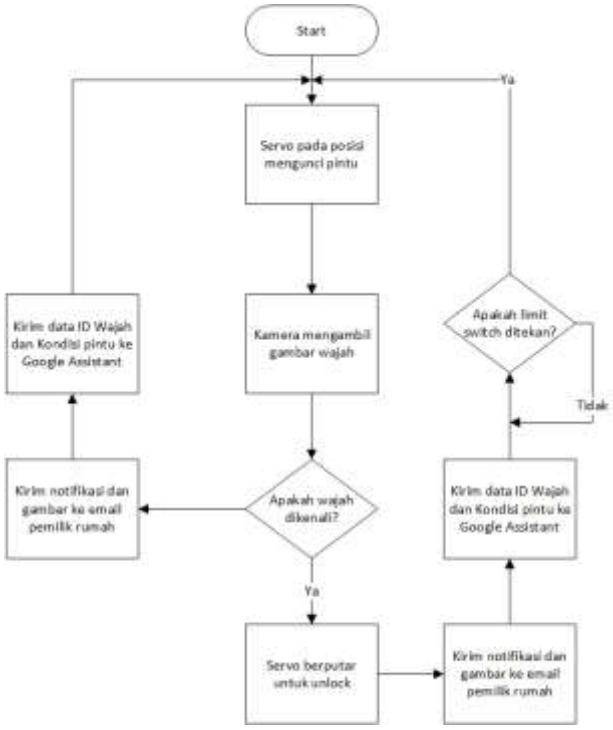


Gambar 1 Rancangan Sistem Keseluruhan

Pada rancangan Gambar 1, Raspberry Pi 3 akan digunakan sebagai mikrokontroller utama untuk mengontrol sensor, aktuator, termasuk proses pemrosesan gambar. Salah satu sensor yang terhubung dengan Raspberry Pi 3 adalah Raspberry Pi Camera V2 sebagai sensor untuk mengambil gambar. Gambar yang diambil akan dideteksi apakah gambar tersebut berisi wajah atau tidak dengan algoritma haar cascade. Apabila terdapat wajah yang terdeteksi, maka proses akan berlanjut ke algoritma Local Binary Pattern Histogram untuk mengenali wajah pada gambar tersebut. Keseluruhan algoritma ini menggunakan bahasa Python dengan library Open Computer Vision (OpenCV).

Setelah langkah image processing dilakukan, maka data yang diperoleh dari proses tersebut dilempar ke platform node-red, node-red selanjutnya akan mengirimkan data pemrosesan ke motor servo sesuai dengan ketentuan wajah dikenali atau tidak. Node-RED juga akan mengirim data ke firebase dan aplikasi Blynk pada smartphone agar pemilik rumah memperoleh notifikasi dari perangkat smart door lock. Dialogflow selanjutnya akan mengambil data dari firebase dan meneruskan ke Google Assistant. Dengan menggunakan dialogflow, pengguna dapat melakukan training phrase yang digunakan untuk melakukan trigger pada Google Assistant sehingga Google Assistant dapat mengetahui kondisi pintu saat itu.

Raspberry Pi 3 juga bertugas mengontrol speaker dan microphone melalui AIY Voice Kit. AIY Voice Kit akan bekerja sebagai perangkat Google Home Assistant di rumah yang juga terintegrasi dengan Google Assistant smartphone.



Gambar 2 Flowchart Cara Kerja *Smartdoor*

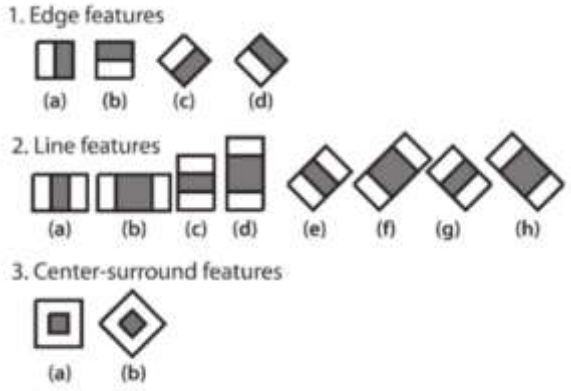
Berdasarkan Gambar 2, ketika sistem pertama kali dimulai maka servo akan berada pada keadaan lock, selanjutnya kamera akan mengambil gambar wajah yang berada di depan pintu pada saat itu dan melakukan pencocokan dengan file training pengenalan wajah. Apabila wajah dikenali maka servo akan berputar untuk melakukan unlock dan kemudian sistem akan mengirimkan notifikasi dan email ke pemilik rumah sekaligus mengirim informasi ID wajah dan kondisi pintu ke Google Assistant. Apabila wajah tidak dikenali, maka servo tidak akan berputar untuk melakukan unlock namun notifikasi tetap dikirimkan ke pemilik rumah. Setelah servo dalam keadaan unlock, maka sistem akan menunggu kondisi limit switch, apabila limit switch ditekan maka servo akan berputar ke kondisi lock.

A. *Haar Cascade dan Local Binary Pattern Histogram sebagai Algoritma Deteksi danPengenalan Wajah*

Untuk proses pendeteksi wajah digunakan algoritma haar cascade. Secara umum, haar-like feature digunakan dalam mendeteksi objek pada image digital. Istilah Haar menunjukkan suatu fungsi matematika (Hhar Wavelet) yang berbentuk kotak, prinsipnya sama seperti pada fungsi Fourier [2]. Awalnya pengolahan gambar hanya dengan melihat dari nilai RGB setiap pixel, namun metode ini ternyata tidaklah efektif [3]. Viola dan Jones kemudian mengembangkannya sehingga terbentuk Haar-Like feature. Haar-like feature memproses gambar dalam kotak kotak, dimana dalam satu kotak terdapat beberapa pixel. Per kotak itu pun kemudian diproses dan menghasilkan perbedaan nilai yang menandakan daerah gelap dan terang. Nilai-nilai inilah yang nantinya dijadikan dasar dalam pemrosesan gambar. Cara menghitung nilai dari fitur ini adalah dengan mengurangkan nilai piksel pada area putih dengan piksel pada area hitam.

Untuk gambar bergerak seperti video, proses ini dilakukan secara diskrit dengan mencuplik video pada *frame rate* tertentu. Macam-macam variasi Haar-like

feature [4] ditunjukkan pada Gambar 3 sebagai berikut:



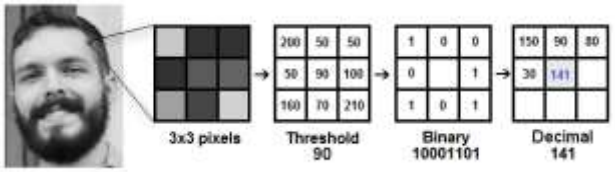
Gambar 3 Diagram Rancangan Hardware *Smartdoor*

Apabila pada gambar *realtime* yang telah dikonversikan menjadi *grayscale* ditemukan *pattern – pattern* seperti pada gambar di atas maka dapat dipastikan bahwa pada gambar tersebut terdapat objek.

Pengenalan wajah merupakan proses lanjutan dari proses pendeteksian wajah. Pendeteksian wajah bisa melalui foto maupun video. Dengan memanfaatkan hasil training dari Haar Cascade, hasil dari proses ini dikombinasikan dengan proses Image Matching dengan algoritma Local Binary Pattern Histogram. Dengan metode ini, foto yang sudah di-learning akan dicocokkan dengan hasil deteksi dari streaming kamera dimana pada streaming nantinya beberapa gambar dalam database kemudian dicocokkan dengan memanfaatkan nilai histogram yang telah diekstraksi dari gambar dengan memanfaatkan persamaan Local Binary Pattern Histogram.

Karakteristik utama dari pengenalan wajah menggunakan metode ini adalah komposisi *microtexture-pattern* yaitu suatu operator nonparametrik yang menggambarkan tata ruang lokal citra. LBPH didefinisikan sebagai perbandingan nilai biner piksel pada pusat citra dengan 8 nilai piksel di sekelilingnya. Misal pada sebuah citra berukuran 3x3, nilai biner pada pusat citra dibandingkan dengan nilai sekelilingnya. Dengan cara mengurangkan nilai piksel pada pusat citra dengan nilai piksel di sekelilingnya, jika hasilnya lebih atau sama dengan 0 maka diberi nilai 1 dan jika hasilnya kurang dari 0 maka diberi nilai 0. Setelah itu, disusun 8 nilai biner searah jarum jam atau sebaliknya dan diubah 8 bit biner ke dalam nilai desimal untuk menggantikan nilai piksel pada pusat citra. Rumus mencari tata ruang biner dan nilai LBPH adalah sebagai berikut [1]

Setelah menyusun binerisasi searah jarum jam, maka apabila salah satu kotak biner threshold bernilai 1 maka masukkan nilai biner sesuai pangkatnya, namun bila 0 maka hasilnya juga sama dengan 0. Terakhir tambahkan nilai LBP .



Gambar 4 Diagram Rancangan Hardware *Smartdoor*

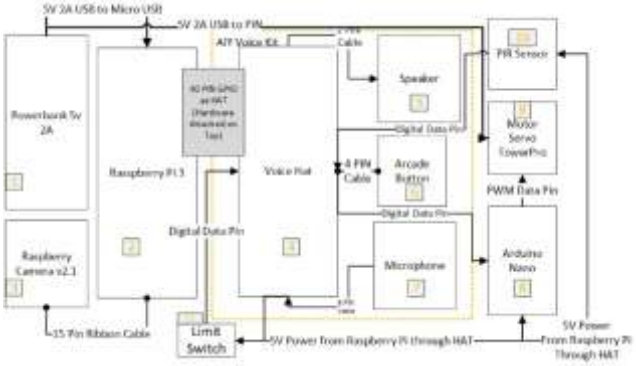
Untuk mencocokkan wajah pemilik digunakan sebuah persamaan untuk mendapatkan pendekatan nilai histogramnya yang nanti digunakan sebagai nilai prediksi untuk mengindentifikasikan pemilik wajah tersebut. Berikut adalah persamaan untuk mencari pendekatan nilai histogram.

Jadi output algoritma adalah (D) dari gambar dengan histogram terdekat. Algoritma juga harus mengembalikan jarak yang dihitung, yang dapat digunakan sebagai pengukuran nilai confidence. Nilai threshold dan confidence selanjutnya dapat digunakan secara otomatis untuk memperkirakan apakah algoritma telah mengenali gambar dengan benar. Apabila nilai confidence lebih rendah daripada threshold, maka algoritma telah berhasil mengenali gambar.

B. *Rancang Bangun Hardware*

Dalam pembuatan penelitian ini, digunakan Raspberry Pi 3 sebagai microcontroller utama. Pada penelitian sebelumnya [5] masih menggunakan laptop sebagai pemroses utama citra wajah yang dinilai masih kurang portable dan kurang fleksibel apabila dioperasikan secara tetap. Dalam penelitian itu, juga disarankan untuk mengganti platform laptop dengan sebuah microcontroller yang lebih kecil yang dapat diterapkan pada berbagai macam sistem yaitu Raspberry Pi.

Penulis menggunakan Raspberry Pi 3 B karena dinilai sudah cukup untuk melakukan proses pengenalan wajah, selain itu dalam sistem ini Raspberry Pi 3 juga kompatibel dengan AIY Voice Kit yang disediakan oleh Google. Selain itu Raspberry Pi 3 memiliki kemudahan GUI dengan sistem operasi Raspbian yang dapat diekspansi dengan berbagai platform seperti OpenCV, Node-Red, MQTT,dan sebagainya. Raspberry Pi 3 juga tidak membutuhkan daya yang besar untuk beroperasi.



Gambar 5 Diagram Rancangan Hardware *Smartdoor*

Pada Gambar 5 Rasberry Pi 3 akan ditenagai oleh Powerbank 5v 2A (Nomor 1) dengan menggunakan interface micro usb. Penggunaan powerbank ini juga dapat digantikan dengan adaptop 5v 2A yang umum digunakan untuk melakukan pengisian baterai smartphone. Powerbank juga akan memberikan supply daya ke motor servo, sehingga motor servo (Nomor 9) dapat memperoleh torsi yang cukup untuk memutar deadbolt.

Raspberry Pi Camera v2 (Nomor 3) digunakan sebagai sensor utama untuk melakukan pengenalan wajah, dimana perangkat ini terhubung dengan Raspberry Pi 3 menggunakan ribbon cable 15 pin. Voice Hat (Nomor 4) akan dipasang di atas Raspberry Pi 3 yang dedesain sebagai Hardware Attached on Top (HAT). Selanjutnya microphone (Nomor 7), speaker (Nomor 5), dan arcade button (Nomor 6) akan terhubung ke Raspberry Pi 3 melalui HAT. Dengan adanya HAT yang terpasang pada Raspberry Pi 3, GPIO (General Purpose Input Output) pada Raspberry Pi 3 tidak bisa diakses secara langsung, sehingga untuk mengakses

GPIO dapat dilakukan melalui Voice Hat. GPIO Voice Hat terhubung dengan arduino nano (Nomor 8) dan PIR Sensor (Nomor 10).

Penggunaan Arduino Nano (Nomor 8) pada perancangan penelitian ini bertujuan untuk mengatasi jitter pada motor servo (Nomor 9). Jitter adalah Pergerakan servo yang tidak stabil diakibatkan oleh sinyal PWM yang tidak bagus. Sebelumnya penulis mengubungkan langsung motor servo dengan Raspberry Pi 3, namun pada saat dilakukan pengujian, pergerakan motor servo tidak stabil khususnya saat Raspberry Pi 3 melakukan banyak proses multitasking. Oleh karena itu dalam perancangan penelitian ini, penulis menggunakan Arduino Nano sebagai generator signal PWM untuk motor servo. Raspberry Pi 3 hanya akan melakukan triggering secara digital pada arduino nano dan selanjutnya arduino nano yang akan mengeluarkan sinyal PWM untuk merubah sudut servo.

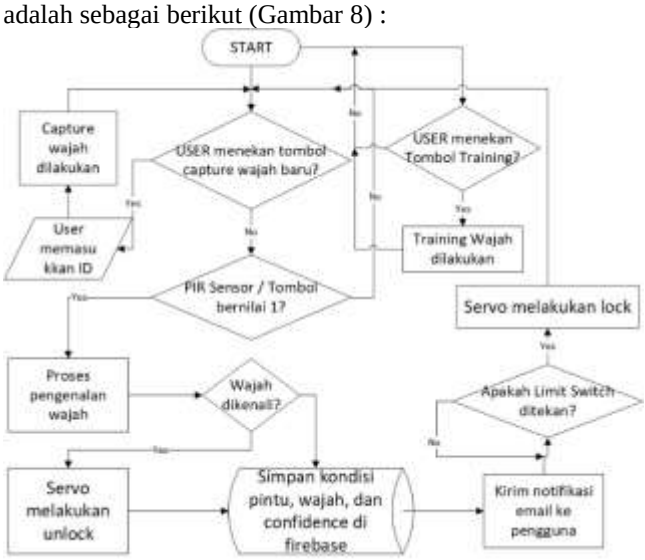
Penggunaan PIR Sensor (Nomor 10) pada perancangan penelitian ini berfungsi untuk mendeteksi apakah ada orang atau tidak di depan pintu. Apabila di depan pintu terdapat orang, maka library OpenCV baru memulai image processing, namun apabila tidak terdapat orang di depan pintu maka OpenCV hanya akan menyiapkan kamera. Dengan adanya PIR Sensor, image processing hanya akan dilakukan apabila dibutuhkan saja sehingga lebih menghemat penggunaan resource memory Raspberry Pi dan tidak membuat Raspberry Pi mengalami kenaikan suhu dengan cepat.

Limit Switch (Nomor 11) akan dipasang pada kusen pintu untuk mengetahui apakah pintu dalam kondisi tertutup atau tidak. Apabila pintu sudah tertutup kembali maka servo akan memutar kunci untuk mengunci pintu, sedangkan apabila pintu belum tertutup, maka servo tidak akan memutar untuk mengunci pintu. Berikut merupakan gambaran maker *smart door* yang sudah dirakit (Gambar 6) :

Gambar 6 Maket *Smart Door* yang Telah Dirakit

C. *Rancang Bangun Software*

Perancangan penelitian ini akan membuat face recognition smart door dengan cara kerja secara umum adalah sebagai berikut (Gambar 8) :



Gambar 7 *Flowchart* fungsional alat secara umum

Ketika smart door pertama kali diinisiasi, jika user tidak menekan tombol untuk menambah data wajah, maka sistem akan mendeteksi apakah ada orang atau tidak melalui PIR Sensor. Apabila terdapat orang, maka kamera akan membaca wajah dan OpenCV akan melakukan image processing untuk mengenali wajah. Apabila wajah dikenali maka servo akan bergerak untuk melakukan unlock. Setelah proses unlock dilakukan, maka data – data seperti status pintu dan data pengenalan wajah akan diupload ke realtime database di Firebase sehingga Google Assistant dapat melakukan pembacaan melalui Firebase.

Setelah keseluruhan data disimpan di Firebase, sistem akan mengirimkan email ke pemilik rumah. Selanjutnya orang yang mengakses pintu akan membuka pintu dan untuk mendeteksi apakah pintu sudah ditutup kembali oleh orang tersebut maka digunakanlah limit switch yang diletakkan di kusen pintu. Jika pintu sudah ditutup kembali maka limit switch akan kembali dalam kondisi on sehingga servo akan memutar kunci ke keadaan lock.

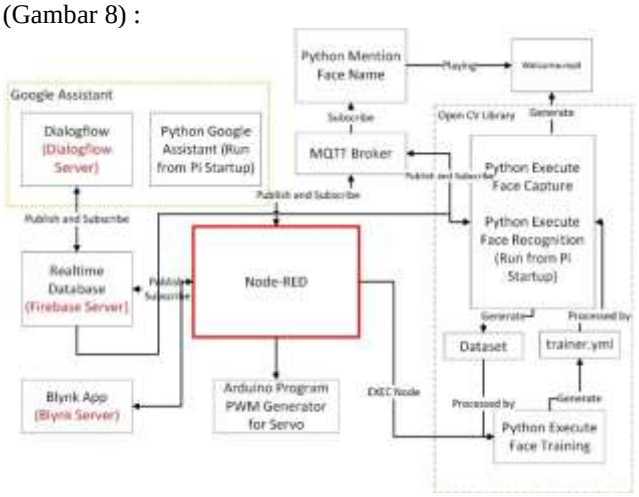
Jika pada proses pengenalan wajah tidak didapatkan hasil yang cocok, maka servo tidak akan memutar kunci ke kondisi unlock, tetapi usaha untuk mengakses pintu tersebut tetap dicatat di Firebase dan email notifikasi tetap dikirimkan ke pemilik rumah bahwa ada orang yang tidak dikenali mencoba mengakses pintu rumah.

Kembali ke bagian awal flowchart, apabila pemilik rumah ingin menambah data wajah baru, maka pemilik rumah akan menekan tombol capture dan kemudian memasukkan nomor yang mewakili ID pemilik wajah, kemudian dengan library OpenCV menggunakan deteksi wajah, Raspberry Pi Camera akan mengambil 30 sampel gambar untuk dijadikan bahan training pengenalan wajah.

Pada flowchart Gambar 7, proses training sebenarnya merupakan proses yang dapat dilakukan secara terpisah, pemilik rumah dapat melakukan training pengenalan wajah kapanpun, training akan menggunakan dataset yang telah di capture sebelumnya. Karena training sudah tidak membutuhkan kamera seperti proses capture dan proses recognition maka prosesnya terpisah dari proses utama.

Berdasarkan rancang bangun fungsional secara umum pada (Gambar 7). Rancang bangun software untuk smart

door lock ini dapat dilihat pada diagram blok berikut (Gambar 8) :



Gambar 8 Diagram Blok Software

Pada Gambar 8, keseluruhan sistem akan terhubung dengan Node-Red yang akan mengintegrasikan keseluruhan sistem. Node-RED sendiri merupakan bahasa pemrograman visual berbasis browser yang dikembangkan berdasarkan 'flow'. Pada rancangan software tersebut, terdapat beberapa platform yang terlibat.

Berikut merupakan penjelasan skema dari Gambar 8, Node-RED merupakan pengatur utama dan penghubung keseluruhan sistem. Node-RED akan berkomunikasi dengan Realtime Database (Firebase Server) dimana Node-RED akan memberikan informasi message yang diterima dari program pengenalan wajah Python melalui MQTT Broker kepada path tertentu di firebase. Firebase akan berkomunikasi dengan Dialogflow yang akan terintegrasi secara langsung dengan layanan Google Assistant pada smartphone pengguna. Komunikasi ini menggunakan fullfillment request dimana kita dapat membuat suatu fungsi sehingga menyebabkan dialogflow dapat mengambil data pada path tertentu di Firebase.

Node-RED juga akan berkomunikasi dengan aplikasi Blynk, komunikasi Node-RED dengan Blynk nantinya berdasarkan triggering dari virtual pin. Apabila Blynk memerintahkan untuk mengunci pintu, maka Blynk akan memberikan message pada topik MQTT tertentu sehingga Node-RED akan mengendalikan servo berdasarkan message dari MQTT.

Program face recognition dijalankan menggunakan Python dan bertukar data dengan Node-RED melalui message pada MQTT broker. Python Face Capture dan Face Recognition merupakan satu kesatuan program karena keduanya membutuhkan library deteksi wajah yang sama yaitu Haars Cascade. Apabila face capture dilakukan makan Python akan menggenerate 30 Gambar dataset untuk satu wajah. Dataset ini kemudian dilakukan proses training menggunakan Local Binary Pattern Histogram untuk dibuatkan pattern histogram pengenalan wajah dimana nantinya setiap wajah akan memiliki pattern histogram yang unik. Hasil training ini akan bernama trainer.yml. File ini akan digunakan pada program pengenalan wajah untuk mengenali wajah. Apabila wajah dikenali misalkan id wajah tersebut adalah 1, maka Python akan berkomunikasi secara langsung dengan Firebase untuk mengambil nama pada index 1 di path tertentu pada Firebase. Nama tersebut nantinya akn digenerate audio penyebutan namanya menggunakan library Google TTS dengan file welcome.mp3.

III. HASIL PENGUJIAN

Pada bab ini akan dilakukan pengujian pada proses pengambilan dataset yaitu jarak kamera dengan resolusi dataset, kemudian pengaruh resolusi yang dihasilkan dengan lama waktu training dan akurasi pengenalan wajah. Selanjutnya akan dilakukakan pengujian parameter pengenalan wajah dalam Haar cascade yaitu ScaleFactor dan MinNeighbor. Setelah mendapatkan nilai ScaleFactor dan MinNeighbor terbaik selanjutnya dilakukan pengujian lama training, jumlah pengambilan dataset, dan juga akurasi dan kecepatan sistem dalam mengenali wajah. Pengujian juga dilakukan untuk menguji integrasi Google Assistant dengan smart door menggunakan Dialogflow.Berikut adalah pengujian yang dilakukan pada penelitian ini :

A. *Pengujian Jarak Kamera terhadap Resolusi, Waktu Training, Akurasi, dan Waktu Pengenalan Wajah*

Berdasarkan pengujian, jarak kamera memiliki pengaruh terhadap resolusi *dataset* yang dihasilkan. Resolusi dataset yang dihasilkan selanjutnya juga memiliki pengaruh terhdapap waktu training, akurasi, dan waktu pengenalan wajah.

Tabel 1 Perbandingan jarak kamera, resolusi, waktu training, akurasi, dan waktu pengenalan wajah

Jarak Wajah dengan Kamera	Resolusi Dataset	Waktu Training	Akurasi	Waktu Pengenalan Wajah
45 cm	191x191 px	3.65 s	42.20%	3.031 s
30 cm	320x320 px	9.45 s	55.30%	2.49 s

Berdasarkan tabel diatas semakin dekat jarak wajah dengan kamera semakin tinggi resolusi *dataset* yang dihasilkan. Semakin tinggi resolusi *dataset* maka akan semakin tinggi waktu training yang diperlukan namun akurasi pengenalan wajah meningkat hinnga 55.03% dan waktu pengenalan wajah semakin cepat menjadi 2.49 s apabila dibandingkan dengan resolusi yang lebih kecil. Resolusi yang tinggi menyebabkan LBPH dapat memetakan histogram dengan semakin detail pada wajah dataset sehingga pada akhirnya mampu meningkatkan akurasi pengenalan wajah. Dari pengujian diatas maka digunakan jarak pengenalan wajah adalah 30 cm.

B. *Pengujian Parameter MinNeighbor dan ScaleFactor terhadap Akurasi dan Waktu Pengenalan Wajah*

Pengujian dilakukan untuk mencari parameter *ScaleFactor* dan *MinNeighbor* yang paling optimal berdasarkan akurasi terbaik dan waktu pengenalan wajah paling cepat. Rata – rata akurasi dan waktu diambil dari 10x percobaan pengenalan wajah. Pada pengujian pertama, nilai *MinNeighbor* dibuat tetap dengan nilai 3 dan nilai *ScaleFactor* diubah – ubah. Selanjutnya setelah ditemukan nilai *ScaleFactor* yang paling optimal, nilai *ScaleFactor* dibuat tetap dan nilai *MinNeighbor* diubah – ubah. Berdasarkan pengujian parameter yang paling optimal untuk *ScaleFactor* adalah 1.5 (Tabel 2) dan untuk *MinNeighbor* adalah 2 (Tabel 3).

Pada nilai ScaleFactor 1.1 , rata – rata waktu memiliki angka paling buruk yaitu 6.1227 detik (Tabel 2), hal ini dapat dianalisa karena proses reduksi untuk menyamai

dataset tidak cukup besar sehingga reduksi terjadi berulangkali dan membutuhkan waktu yang lebih lama untuk menyamai dataset. Pada peningkatan tiap nilai scaleFactor, waktu yang dihasilkan semakin cepat dan pada nilai 1.5 waktu berada pada nilai terbaik. Pada nilai ScaleFactor berikut proses pengenalan wajah mengalami peningkatan waktu karena nilai reduksi yang dihasilkan terlalu besar sehingga tidak mampu menyamai dataset.

Tabel 2 Perbandingan nilai ScaleFactor dengan Akurasi dan Waktu Pengenalan Wajah

Scale Factor	MinNeighbor	Rata - rata Akurasi (%)	Rata - rata Waktu (s)
1	3	Error	
1.1	3	46.93	6.1227
1.2	3	41.76	3.973
1.3	3	50.97	3.111
1.4	3	44.08	2.423
1.5	3	58.39	2.059
1.6	3	31.537	2.204
1.7	3	55.52	2.38
1.8	3	26.29	5.458
1.9	3	Error	

Berdasarkan pengujian (Tabel 3) nilai MinNeighbor terbaik adalah 2, dimana akurasi pengenalan wajah rata – rata dapat mencapai 62.04% dengan waktu rata – rata 2.093 detik. MinNeighbor pada percobaan penelitian ini tidak terlalu mempengaruhi akurasi dan kecepatan pengenalan wajah. Hal ini dapat dianalisa karena wajah pada percobaan ini sudah dekat dengan kamera yaitu 45 cm dan wajah sudah memenuhi keseluruhan frame kamera sehingga kemungkinan terjadi false positive sangat kecil dan pada akhirnya tidak mempengaruhi akurasi. Walaupun demikian pada nilai MinNeighbor kecil waktu untuk mengenali wajah semakin cepat walaupun tidak berubah signifikan (Tabel 4.5), hal ini dikarenakan dengan MinNeighbor yang semakin kecil maka sistem akan semakin sensitif dalam mengenali wajah (meningkatkan kemungkinan false positive) sehingga waktu untuk mengenali wajah semakin cepat.

Tabel 3 Perbandingan nilai MinNeighbor dengan Akurasi dan Waktu Pengenalan Wajah

Scale Factor	MinNeighbor	Rata - rata Akurasi (%)	Rata - rata Waktu (s)
1.5	1	42.032	2.0484
1.5	2	62.04	2.093
1.5	3	59.06	2.081
1.5	4	51.71	2.118
1.5	5	47.827	2.0874
1.5	6	57.12	2.105
1.5	7	58.11	2.024
1.5	8	50.71	2.079
1.5	9	46.18	2.171
1.5	10	58.47	6.037

C. *Pengujian untuk Mencari Average Confidence sebagai Threshold untuk Unlock Smart Door*

Pengujian ini bertujuan untuk mengetahui kemampuan sistem dalam membedakan wajah yang dikenali dan wajah yang tidak dikenali. Pengujian ini perlu dilakukan karena memiliki pengaruh dalam otorisasi untuk membuka pintu. Jika wajah yang tidak terdaftar dalam dataset dikenali sebagai wajah orang lain yang terdaftar dalam dataset maka kunci pintu dapat terbuka. Dalam pengujian ini, dataset yang akan didaftarkan hanya terdiri dari 3 orang responden, sementara 1 responden lainnya tidak akan didaftarkan dalam dataset dan seharusnya oleh sistem tidak akan diberikan otoritas untuk membuka kunci pintu (Tidak dikenali).

Tabel 4 Tabel Pengujian dengan Dataset Unknown

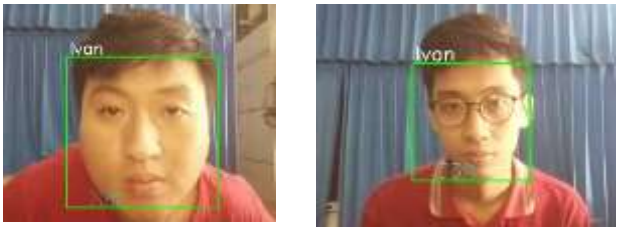
Isi Dataset	Nama	Rata" Akurasi	Rata" waktu
Ivan, Michael, Gavrielle	Ivan	65.34	3.876
	Michael	73.63	3.972
	Gavriel	67.48	4.047
	Jischak	44.18	5.252
Ivan, Michael	Ivan	63.39	3.474
	Michael	70.711	3.477
	Gavriel	55.34	3.496
	Jischak	50.61	3.388
Ivan	Ivan	62.59	2.642
	Michael	50.78	3.171
	Gavriel	53.41	3.019
	Jischak	51.57	2.983
Rata - rata false confidence		50.98166667	
Rata - rata true confidence		67.19016667	

Dari hasil pengujian pada tabel diatas, didapati bahwa hasil pengenalan wajah terhadap wajah yang tidak ada dalam dataset tidak pernah unknown melainkan selalu dikenali sebgai wajah yang ada dalam dataset. Pada pengujian pertama Jischak tidak dicapture dalam dataset, seharusnya sistem tidak mengenali wajah tersebut, namun wajah Jischak dikenali sebagai Michael. Walaupun demikian rata – rata confidence yang dihasilkan tidak sebaik wajah asli Michael yaitu 44.18% (Jischak) berbanding 73.63% (Michael). Pada pengujian kedua, wajah Gavriel dan Jischak tidak dicapture dalam dataset namun demikian keduanya dikenali sebagai Michael dengan false confidence masing – masing 55.34% dan 50.61% sedangkan wajah asli Michael memiliki true confidence sebesar 70.711%. Pada pengujian ketiga hanya wajah Ivan yang dicapture dalam dataset, namun demikian wajah Michael, Gavriel, dan Jischak dikenali sebagai wajah Ivan dengan false confidence masing – masing 50.78%, 53.41%, dan 51.57% sedangkan wajah Ivan memiliki true confidence sebesar 62.59%.

Melalui ketiga pengujian tersebut, masing – masing true confidence dan false confidence dihitung rata – ratanya dan dihasilkan rata – rata false confidence sebesar 50.981% dan true confidence 67.176%. Dari hasil rata – rata tersebut, maka untuk mencegah wajah tidak dikenal membuka kunci pintu maka rata - rata confidence untuk membuka pintu harus di atas 50.981%. Namun demikian agar wajah yang berwenang dapat mengakses pintu maka rata – rata

confidence harus di bawah 67.176%. Dengan memperhatikan rata – rata false confidence tertinggi adalah 55.34% dan rata – rata true confidence terendah 62.59% maka rata – rata confidence untuk membuka pintu dapat ditentukan 60%. Dengan menggunakan nilai tersebut, maka wajah dengan confidence lebih besar dari 60% dianggap dikenali dan dapat membuka pintu, sementara itu wajah dengan confidence kurang dari 60% walaupun dikenali sebagai wajah dalam dataset (false confidence) tidak memiliki kewenangan membuka kunci pintu dan dianggap unknown dalam sistem.

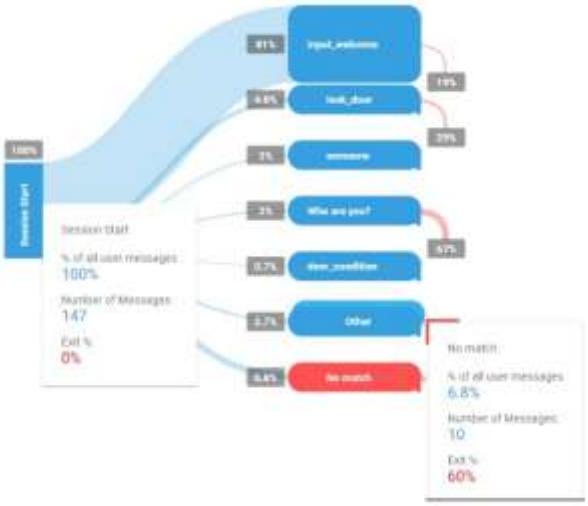
Berikut ini merupakan gambar dari pengujian wajah known dan unknown dataset dimana wajah yang seharusnya tidak diketahui dikenali sebagai wajah orang dalam dataset dengan *average confidence* lebih rendah. Gambar 9 merupakan gambar yang menunjukkan wajah asli yang dikenali sebagai Ivan dengan *average confidence* hingga 70% (Gambar 9 kiri). Sedangkan pada gambar 9 kanan, wajah *unknown dataset* dikenali sebagai wajah Ivan (*known dataset*) dengan *average confidence* lebih rendah hanya 33%.



Gambar 9 - Perbedaan *average confidence* pada wajah *known dataset* (Kiri) dan wajah *unknown dataset* (Kanan)

D. *Pengujian Integrasi Google Assistant*

Pengujian juga dilakukan untuk menguji frekuensi tingkat keberhasilan Google Assistant dalam merespon perintah dari pengguna. Pengujian frekuensi keberhasilan tersebut dapat dilihat pada session flow dari Dialogflow.



Gambar 10 Session Flow

Berdasarkan session flow (Gambar 10),dari 147 percobaan yang dilakukan, tingkat keberhasilan mencapai 93.2%. Kegagalan Google Assistant mengenali pembicaraan pengguna hanya 6.8% yaitu 10 percobaan dari 147 percobaan. Keseluruhan pengujian ini menunjukkan integrasi Dialogflow dengan Firebase dan Node-RED serta akun Google Assistant penulis sudah berhasil.

IV. KESIMPULAN

Dari seluruh hasil perakitan, pembuatan, dan pengujian smart door yang diintegrasikan dengan face recognition dan google assistant, didapatkan kesimpulan sebagai berikut :

1. Face recognition menggunakan deteksi wajah Haar Cascade dan pengenalan wajah LBPH berhasil diintegrasilan pada smart door dengan jarak pengambilan dataset paling efektif 30 cm, MinNeighbor 2, dan ScaleFactor 1.5. Jumlah wajah yang efektif disimpan dalam database adalah 3 wajah, lebih dari 3 wajah sistem mulai mengalami false recognition. Apabila average confidence wajah lebih dari 60% smart door akan melakukan unlock sedangkan apabila kurang dari 60% akan diklasifikasikan sebagai unknown.
2. Pengintegrasian Google Assistant dengan smart door berhasil dengan baik. Google Assistant dapat digunakan untuk mengambil informasi, mengendalikan perangkat smart door dengan memanfaatkan Dialogflow dan realtime database dari Firebase. Berdasarkan pengujian, dapat disimpulkan bahwa Dialogflow dan Firebase cukup handal untuk dimanfaatkan dan diintegrasikan dengan smart door dengan tingkat keberhasilan merespon mencapai 93.2%

V. DAFTAR PUSTAKA

[1] M. P. Beham and S. M. M. Roomi, “a Review of Face Recognition Methods,” *Int. J. Pattern Recognit. Artif. Intell.*, vol. 27, no. 04, p. 1356005, 2013.

[2] P. Purwanto, B. Dirgantoro, and A. N. Jati, “Implementasi Face Identification Dan Face Recognition Pada Kamera Pengawas Sebagai Pendeteksi Bahaya,” *eProceedings Eng.*, vol. 2, no. 1, Apr. 2015.

[3] P. Viola and M. Jones, “Rapid object detection using a boosted cascade of simple features,” in *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, vol. 1, p. I-511-I-518.

[4] S.-K. Pavani, D. Delgado, and A. F. Frangi, “Haar-like features with optimally weighted rectangles for rapid object detection,” *Pattern Recognit.*, vol. 43, no. 1, pp. 160–172, Jan. 2010.

[5] F. Rotinsulu, “Sistem Kendali Akses Ruangan menggunakan Citra Pengenalan Wajah,” Petra Christian University, 2015.