

Sistem Presensi Mahasiswa Berbasis *Realtime* Kamera Metode Klasifikasi *Haar*

Aldiansyah Famni Saputra¹, Cahyo Darujati²

Jurusan Sistem Komputer, Universitas Narotama Surabaya, Jl. Arief Rachman Hakim 51, 60117, Indonesia

aldiansyah@fik.narotama.ac.id¹, cahyod@fasilkom.narotama.ac.id²,

Diterima: 22 Juli 2020; direvisi: 24 Agustus 2020; disetujui: 12 September 2020

Abstract — At Narotama University, in Surabaya, it has already implemented a presence system by recognizing facial patterns but it is only intended for lecturers and campus staff. Realtime facial recognition is an interesting problem with many conditions that affect it. This journal is intended for the presence of a presence system using face recognition with using the Haar Classification method through ESP32-Cam. For video streaming using the OV2640 2MP camera, the results of facial recognition can be sent directly to the academic website made by utilizing the Internet of Things, and the ESP hardware used for research is quite economical.

Keywords — academic webpage; haar classification; internet of things; presence system.

Abstrak — Di Universitas Narotama Surabaya sudah menerapkan sistem presensi dengan mengenali pola wajah namun hanya diperuntukan bagi dosen dan staf kampus, Pengenalan wajah *realtime* adalah masalah yang menarik dengan banyaknya kondisi yang mempengaruhi. Jurnal ini ditujukan untuk implementasi sistem presensi menggunakan pengenalan wajah dengan metode Klasifikasi *Haar* melalui ESP32-Cam. Untuk *streaming* video menggunakan kamera OV2640 2MP, hasil dari pengenalan wajah bisa langsung dikirim ke *website akademik* buatan dengan memanfaatkan *Internet of Things*, dan *hardware ESP* yang digunakan saat penelitian cukup hemat.

Kata kunci — *internet of things*; klasifikasi *haar*; sistem presensi; *webpage akademik*.

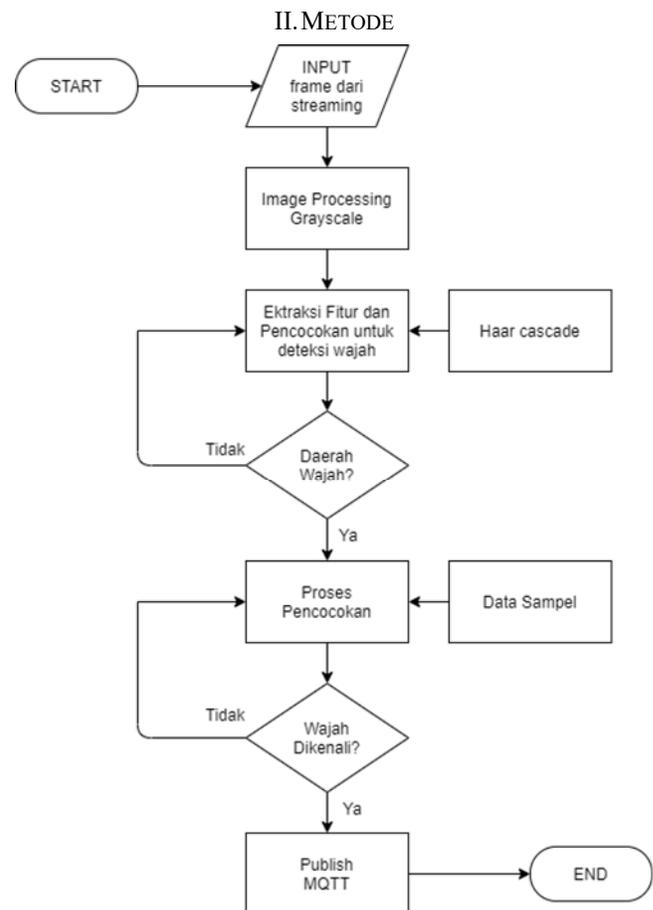
I. PENDAHULUAN

Wajah merupakan bagian dari kepala manusia, pada wajah terdapat dahi, alis, mata, hidung, pipi, mulut, gigi, dan dagu. Wajah terutama digunakan manusia sebagai media peng ekspresian emosi, penampilan dan juga sebagai identitas seseorang, Tidak ada satupun wajah yang serupa mutlak, bahkan wajah pada manusia kembar identik sekalipun masih ada perbedaan.

Di era industri 4.0 ini, teknologi telah berkembang dan akan terus berkembang selama dibutuhkan. Teknologi berkembang pada berbagai macam bidang, salah satu bidangnya adalah sebagai media informasi yang hampir keseluruhan berupa sistem. Sebagai media pelaksana yang mencakup pengambilan, pengolahan, dan eksekusi data yang dapat menghasilkan suatu keluaran untuk mempermudah pekerjaan

manusia[1].

Alat presensi awalnya berupa *konvensional* yaitu dengan media tulis, di era industri 4.0 ini alat presensi mulai berkembang dan memanfaatkan adanya internet dalam memasukan dan menyimpan data[2]. Salah satu sistem presensi melakukan pengenalan pola wajah, yang dimana terdapat bagian-bagian yang cukup unik dan berbeda pada setiap orang. Dengan adanya metode klasifikasi Haar yang digunakan oleh sistem presensi proyek ini, mengenali pola wajah lebih mudah dengan pemberian *ID* pada setiap wajah orang yang sudah dikenali.



Gambar 1. Flowchart Sistem

Gambar 1 merupakan *flowchart* sistem yang menjelaskan cara kerja ESP32-Cam dalam mengenali wajah, mulai dari *input frame*, *image processing*, ekstraksi fitur, pencocokan data sampel, sampai dengan *publish* data berupa nomor id hasil dari wajah yang dapat dikenali lalu data tersebut dikirim melalui *broker mqtt*

A. Image Processing Grayscale

Melakukan pemrosesan citra dengan tujuan mengubah gambar asli ke citra *grayscale* dan dapat mempermudah kinerja sistem dalam pendeteksian letak wajah dengan batas satu warna yaitu pada warna abu-abu[3].

Gambar skala abu-abu dapat menjadi hasil pengukuran intensitas cahaya pada setiap piksel berdasarkan kombinasi frekuensi tertentu (atau panjang gelombang) tertentu, dan dalam kasus seperti itu *monokromatik* tepat ketika hanya satu frekuensi (dalam praktiknya, pita frekuensi sempit) yang ditangkap[4]. Frekuensi pada prinsipnya didapatkan dari mana saja dalam sebuah spektrum elektromagnetik (misalnya pada Inframerah, cahaya tampak, ultraviolet, dll.)[5].

B. Ekstraksi Fitur dan Pencocokan dengan Haar

Ekstraksi fitur merupakan suatu pengambilan fitur dari suatu bentuk yang nantinya nilai yang didapatkan akan dianalisa untuk proses selanjutnya[6]. Ekstraksi fitur dilakukan dengan cara menghitung jumlah titik atau piksel yang ditemui dalam setiap pengecekan[7]. Pengecekan dilakukan dalam berbagai arah koordinat kartesian dari citra digital yang dianalisis, yaitu vertikal, horizontal, diagonal kanan, dan diagonal kiri[8]. Fitur yang didapat dari sebuah citra merupakan ciri khas pembeda dengan citra-citra yang lain[9].

Pencocokan dilakukan dengan memanfaatkan klasifikasi *haar* dalam membedakan suatu frame dan memeriksa apakah suatu wilayah tertentu dari frame tersebut terlihat seperti wajah atau tidak[5]. Jika frame itu berbentuk wajah maka akan langsung membuat kotak disekitar area yang merupakan wajah pada frame tersebut[10].

C. Data Sampel

Data sampel ini diambil pada saat *enroll* wajah yang disimpan didalam *Flash Memory* pada ESP32 untuk membantu sistem mengenali wajah pada saat *streaming* video. Pada tabel I menunjukkan bahwa pada proyek perancangan sistem

presensi berbasis *realtime* kamera, terdapat total lima nama mahasiswa mulai dari mahasiswa dengan nama Elvina, Puji, Abdul, Aldiansyah, dan Nuril, dari kelima nama mahasiswa tersebut nantinya akan dilakukan proses pengujian pada setiap mahasiswa melalui *streaming* video, kelima nama mahasiswa tersebut mempunyai masing-masing 25 data sampel id, untuk mahasiswa dengan nama Elvina mempunyai sampel id dari 51 sampai 75, untuk mahasiswa dengan nama Puji mempunyai sampel id dari 76 sampai 100, untuk mahasiswa dengan nama Abdul mempunyai sampel id dari 26 sampai 50, untuk mahasiswa dengan nama Aldiansyah mempunyai sampel id dari 1 sampai 25, untuk mahasiswa dengan nama Nuril mempunyai sampel id dari 101 sampai 125. jadi dapat disimpulkan bahwa jumlah keseluruhan dari data sampel id untuk kelima nama mahasiswa mencapai total 125 data sampel id dan telah tersimpan pada *Flash Memory* ESP32-Cam.

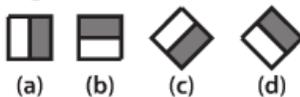
Data yang disimpan di *Flash Memory* akan tetap tersimpan dan tidak terhapus walaupun ESP32 dimatikan, ESP32 sendiri mempunyai *Flash Memory* dengan ukuran 4MB, *Flash Memory* sendiri dan lebih dikenal dengan istilah *EEPROM*[12]. Kuantitas dari data sampel sangat menentukan tingkat keakuratan pada proses pengujian nantinya[11].

Gambar 3 adalah beberapa contoh data sampel yang ditunjukkan untuk mahasiswa dengan nama Aldiansyah.

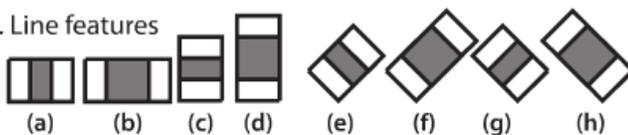
TABEL I
TABEL DATA SAMPEL

No	ID	Nama
1	51~75	Elvina
2	76~100	Puji
3	26~50	Abdul
4	1~25	Aldiansyah
5	101~125	Nuril

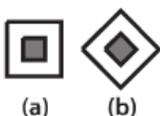
1. Edge features



2. Line features



3. Center-surround features



Gambar 2. Klasifikasi Haar



Gambar 3. Beberapa data sampel untuk Aldiansyah

D. Publish MQTT

Publish adalah istilah lain untuk mengirim data melalui protokol MQTT tentunya harus terkoneksi internet terlebih dahulu, *protokol* MQTT ini memerlukan suatu broker sebagai tempat lalu lintas data yang masuk maupun keluar. MQTT banyak digunakan untuk keperluan *Internet of Things*[13]. Sedangkan *Subscribe* adalah istilah lain untuk menerima data dari *broker* MQTT yang sudah terkoneksi.

MQTT (*MQ Telemetry Transport* atau *Message Queuing Telemetry Transport*) adalah OASIS terbuka dan standar ISO (ISO / IEC 20922) yang ringan, menerbitkan protokol jaringan berlangganan yang mengangkut pesan antar perangkat. Protokol biasanya berjalan melalui TCP / IP; namun, *protokol* jaringan apa pun yang menyediakan koneksi dua arah, tanpa kehilangan, dua arah dapat mendukung MQTT. *Protokol* jaringan ini dirancang untuk koneksi dengan lokasi terpencil di mana "jejak kode kecil" diperlukan atau *bandwidth* jaringan terbatas[13].

Protocol MQTT ini menggunakan konsep *publish* atau *subscribe*, Pusat komunikasi ada di MQTT *broker*, *broker* ini bertanggung jawab atas terkirimnya semua pesan termasuk jalur distribusinya. Setiap *client* yang mengirim pesan ke *broker*, termasuk juga mengirimkan *topic* ke dalam pesan tersebut. *Topic* merupakan bagian dari *routing information* untuk *broker*-nya. Tiap klien yang menginginkan menerima pesan, bisa meng-*subscribe* ke suatu *topic* tertentu dan *broker* akan mengirimkan semua *message* yang cocok dengan pola (*pattern*) *topic* tersebut kepada *client* yang sesuai.

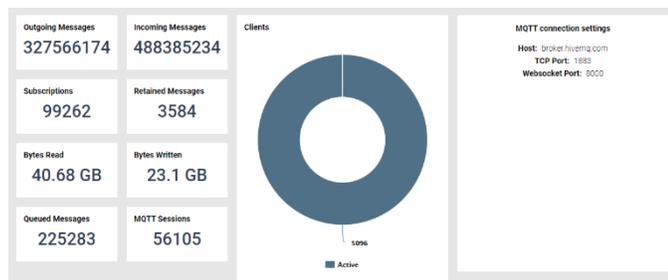
Banyak penyedia layanan *broker* MQTT gratis dan juga ada yang disediakan versi *browser* untuk mencoba langsung, salah satu penyedia layanan *broker* gratis adalah milik HiveMQ yang bekerja pada *port* *Websocket* 8000 dan *port* *TCP* 1883.

E. Webpage

Webpage adalah sebuah halaman *web* yang pembuatannya bertujuan untuk menyampaikan informasi kepada pengunjung *webpage* secara tulisan maupun berupa media foto/video.



Gambar 4. Broker gratis HiveMQ



Gambar 5. Diagram lalu lintas data pada HiveMQ

Gambar 6 merupakan struktur elemen yang umumnya digunakan pada *webpage* yang meliputi berbagai jenis *tag* mulai dari *tag html*, *head*, *style css*, *script javascript*, *body*. Sebuah *webpage* ditulis dengan bahasa html sebagai wadah utama yang hampir semua elemen untuk isi *webpage* ada di dalam *tag body*, sedangkan untuk *style css* sebagai penghias *page* agar tampilan *webpage* terlihat *responsif* sekaligus menarik bagi pengunjung *webpage*[15], dan *javascript* sebagai tambahan *event* tertentu seperti *onClick event* yang fungsinya jika *script* ini diletakan pada suatu *button*, maka yang terjadi jika *button* itu diklik akan mengaktifkan *onClick event*[16], seperti *webpage* yang digunakan dalam proyek sistem presensi mahasiswa berbasis *realtime* kamera ini juga menggunakan *onClick event* pada setiap *button* mulai dari *button connect*, *button subscribe*, dan *button disconnect* yang ditampilkan pada gambar 7.

Webpage yang menarik selain memberikan informasi, juga didesain secara rapi agar pengunjung tidak jenuh membaca sebuah artikel di *webpage*[14].

Pertama kali untuk mengkoneksikan *webpage* dengan *broker* hanya dengan menekan tombol *connect*, tombol *connect* ini akan mencoba tersambung dengan *address broker* tentunya harus dengan koneksi internet, setelah terkoneksi hal yang dilakukan selanjutnya adalah *subscribe*. *Webpage* akan menunggu ESP32-Cam untuk mengirim data.

```

<html>
<head>
<style>CSS</style>
<script>Javascript</script>
</head>
<body>
<div>Isi</div>
</body>
</html>
    
```

Gambar 6. Struktur elemen umum pada webpage

Connect Subscribe Disconnect

Disconnected

Presensi Mahasiswa

Fakultas	: Ilmu Komputer	Dosen Wali	: RANGSANG PURNAMAS Kom., M.Kom
Program Studi	: Sistem Komputer	Dosen Pembimbing	: CAHYO DARUJATI S.T., M.T
Mata Kuliah	: Skripsi	Kelas	: SK-Pagi
Kode Mata Kuliah	: SK81851		

No	NIM	Nama Mahasiswa
1	04116001	Elvina Rachma Astmasolihani
2	04116002	Puji Fitriah Hartini
3	04116003	Abdul Mumit
4	04116004	Aldiansyah Fanni Saputra
5	04116005	Nuril Huda

Gambar 7. Contoh webpage yang digunakan proyek ini

F. ESP32-Cam

ESP32-Cam adalah papan pengembang mode ganda *WiFi + Bluetooth* yang menggunakan antena dan ini papan PCB berbasis *chip* ESP32.

ESP32-Cam umumnya digunakan untuk berbagai macam IoT aplikasi, dikarekanan cocok untuk alat rumah pintar, pengendali sekaligus memonitori *wireless*, identifikasi *wireless*, dan aplikasi IoT lainnya yang menggunakan jaringan *wireless* maupun *bluetooth*[12].

ESP32-CAM adalah mikrokontroler berfitur lengkap yang juga memiliki kamera video terintegrasi dan soket kartu microSD. Ini murah dan mudah digunakan, dan sangat cocok untuk perangkat IoT yang membutuhkan kamera dengan fungsi-fungsi canggih seperti pelacakan dan pengenalan gambar[17].

ESP32-Cam adalah modul kamera yang sangat kecil dengan chip ESP32-S yang harganya sekitar \$ 10. Selain kamera OV2640, dan beberapa GPIO untuk menghubungkan *peripheral*, juga dilengkapi *slot* kartu microSD yang dapat berguna untuk menyimpan gambar yang diambil dengan kamera atau untuk menyimpan *file* untuk melayani klien[18].

Sepenuhnya sesuai dengan standar WiFi 802.11b / g / n / e / i dan *Bluetooth* 4.2, dapat digunakan sebagai mode master untuk membangun pengontrol jaringan independen, atau sebagai *client* MCU *host* lain untuk menambahkan kemampuan jaringan ke perangkat yang ada[13].



Gambar 8. ESP32-Cam

CAM	ESP32	SD	ESP32
D0	PIN5	CLK	PIN14
D1	PIN18	CMD	PIN15
D2	PIN19	DATA0	PIN2
D3	PIN21	DATA1/flash	PIN4
D4	PIN36	DATA2	PIN12
D5	PIN39	DATA3	PIN13
D6	PIN34		
D7	PIN35		
XCLK	PIN0		
PCLK	PIN22		
VSYNC	PIN25		
HREF	PIN23		
SDA	PIN26		
SCL	PIN27		
POWER PIN	PIN32		

Gambar 9. Pin ESP32-Cam

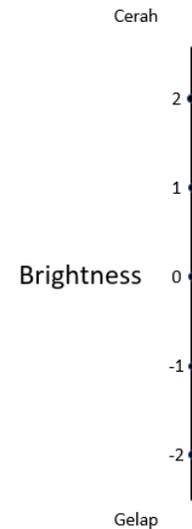
Gambar 9 merupakan pinout dari ESP32-Cam yang menyediakan pin digital sebanyak 8 pin dan pin analog sebanyak 40 pin.

III. HASIL DAN PEMBAHASAN

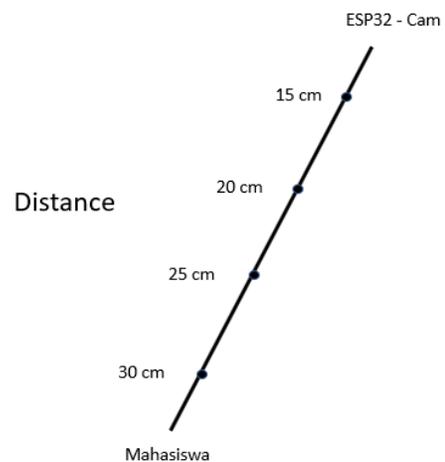
Berbagai macam kendala sering terjadi pada saat membutuhkan hasil yang bagus untuk mengenali wajah mahasiswa dengan tingkat keakuratan tinggi, semisal dalam pengambilan data sampel dan pada saat pengujian dengan *streaming* video, kendala yang muncul adalah dibutuhkannya kecukupan nilai kecerahan yang sesuai dan jarak yang tepat agar wajah bisa dikenali oleh sistem ESP32-Cam dalam proyek sistem presensi mahasiswa berbasis *realtime* kamera ini.

Tingkat kecerahan dan jarak antara sistem dengan mahasiswa yang diterapkan pada proses pengujian dengan *streaming* video akan ditampilkan pada gambar berikut ini.

Data sampel yang ada, diperoleh dari tingkat kecerahan yang berbeda, pada gambar 10 menampilkan skala batas pengambilan data sampel mulai dari skala tingkat kecerahan +2 sampai skala tingkat kecerahan -2. Dalam pengujian dengan *streaming* video, sistem mampu mengenali wajah pada kisaran skala tingkat kecerahan tersebut yang menjadi salah satu faktor yang mempengaruhi tingkat keakuratan.



Gambar 10. Tingkat Kecerahan



Gambar 11. Jarak ESP32-Cam dengan Mahasiswa

Gambar 11 menampilkan skala jarak yang penulis uji, yaitu antara ESP32-Cam sampai ke mahasiswa dengan variasi jarak yang berbeda mulai dari 15cm sampai dengan 30cm dan hasilnya adalah sistem presensi mahasiswa ini mampu mengenali wajah dengan baik dalam skala jarak antara 15 cm sampai dengan 30 cm.

Pada gambar 12 diambil ketika proses pengujian dengan streaming video pada mahasiswa dengan nama aldiansyah dan menunjukkan kecocokan pada sampel yang mempunyai nomor id 1 dengan tingkat kecerahan (0) dan dengan jarak (15cm), pada gambar 13 menunjukkan kecocokan pada sampel yang mempunyai nomor id 2 dengan tingkat kecerahan (1) dan dengan jarak (20cm), dan pada gambar 14 menunjukkan kecocokan pada sampel yang mempunyai nomor id 3 dengan tingkat kecerahan (-1) dan dengan jarak (30cm).



Gambar 12. Kecerahan (0), Jarak (15cm)



Gambar 13. Kecerahan (1), Jarak (20cm)



Gambar 14. Kecerhan (-1), Jarak (30cm)

$$a = \frac{n}{20} \times 100\% \quad [1]$$

Rumus [1] tersebut digunakan untuk menghitung tingkat keakuratan berdasarkan total jumlah pengujian. Variabel a pada rumus tersebut menunjukkan tingkat keakuratan, untuk mencari tingkat keakuratan diperlukan data sampel yang dilambangkan dengan variabel n dan jika ada yang berhasil dikenali langkah selanjutnya adalah data yang berhasil dikenali tersebut dibagi dengan jumlah keseluruhan dari proses pengujian yaitu sebanyak 20 pengujian dan langkah terakhir dikalikan dengan 100% untuk mendapatkan tingkat keakuratan.

Hasil yang diperoleh dari tingkat keakuratan sangat bervariasi mulai dari 80%-100% dalam kombinasi dari kondisi tingkat kecerahan dan jarak yang berbeda disajikan pada tabel II sampai tabel V ini.

Hasil akhir dari proyek ini adalah mampu menampilkan output dari ESP32-Cam ke dalam sebuah *webpage* akademik dengan

TABEL II

TABEL DATA KEAKURATAN PADA SETIAP TINGKAT KECERAHAN DALAM JARAK 15 CM

No	Kecerahan	Jarak	Keakuratan
1	-2	15 cm	85%
2	-1	15 cm	90%
3	0	15 cm	95%
4	1	15 cm	90%
5	2	15 cm	85%

TABEL III

TABEL DATA KEAKURATAN PADA SETIAP TINGKAT KECERAHAN DALAM JARAK 20 CM

No	Kecerahan	Jarak	Keakuratan
1	-2	20 cm	90%
2	-1	20 cm	95%
3	0	20 cm	100%
4	1	20 cm	95%
5	2	20 cm	90%

TABEL IV

TABEL DATA KEAKURATAN PADA SETIAP TINGKAT KECERAHAN PADA JARAK 25 CM

No	Kecerahan	Jarak	Keakuratan
1	-2	25 cm	85%
2	-1	25 cm	90%
3	0	25 cm	95%
4	1	25 cm	90%
5	2	25 cm	85%

TABEL V

TABEL DATA KEAKURATAN PADA SETIAP TINGKAT KECERAHAN PADA JARAK 30 CM

No	Kecerahan	Jarak	Keakuratan
1	-2	30 cm	80%
2	-1	30 cm	85%
3	0	30 cm	90%
4	1	30 cm	85%
5	2	30 cm	80%

bantuan *library mqtt* dengan *script* seperti *client.connect*, *client.publish* dan *client.subscribe*.

Skema koneksi pada gambar 15 membutuhkan masing-masing ESP32-Cam dan laptop yang menampilkan *webpage* terkoneksi dengan jaringan internet terlebih dahulu agar bisa tersambung ke layanan *broker* gratis HiveMQ. Setelah ESP32-Cam berhasil mengenali wajah dari mahasiswa selama proses pengujian dalam *streaming* video, akan segera mengirim *javascript client.publish* berupa nomor id sampel ke layanan *broker* gratis HiveMQ.

Sistem presensi mahasiswa berbasis *realtime* kamera ini menggunakan *username* dan *password* serta topik *subscribe* yang dibuat khusus untuk proses pengujian dengan adanya properti *javascript client.connect* yang berguna untuk bisa saling terkoneksi antara ESP32-Cam dengan laptop yang menampilkan *webpage* tersebut.

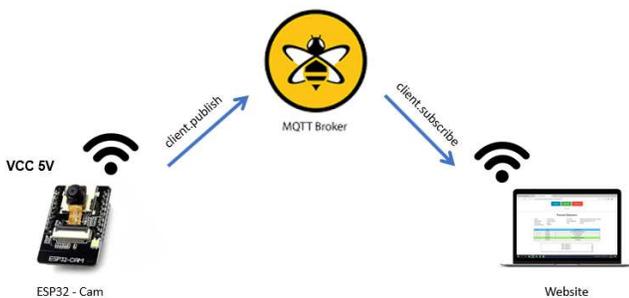
Gambar 16 menampilkan hasil *webpage* akademik buatan penulis yang bertujuan untuk menampilkan *output* dari ESP32-Cam dalam proses pengujian, dengan memanfaatkan adanya *library mqtt*, dan juga dengan memanfaatkan adanya *javascript client.subscribe* yang fungsinya adalah menunggu pesan yang diterima dari *output* ESP32-Cam dan langsung mencocokkan adanya kesamaan dengan pesan yang telah ditulis pada *javascript*.

Gambar 17 merupakan tampilan *webpage* ketika kelima mahasiswa hadir dan wajah-wajah kelima mahasiswa tersebut juga berhasil dikenali oleh sistem ESP32-Cam. Pada bagian bawah merupakan pesan *monitor* yang fungsinya untuk menampilkan nama topik yang telah di *subscribe* dan juga

menampilkan pesan yang dikirim ke topik yang sama. Terlihat berbagai macam sampel id yang berhasil dikirim oleh ESP32-Cam ke *webpage* melalui topik *aldi/nemu* yang dibuat khusus oleh penulis.

Pada gambar 18 adalah hasil tampilan layar penuh pada bagian pesan *monitor* yang menampilkan nama topik yang telah di *subscribe* dan pesan yang diterima berupa nomor sampel id, pesan *monitor* ini berada didalam *webpage* sistem presensi mahasiswa pada saat proses pengujian dengan *streaming* video. Terlihat berbagai macam pesan yang diterima melalui *client.subscribe* dan langsung ditampilkan pada pesan *monitor*, yang telah dikirim oleh sistem ESP32-Cam melalui *client.publish*, yang semua pesan-pesan itu merupakan nomor sampel id untuk wajah pada masing-masing mahasiswa yang cocok dengan data sampel dan telah berhasil dikenali dalam proses pengujian dengan *streaming* video.

Pada gambar 18 menampilkan berbagai macam nomor id pada



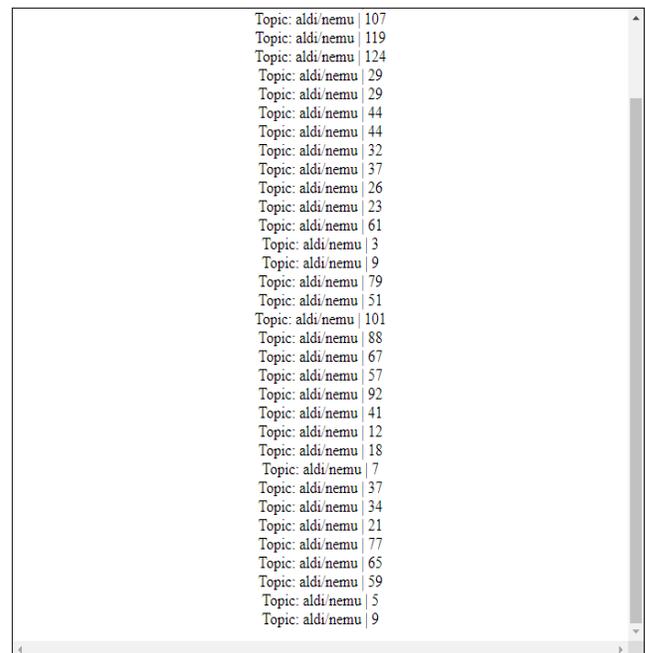
Gambar 15. Skema koneksi ke *broker mqtt*



Gambar 16. Tampilan *webpage* akademik



Gambar 17. Tampilan *webpage* akademik ketika semua hadir



Gambar 18. Tampilan penuh pada pesan saat pengujian *streaming*

pesan *monitor* yang telah dikirim oleh sistem ESP32-Cam berurutan dari nomor yang baru diterima sampai nomor yang sudah lama diterima, nomor tersebut mulai dari nomor id 107,119, 124, 29, 44, 32, 37, 26, 23, 61, 3, 9, 79, 51, 101, 88, 67, 57, 92, 41, 12, 18, 7, 37, 34, 21, 77, 65, 59, 5, dan 9.

Mahasiswa dengan nama Elvina mempunyai nomor id mulai dari 51 sampai dengan nomor id 75, maka untuk nomor id yang diterima dari sistem ESP32-Cam mulai dari nomor id 61, 51, 67, 57, 65, dan 59 mewakili mahasiswa dengan nama Elvina, kemudian di dalam *webpage* akan langsung mengubah warna pada latar belakang nama tersebut menjadi warna hijau.

Mahasiswa dengan nama Puji mempunyai nomor id mulai dari 76 sampai dengan nomor id 100, maka untuk nomor id yang diterima dari sistem ESP32-Cam mulai dari nomor id 79, 88, 92, dan 77 mewakili mahasiswa dengan nama Puji, kemudian di dalam *webpage* akan langsung mengubah warna pada latar belakang nama tersebut menjadi warna hijau.

Mahasiswa dengan nama Abdul mempunyai nomor id mulai dari 26 sampai dengan nomor id 50, maka untuk nomor id yang diterima dari sistem ESP32-Cam mulai dari nomor id 29, 44, 32, 37, 26, 23, 41, 37, 34, dan 21 mewakili mahasiswa dengan nama Abdul, kemudian di dalam *webpage* akan langsung mengubah warna pada latar belakang nama tersebut menjadi warna hijau.

Mahasiswa dengan nama Aldiansyah mempunyai nomor id mulai dari 1 sampai dengan nomor id 25, maka untuk nomor id yang diterima dari sistem ESP32-Cam mulai dari nomor id 3, 9, 12, 18, 7, 5, dan 9 mewakili mahasiswa dengan nama Aldiansyah, kemudian di dalam *webpage* akan langsung mengubah warna pada latar belakang nama tersebut menjadi warna hijau.

Mahasiswa dengan nama Nuril mempunyai nomor id mulai dari 101 sampai dengan nomor id 125, maka untuk nomor id yang diterima dari sistem ESP32-Cam mulai dari nomor id 107, 119, 124, dan 101 mewakili mahasiswa dengan nama Nuril, kemudian di dalam *webpage* akan langsung mengubah warna pada latar belakang nama tersebut menjadi warna hijau.

Semua nomor id yang berhasil diterima dari ESP32-Cam dan ditampilkan di *webpage* tersebut masing-masing mewakili salah satu dari kelima nama mahasiswa yaitu Elvina, Puji, Abdul, Aldiansyah, dan Nuril, yang akan ditampilkan dalam bentuk tabel VI ini.

TABEL VI

TABEL NOMOR ID YANG DITERIMA DARI ESP32-CAM

No	Nomor ID	Nama Mahasiswa
1	61, 51, 67, 57, 65, dan 59	Elvina
2	79, 88, 92, dan 77	Puji
3	29, 44, 32, 37, 26, 23, 41, 37, 34, dan 21	Abdul
4	3, 9, 12, 18, 7, 5, dan 9	Aldiansyah
5	107, 119, 124, dan 101	Nuril

IV. PENUTUP

A. Kesimpulan

Berdasarkan hasil dari pengujian yang dilakukan dalam perancangan sistem presensi mahasiswa berbasis *realtime* kamera dengan metode klasifikasi haar, dilihat dari hasil penelitian dan apa yang sudah dibahas sebelumnya, maka dapat disimpulkan bahwa sistem presensi bisa dilakukan menggunakan ESP32-Cam yang memanfaatkan metode klasifikasi haar dalam menemukan daerah yang termasuk wajah dan mampu memcocokkan dengan data sampel sehingga bisa mengenali wajah mahasiswa dalam berbagai kondisi yang bisa memengaruhi, misalnya pada tingkat kecerahan dan jarak antara sistem ESP32-Cam dengan mahasiswa saat *streaming* video, sehingga dalam proses uji tingkat keakuratan dengan pengujian sebanyak 20 kali pengujian untuk mahasiswa ditemukan beragam tingkat keakuratan mulai dari 80% sampai 100%.

Berdasarkan pengujian pada *webpage* buatan penulis dapat disimpulkan bahwa komunikasi dapat dilakukan antara sistem ESP32-Cam dengan laptop yang menampilkan *webpage* yang masing-masing terhubung dengan koneksi internet dan dengan adanya penyedia layanan *broker* gratis dari HiveMQ yang digunakan sebagai tempat *transfer* data selama proses pengujian untuk menampilkan hasil *output* berupa nomor sampel id setiap wajah mahasiswa yang tersimpan didalam *Flash Memory* dari sistem ESP32-Cam.

B. Saran

Saran pertama yang bisa diberikan untuk penelitian kedepannya adalah untuk mencoba menggunakan kamera yang mempunyai resolusi lebih tinggi dari ESP32-Cam, dikarenakan pada ESP32-Cam yang digunakan pada pengujian proyek sistem presensi mahasiswa hanya dibekali dengan kamera beresolusi 2MP, dan ESP32-Cam hanya dibekali dengan *Flash Memory* yang ukurannya adalah 4MB.

Saran kedua yang bisa diberikan untuk penelitian kedepannya adalah mampu mencari faktor lain yang dapat mempengaruhi tingkat keakuratan dalam mengenali wajah, selain dari yang telah dibahas sebelumnya oleh penulis yaitu faktor tingkat kecerahan dan jarak antara sistem ESP32-Cam dengan mahasiswa.

Saran ketiga yang bisa diberikan untuk penelitian kedepannya adalah adanya penambahan desain pada *webpage* dengan menambah *tag html*, *style css*, dan juga *script javascript* pada tampilan *web* agar terlihat lebih menarik.

Saran keempat yang bisa diberikan untuk penelitian kedepannya adalah adanya sebuah *database* yang terkoneksi dengan *webpage*, fungsinya untuk menyimpan data mahasiswa yang nantinya bisa disetorkan ke fakultas, karena pada proyek sistem presensi mahasiswa yang dibuat penulis ini masih belum terkoneksi ke *database*.

Saran kelima yang bisa diberikan untuk penelitian kedepannya adalah adanya penambahan mahasiswa dan juga penambahan data sampel pada setiap mahasiswa tambahan tersebut karena pada proyek sistem presensi yang dibuat penulis ini hanya menggunakan 125 total keseluruhan data sampel pada kelima mahasiswa.

V. KUTIPAN

- [1] Y. H. Huang and C. S. Fuh, "Face Detection and Smile Detection," *Proc. IPPR Conf. Comput. Vision, Graph. Image Proc. Shitou, Taiwan, A*, vol. 5, p. 108, 2007.
- [2] Nerissa, "Real-time system overview," pp. 1–9, 2009.
- [3] P. Prabhu, "DIGITAL IMAGE PROCESSING TECHNIQUES," no. May 2016, 2017.
- [4] C. Suhery and I. Ruslianto, "Identifikasi Wajah Manusia untuk Sistem Monitoring Kehadiran Perkuliahan menggunakan Ekstraksi Fitur Principal Component Analysis (PCA)," *J. Edukasi dan Penelit. Inform.*, vol. 3, no. 1, p. 9, 2017, doi: 10.26418/jp.v3i1.19792.
- [5] S. Sriram and B. Illuri, "Real Time Smile Detection using Haar Classifiers on SoC," *Int. J. Comput. Appl.*, vol. 104, no. 10, pp. 30–34, 2014, doi: 10.5120/18240-9297.
- [6] R. Padilla, C. C. Filho, and M. Costa, "Evaluation of haar cascade classifiers designed for face detection," *J. WASET*, vol. 6, no. 4, pp. 323–326, 2012.
- [7] H. Santoso, A. Harjoko, and A. Eko, "Efficient K-Nearest Neighbor Searches for Multiple-Face Recognition in the Classroom based on Three Levels DWT-PCA," *Int. J. Adv. Comput. Sci. Appl.*, vol. 8, no. 11, pp. 112–122, 2017, doi: 10.14569/ijacsa.2017.081115.
- [8] A. A. Tamimi, O. N. Al-Allaf, and M. A. Alia, "Eigen Faces and Principle Component Analysis for Face Recognition Systems: A Comparative Study," *Int. J. Comput. Technol.*, vol. 14, no. 4, pp. 5650–5660, 2015, doi: 10.24297/ijct.v14i4.1967.
- [9] M. Slavkovic and D. Jevtic, "Face recognition using eigenface approach," *Serbian J. Electr. Eng.*, vol. 9, no. 1, pp. 121–130, 2012, doi: 10.2298/sjee1201121s.
- [10] S. Singh, "Facial Expression Detection and Recognition Haar Cascade Classifier and Fisherface Algorithm," pp. 1–3, 2019.
- [11] Derisma, "Faktor-Faktor yang Mempengaruhi Sistem Pengenalan Wajah Menggunakan Metode Eigenface pada Perangkat Mobile Berbasis Android," *J. Politek. Caltex Riau*, vol. 2, no. 2, pp. 127–136, 2016.
- [12] S. A.-T. Technology Co, "ESP32-CAM Wi-Fi +BT SoC Modul e V1.0," pp. 1–2, 2017.
- [13] N. R. Ahsy, A. Bhawiyuga, and D. P. Kartikasari, "Implementasi Sistem Kontrol dan Monitoring Smart Home Menggunakan Integrasi Protokol Websocket dan MQTT," *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 3, no. 4, 2019.
- [14] Y. Sen Sun, B. Qiu, and Q. S. Li, "Pemanfaatan Sistem Informasi Perpustakaan Digital Berbasis Website Untuk Para Penulis," *Adv. Mater. Res.*, vol. 756–759, no. 1, pp. 138–140, 2013, doi: 10.4028/www.scientific.net/AMR.756-759.138.
- [15] Y. A. Binarso, E. A. Sarwoko, and N. bahtiar Ba, "434-853-1-Sm," *Pembang. Sist. Inf. Alumni Berbas. Web Pada Progr. Stud. Tek. Inform. Univ. Diponegoro*, vol. 1, no. 1, pp. 72–84, 2012.
- [16] F. Constantianus and B. Suteja, "Analisa dan Desain Sistem Bimbingan Tugas Akhir Berbasis Web dengan Studi Kasus Fakultas Teknologi Informasi," *J. Inform.*, vol. 1, no. 2, pp. 93–106, 2005.
- [17] F. N. Iksan and G. Tjahjadi, "Perancangan Stop Kontak Pengendali Energi Listrik Dengan Sistem Keamanan Hubung Singkat Dan Fitur Notifikasi Berbasis Internet of Things (IoT)," *J. Elektro*, vol. 11, no. 2, pp. 83–92, 2018.
- [18] A. Setiawan and A. I. Purnamasari, "Pengembangan Passive Infrared Sensor (PIR) HC-SR501 dengan Microcontrollers ESP32-CAM Berbasis Internet of Things (IoT) dan Smart Home sebagai Deteksi Gerak untuk Keamanan Perumahan," *Pros. Semin. Nas. Sist. Inf. dan Teknol.*, pp. 148–154, 2019.

formal secara berturut-turut di TK Al-Utsmaniyah Surabaya (2003-2004) SDN Petemon XIII Surabaya (2004-2010), SMPN 10 Surabaya (2010-2014), SMK Barunawati Surabaya Jurusan Multimedia (2014-2016). Pada tahun 2016, penulis melanjutkan Pendidikan S1 di Program Studi Sistem Komputer, Fakultas Ilmu Komputer, Universitas Narotama Surabaya.

TENTANG PENULIS



Aldiansyah Famni Saputra, yang biasa dipanggil dengan nama Aldi lahir di Surabaya pada tanggal 15 Juni 1998. Alamat tempat tinggal penulis di Perumahan Taman Anggun Sejahtera III Blok N6 Nomor 24 Wonoayu, Sidoarjo. Penulis menempuh Pendidikan