

## BAB II TINJAUAN PUSTAKA

### 2.1 Tinjauan Literatur

Penelitian mengenai sistem *face recognition* telah banyak dilakukan namun penulis menggunakan modul yang berbeda yang lebih murah dan mudah didapatkan. Berikut penulis jelaskan secara detail referensi dari penelitian-penelitian sebelumnya.

Pada tahun 2014 perancangan sistem identifikasi wajah dengan teknik feature-based diterapkan pada aplikasi Safety Box dengan cara membandingkan wajah yang identik dengan database dengan wajah yang tidak identik dengan database dengan menggunakan metode *SURF (Speeded-Up Robust Features)*. Pada metode ini citra masukan di deteksi kemudian dipotong dan disesuaikan skalanya setelah itu citra dinormalisasikan dengan mengubah citra dan RGB ke grayscale. Metode selanjutnya *CLAHE (Contrast Limited Adaptive Equalization)* yang dilanjutkan dengan matching feature dengan cara mempasangkan pasangan yang merupakan bukan hanya mendeteksi wajah saja tetapi sesuai dengan nilai poin kesamaan dimana wajah berhasil diidentifikasi pada pengujian I dan II dikarenakan wajah yang identik dengan database dengan nilai kesamaan lebih dari 7 namun pada pengujian III dan IV persentase identifikasi wajah 0 persen dikarenakan wajah tidak identik dengan database [4].

Pada tahun 2018 dilakukan penelitian tentang *Techniques and Challenges of Face Recognition: A Critical Review* penelitian ini membandingkan akurasi sistem face recognition dengan menggunakan berbagai teknik dan aplikasi. Berdasarkan penelitian ini akurasi yang didapatkan dari sistem face recognition menggunakan metode sebesar lebih dari 97 persen bahkan yang tertinggi mencapai 99,93 persen ini menyatakan bahwa akurasi dari sistem face recognition sama baiknya dengan sistem *fingerprint* [5].

Pada tahun 2019 dilakukan penelitian mengenai Sistem Absensi Mahasiswa Berdasarkan Fitur Pengenalan Wajah dengan Menggunakan Metode *Haar-Like Feature* dengan menggunakan Raspberry Pi dengan cara mengenali nilai sederhana

dari fitur. Pada umumnya mahasiswa melakukan absensi secara manual maka dari itu dengan dibuatnya sistem absensi pengenalan wajah diharapkan mahasiswa akan lebih mudah dalam melakukan absensi dan menghindari kemungkinan kecurangan yang bisa dilakukan. Software yang digunakan adalah OpenCV (*Open-Source Computer Vision*) yaitu library software yang digunakan untuk library image processing dan computer vision, OpenCV dapat menggunakan berbagai bahasa pemrograman seperti Java, C, C++, Python dan mensupport berbagai Operating System mulai dari Windows, Mac OS, Linux, Ubuntu, iOS dan Android. Pola wajah setiap mahasiswa dilakukan sebanyak 30-50 kali agar akurasi yang dihasilkan dari sistem meningkat. Sistem absensi berdasarkan pola wajah yang dilakukan dinilai efektif karena berhasil dibangun dan diimplementasikan, dan data berhasil disinkronisasikan menggunakan website yang membuat proses rekap data absensi mahasiswa menjadi lebih singkat menjadi 3-4 hari dari sebelumnya 6-8 ketika dilakukan menggunakan absensi manual [6].

Pada tahun 2021 implementasi ESP32-CAM dengan Arduino pro mini digunakan sebagai thermogun otomatis berbasis IoT untuk monitoring suhu. Digunakan modul ESP32-CAM yang dihubungkan dengan Arduino pro mini dengan Sensor Suhu MLX 90614 dengan buzzer sebagai indikator suara. Data akan diterima dan dihubungkan dengan server dimana ESP32-CAM akan mengambil foto dan mengirimkannya ke server di dalam sistem. Sensor suhu dibandingkan dengan Thermogun untuk mengukur akurasi. Akurasi yang didapatkan MLX90614 sebesar 99,6 persen. Pada penelitian ini sensor proksimitas sering terkendala dikarenakan terlalu sensitif untuk mengukur objek namun ini bisa diperbaiki dengan membuat konstruksi mekanis yang dapat membatasi sensitivitas sensor proksimitas [7].

Pada tahun 2021 penelitian pengenalan wajah berbasis ESP32-CAM untuk sistem kunci sepeda motor dilakukan untuk mengurangi angka pencurian motor dengan menggunakan sistem keyless (tanpa kunci) yaitu dengan menggunakan modul ESP32-CAM yang disambungkan dengan relay sebagai switch untuk menyambungkan dan memutuskan kelistrikan sepeda motor yang sebelumnya disambungkan dengan modul Stepdown 12V to 5V agar dapat dijadikan daya untuk ESP32-CAM yang kemudian disambungkan dengan LED agar wajah tetap

terdeteksi walaupun dalam kondisi kurang pencahayaan. Pada pengujian yang dilakukan sebanyak 40 kali alat dapat membedakan wajah pemilik motor dengan wajah yang bukan pemilik motor sebesar 80 persen untuk kondisi benar dan 17,5% untuk kondisi salah [8].

Pada tahun 2022 penelitian mengenai evaluasi kamera ESP32-CAM untuk berbagai projek dilakukan untuk mengetahui performa dari ESP32-CAM yg dihubungkan dengan Arduino sebagai mikrokontroler, percobaan yang dilakukan seperti solenoid valve dan projek lainnya yang menggunakan antenna UFL dan Analisa OV2640 kamera meliputi waktu tunda, resolusi dan kecepatan penangkapan gambar. UXVGA dan SXVGA adalah resolusi dengan kecepatan rata-rata tertinggi >80ms dibandingkan dengan tipe resolusi yang lain pada kamera OV2640 ESP32-CAM sebagai perbandingan tipe yang kompatibel untuk *face recognition* dan image recognition adalah CIF dan QQVGA yang memiliki kecepatan rata-rata 40ms dan 19ms [9].

## 2.2 Teori Pengolahan Citra Digital

*Image processing* atau penyusun citra adalah salah satu teknik yang digunakan dalam pengolahan citra digital untuk memanipulasi atau memproses gambar dalam tingkatan dua dimensi. Pemrosesan gambar digital didasarkan pada konsep dasar yang diambil dari kemampuan penglihatan manusia, yang kemudian dihubungkan dengan kemampuan otak manusia untuk memproses dan mengolah gambar digital tersebut [25].

Seiring perkembangan zaman, *image processing* mampu dioptimalisasikan dan diterapkan pada berbagai bidang. Image processing telah dengan sukses diterapkan pada berbagai ragam ilmu pengetahuan dalam berbagai kepentingannya masing-masing dimana didapatkan hasil tingkat kesuksesan yang tinggi. Cabang-cabang ilmu pengetahuan tersebut mencakup elektronika, biometrik, fotografi, matematika, dan teknologi komputer. Tujuan utama digital image processing adalah mengolah dan menganalisis gambar atau citra digital dengan seksama, sehingga menghasilkan informasi baru yang lebih menguntungkan.

Pada pengolahan citra terdapat empat kategori dasar, yakni: *point*, *area*, *geometric* dan *frame point*. Pada *point*, citra diproses berdasarkan nilai atau posisi pikselnya. Sebagai ilustrasi, terdapat *adding*, *subtracting*, dan *stretching*. Pada *area*, pengolahan dilakukan berdasarkan nilai piksel dan nilai piksel di sekitarnya. Contoh termasuk *convolution*, *blurring*, dan *sharpening*. Kategori *geometric* dimanfaatkan untuk mengubah lokasi piksel. Sebagai contohnya adalah *scaling*, *rotation*, *translation*, dan *mirroring*. Pada *frame*, citra diproses dengan menggabungkan dua citra atau lebih. Contoh dari proses bingkai mencakup penambahan, pengurangan, dan logika "and" atau "or".

Pengolahan citra digital dibagi menjadi tiga jenis, yaitu: Proses tingkat rendah (*Low-level*) yang melibatkan operasi-operasi primitif seperti pra-pemrosesan citra untuk mengurangi noise, meningkatkan kontras, dan mempertajam gambar. Pada proses tingkat rendah, masukan dan keluaran berupa citra. *Mid-level process* melibatkan tugas-tugas seperti segmentasi citra (memecah gambar menjadi objek-objek), pengenalan objek individu. Pada *mid-level process*, masukan umumnya berupa citra tetapi keluarannya adalah atribut yang dihasilkan dari proses pada citra tersebut, seperti garis, kontur garis, dan objek individu. Proses tingkat tinggi (*High-level*) adalah tahap lanjutan yang berkaitan dengan hasil dari proses tingkat menengah (*Mid-level*).

Pengolahan Citra (*image processing*) memiliki komponen-komponen berikut, diantaranya adalah: Piksel (*picture element*) merupakan titik terkecil yang membentuk gambar, juga disebut sebagai "dot". Resolusi menunjukkan jumlah piksel yang membentuk gambar atau lapisan. Ukurannya adalah hasil kali jumlah piksel secara horizontal dan vertikal. Sebagai contoh, ukuran 640 x 480 berarti 640 piksel secara horizontal dan 480 piksel secara vertikal. *Dots per inch* (dpi) berfungsi untuk Mengukur berapa banyak piksel atau titik yang ada dalam satu inci. Sebagai contoh, 600 dpi berarti terdapat 600 x 600 piksel dalam setiap inci persegi. Warna, pada warna terdapat empat model umum dalam pengolahan citra digital, yaitu model warna *gray-scale*, model warna RGB, model warna HSV, dan model warna YCrCb. Kedalaman warna (*color depth*) merupakan seberapa besar informasi yang disimpan dalam satu sampel (dot/piksel). Setiap dot/piksel dapat memiliki warna hitam, putih, abu-abu, atau warna lainnya. Kedalaman warna diukur dalam angka

yang mengindikasikan berapa banyak variasi warna yang dapat ditampilkan oleh satu *bitmap/dot/piksel* secara kuantitatif. Rumusnya adalah  $2^n$  ( $2$  pangkat  $n$ ), di mana  $n$  adalah jumlah bit dalam kedalaman warna. Sebagai contoh: 1 bit =  $2^1 = 2$  variasi warna (monokrom), 2 bit =  $2^2 = 4$  variasi warna, 4 bit =  $2^4 = 16$  variasi warna, 8 bit =  $2^8 = 256$  variasi warna, 16 bit =  $2^{16} = 65536$  variasi warna, 32 bit =  $2^{32} = 4294967296$  variasi warna. Perhitungan ukuran gambar dilakukan dengan rumus: Ukuran dalam *byte* = lebar x tinggi x kedalaman warna. Penjelasan singkat: Lebar adalah dimensi horizontal gambar, diukur dalam piksel. Tinggi adalah dimensi vertikal gambar, diukur dalam piksel. Kedalaman warna adalah jumlah bit warna per piksel. 1 byte = 8 bit. 1 Kb = 1024 *byte*. 1 Mb = 1024 *kilobyte*.

### 2.2.1 Operasi Pengolahan Citra

Berbagai jenis operasi dilakukan dalam pengolahan citra. Secara garis besar, operasi ini dapat dikelompokkan menjadi beberapa jenis, seperti perbaikan kualitas citra (*image enhancement*) yang bertujuan meningkatkan mutu citra melalui manipulasi parameter-parameter citra. Dengan metode ini, fitur-fitur spesifik dalam citra dapat lebih ditekankan. Beberapa contoh operasi perbaikan citra mencakup peningkatan kontras dalam area gelap atau terang, peningkatan tepi objek (*edge enhancement*), penguatan (*sharpening*), pemberian tampilan warna palsu (*pseudocoloring*), dan penyaringan derau (*noise filtering*) [26].



(a)

(b)

Gambar 2. 1 (a) Citra Lena asli, (b) Citra Lena setelah ditajamkan [30]

Gambar 2.1 adalah contoh operasi penguatan. Pada operasi ini citra menerima masukan yang gambarnya akan dibentuk jadi lebih tajam. Pada Gambar 2.1 (b) citra yang dilakukan penajaman adalah tepi-tepi objek.

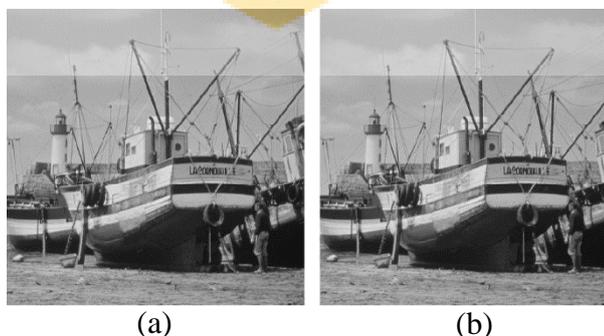
Pemulihan citra (*image restoration*) berfungsi untuk menghapus atau meminimalkan cacat pada citra. Tujuannya hampir serupa dengan operasi perbaikan citra. Namun, perbedaannya terletak pada pemulihan citra yang melibatkan pengetahuan tentang penyebab degradasi gambar. Beberapa contoh operasi pemulihan citra mencakup pengembalian ketajaman (*deblurring*) dan pengurangan derau (*noise reduction*) [28].



Gambar 2. 2 (a) Citra Lena *blur* (b) citra Lena setelah *deblurring* [27]

Gambar 2.2 menunjukkan sebuah contoh dari operasi penghilangan ketajaman. Citra asalnya mengalami efek kabur (*blur*). Kemungkinan penyebabnya bisa berasal dari penyeteralan fokus lensa yang tidak akurat atau getaran kamera saat pengambilan gambar. Dengan menerapkan operasi *deblurring*, kualitas citra asli dapat dipulihkan sehingga menghasilkan tampilan yang lebih baik.

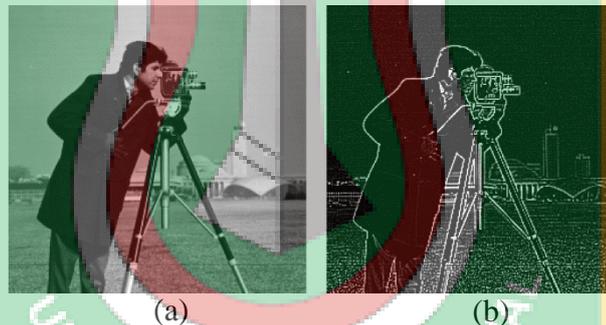
Pemampatan citra (*image compression*) dilakukan untuk menggambarkan citra dalam format yang lebih ringkas, mengurangi kebutuhan ruang penyimpanan. Dalam proses kompresi, penting untuk mempertahankan kualitas citra meskipun ukurannya berkurang. Salah satu contoh metode kompresi citra adalah metode JPEG [31].



Gambar 2. 3 (a) Citra boat.bmp sebelum dimampatkan, (b) citra boat.jpg sesudah dimampatkan [29]

Pada Gambar 2.3 di sebelah kiri adalah citra kapal berukuran 258 kb. Metode yang digunakan adalah metode JPEG dari hasil pemampatan citra ukuran citra dapat direduksi dari semula hingga menjadi 49 kb.

Segmentasi citra (*image segmentation*) adalah proses membagi citra menjadi beberapa segmen berdasarkan kriteria tertentu. Operasi ini berhubungan erat dengan pengenalan pola. Sementara itu, analisis citra (*image analysis*) memiliki tujuan menghitung besaran kuantitatif dari citra guna membuat deskripsi. Teknik analisis citra membantu mengambil ciri-ciri spesifik yang mendukung identifikasi objek. Terkadang, proses segmentasi diperlukan untuk mengisolasi objek yang diinginkan dari lingkungannya. Beberapa contoh operasi analisis citra termasuk deteksi tepian objek (*edge detection*), ekstraksi batas (*boundary*), serta representasi wilayah (*region*). Gambar 2.4 menunjukkan contoh operasi deteksi tepi pada citra kamera. Operasi ini menghasilkan seluruh tepi (*edge*) yang ada dalam citra tersebut.



Gambar 2. 4 (a) Citra camera, (b) citra hasil pendeteksian seluruh tepi [33]

Rekonstruksi citra (*image reconstruction*) berfungsi untuk mengembalikan objek ke bentuk semula dari serangkaian citra proyeksi. Operasi rekonstruksi citra sering diterapkan dalam bidang medis. Sebagai contoh, beberapa gambar sinar-X dari berbagai sudut digunakan untuk membangun kembali citra organ tubuh dalam bidang radiologi [26].

### 2.2.2 Image Resizing

*Image resizing* adalah salah satu aspek dalam bidang pengolahan citra. Proses ini mengubah ukuran gambar digital, baik memperbesar maupun memperkecil. Fokus disini adalah pada perubahan ukuran gambar menjadi lebih besar dengan mempertahankan kualitas citra digital. Terdapat berbagai algoritma

*image resizing* yang tersedia saat ini, seperti interpolasi bilinear, interpolasi kubik, *nearest neighbor interpolation*, dan interpolasi bikubik. Algoritma interpolasi ini digolongkan menjadi dua jenis, yaitu algoritma adaptif dan algoritma non-adaptif.

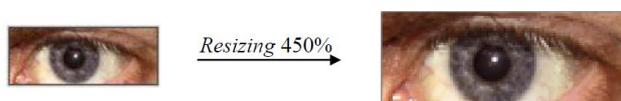
*Adaptive algorithm* adalah jenis algoritma yang dapat menyesuaikan diri tergantung pada apa yang diinterpolasi, seperti tekstur halus dan sebagainya. Algoritma ini dapat membedakan dan menyesuaikan proses untuk berbagai bagian dalam citra digital yang berbeda. Beberapa aplikasi yang menggunakan algoritma adaptif termasuk *Qimage*, *PhotoZoom Pro*, dan *Genuine Fractals* [36].



Gambar 2. 5 Adaptive Algorithm [36]

*Non-adaptive algorithm* adalah jenis algoritma yang tidak membedakan bagian-bagian khusus dalam citra digital, sehingga memproses setiap piksel dengan cara yang seragam. Beberapa metode dalam kategori *non-adaptive algorithm* mencakup *nearest neighbor*, *bilinear*, *bicubic*, *spline*, *sinc*, dan *lanczos*.

Algoritma *Nearest neighbor interpolation* merupakan algoritma sederhana dan cepat dalam melakukan perubahan ukuran gambar digital. Pada dasarnya, algoritma ini hanya memperbesar piksel menjadi lebih besar. Kelebihan algoritma ini terletak pada kecepatannya, yang sangat cocok untuk aplikasi yang membutuhkan kinerja tinggi, seperti tampilan *thumbnail* pada *Windows Explorer*. Ilustrasi hasil perubahan ukuran gambar menggunakan algoritma *nearest neighbor interpolation* dapat ditemukan pada Gambar 2.6 [25].



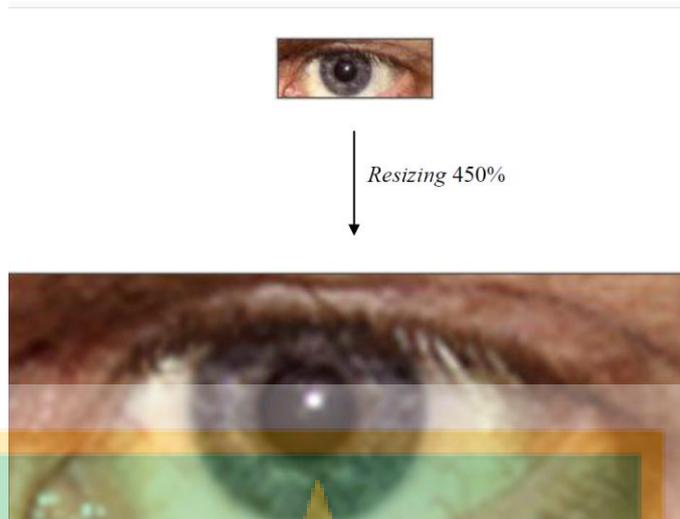
Gambar 2. 6 Nearest Neighbor Interpolation [35]

Algoritma interpolasi bilinear (*Bilinear interpolation algorithm*) menggunakan empat piksel terdekat dengan ukuran  $2 \times 2$  untuk menghitung nilai piksel yang tidak diketahui. Selanjutnya, nilai rata-rata dari keempat piksel ini diambil sebagai nilai interpolasi, tanpa memperhitungkan jarak antara piksel-piksel tersebut. *Bilinear interpolation* menghasilkan perubahan ukuran gambar yang lebih baik dibandingkan dengan *nearest neighbor interpolation*. Contoh hasil perubahan ukuran gambar menggunakan algoritma *bilinear interpolation* dapat dilihat dalam Gambar 2.7.



Gambar 2. 7 Bilinear Interpolation [35]

Dalam konteks ilmu matematika, *bicubic interpolation* merupakan perluasan dari interpolasi kubik (*cubic interpolation*) yang berguna untuk meratakan titik-titik data dalam grid dua dimensi biasa. *Bicubic interpolation* melibatkan penggunaan  $4 \times 4$  piksel tetangga terdekat dengan area yang akan diinterpolasi. Metode ini menghasilkan permukaan yang lebih halus dibandingkan dengan algoritma interpolasi *nearest-neighbor*, *bilinear*, dan kubik. *Bicubic interpolation* bisa diimplementasikan melalui algoritma *Lagrange polinomial*, *cubic splines*, atau *cubic convolution*. Dalam pengolahan gambar digital, *bicubic interpolation* sering dipilih jika prioritasnya adalah menghasilkan kualitas gambar yang lebih tinggi daripada memaksimalkan kecepatan proses *resizing*. Gambar yang dihasilkan dengan algoritma *bicubic interpolation* cenderung lebih halus dengan sedikit artefak interpolasi. Contoh hasil perubahan ukuran gambar menggunakan algoritma *bicubic interpolation* dapat dilihat dalam gambar 2.8 di bawah ini [35].



Gambar 2. 8 Bicubic Interpolation [35]

### 2.3 Teknologi Face Recognition

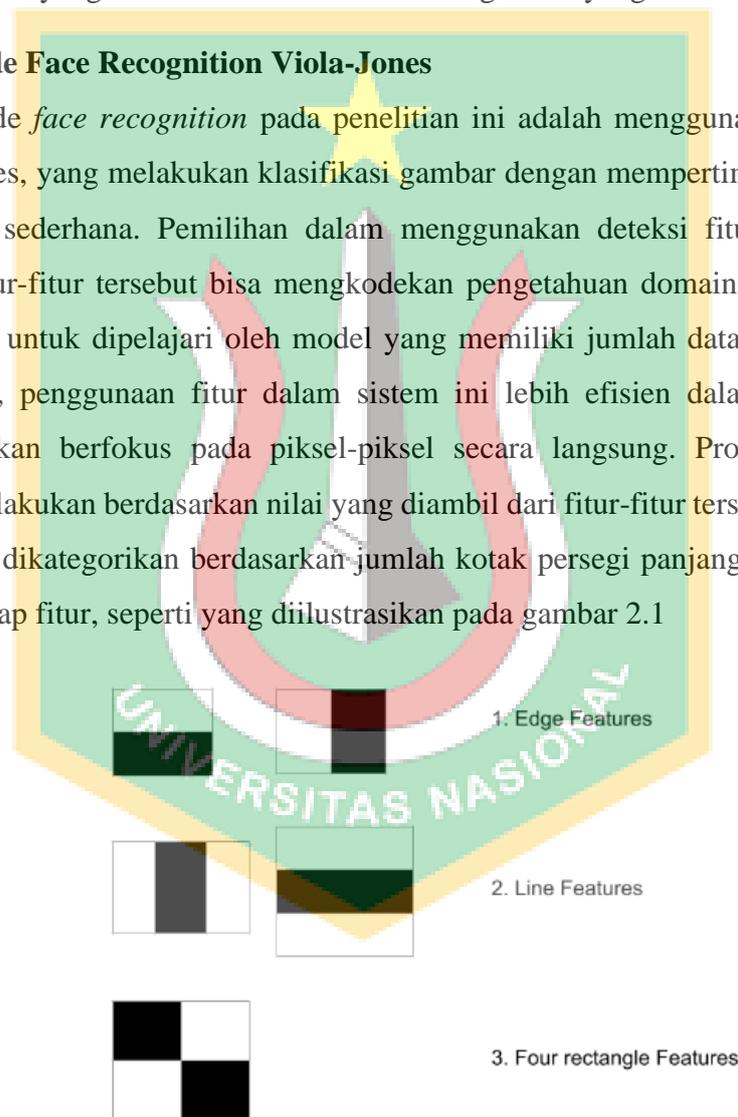
Pengenalan wajah (*Face Recognition*) adalah teknologi yang memiliki kemampuan untuk mengidentifikasi serta mengonfirmasi identitas seseorang dengan menggunakan fitur wajah. Teknologi pengenalan wajah ini merupakan salah satu bentuk sistem identifikasi biometrik yang efektif dalam membedakan individu berdasarkan ciri-ciri unik pada wajah mereka. Menurut Lembaga Akuntabilitas Pemerintah Amerika Serikat (*US Government Accountability Office*), terdapat empat elemen utama yang diperlukan untuk melaksanakan pengenalan wajah, yakni kamera, cetakan wajah (*faceprint*), basis data (*database*), serta algoritma yang bertugas membandingkan cetakan wajah dari individu target dengan cetakan wajah yang ada dalam basis data (Lembaga Akuntabilitas Pemerintah Amerika Serikat. GAO-21-526, 2021) [2].

Ada 3 Tahapan umum dalam pengenalan wajah, yaitu: 1. Deteksi (*Detection*) – Pada tahap ini, sistem akan mengidentifikasi pola dalam suatu gambar dan membandingkannya. Jika pola yang diidentifikasi sesuai, sistem akan mengasumsikan bahwa gambar tersebut mengandung wajah. 2. Pembuatan Cetakan Wajah (*Faceprint Creation*) – Ini merupakan 'cetakan' atau 'model' dari wajah. Untuk membuat cetakan wajah, ada dua pendekatan umum yang digunakan: pendekatan geometris (*geometric approach*) dan pendekatan fotometrik

(*photometric approach*). 3. Verifikasi atau Identifikasi – Pada tahap verifikasi, gambar wajah input dibandingkan dengan data gambar wajah pengguna yang membutuhkan autentikasi. Perbandingannya dilakukan dalam skema 1x1. Sementara itu, pada tahap identifikasi, perbandingan dilakukan dalam skema 1xN. Dalam identifikasi, gambar wajah input dibandingkan dengan seluruh gambar wajah dalam basis data guna dilakukan pencocokan. Faktor yang dapat menghambat analisis meliputi rambut panjang yang menutupi bagian tengah wajah, pencahayaan yang tidak memadai dan resolusi gambar yang rendah [12].

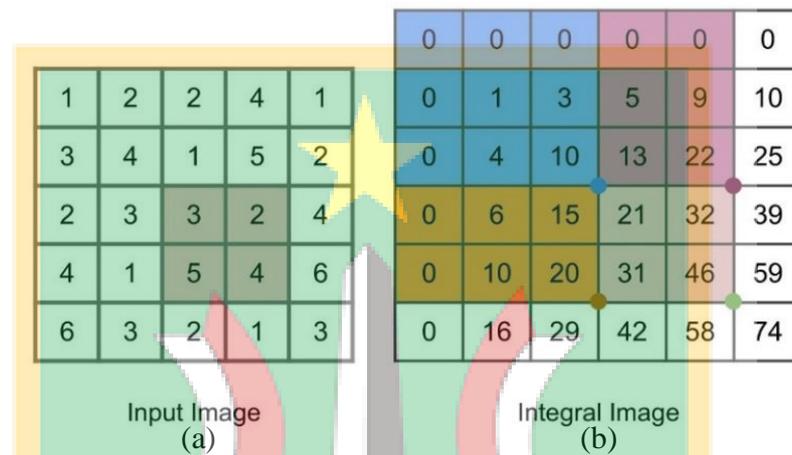
## 2.4 Metode Face Recognition Viola-Jones

Metode *face recognition* pada penelitian ini adalah menggunakan algoritma Viola-Jones, yang melakukan klasifikasi gambar dengan mempertimbangkan nilai fitur-fitur sederhana. Pemilihan dalam menggunakan deteksi fitur ini didasari bahwa fitur-fitur tersebut bisa mengkodekan pengetahuan domain secara ad-hoc yang sulit untuk dipelajari oleh model yang memiliki jumlah data latih terbatas. Selain itu, penggunaan fitur dalam sistem ini lebih efisien dalam operasional dibandingkan berfokus pada piksel-piksel secara langsung. Proses klasifikasi gambar dilakukan berdasarkan nilai yang diambil dari fitur-fitur tersebut [10]. Tiga jenis fitur dikategorikan berdasarkan jumlah kotak persegi panjang yang terdapat dalam setiap fitur, seperti yang diilustrasikan pada gambar 2.1

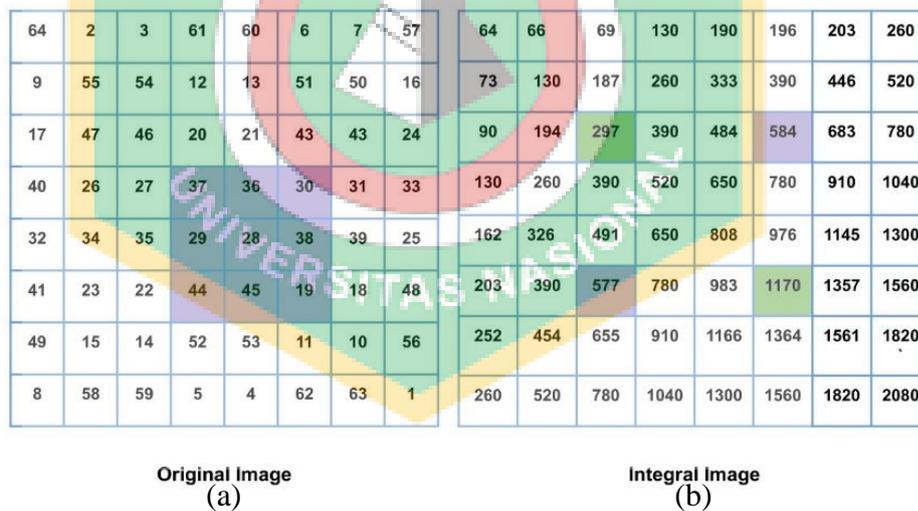


Gambar 2. 9 Klasifikasi fitur berdasarkan metode Viola-Jones [6]

Fitur ini bermanfaat dalam menghitung perbedaan nilai piksel antara area putih dan area hitam. Guna menyederhanakan perhitungan nilai fitur, algoritma Viola-Jones memanfaatkan pendekatan berupa *Integral Image*. *Integral Image* adalah representasi citra di mana nilai setiap pikselnya merupakan hasil akumulasi nilai piksel dari pojok kiri atas hingga piksel yang bersangkutan. Sebuah contoh dari *Integral Image* dapat dilihat dalam gambar di bawah ini:



Gambar 2. 10 (a) Input Image & (b) Integral Image [11]



Gambar 2. 11 (a) Original Image & (b) Integral Image [11]

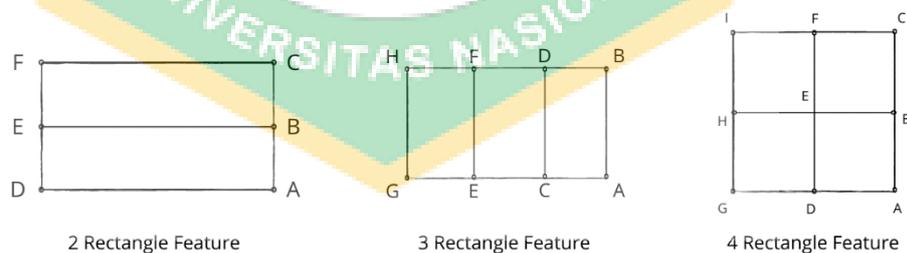
*Haar-like feature* merupakan fungsi persegi yang memberikan petunjuk spesifik pada suatu gambar. Prinsip dasar dari *Haar-like feature* adalah mengidentifikasi objek berdasarkan nilai fitur yang sederhana, bukan berdasarkan

nilai piksel langsung dari objek gambar tersebut. Pendekatan ini memiliki keunggulan dalam kecepatan komputasi yang tinggi, karena hanya bergantung pada jumlah piksel dalam kotak, bukan seluruh nilai piksel dalam gambar [13].

Pada tahap awal, pengolahan gambar hanya melibatkan analisis nilai RGB setiap piksel. Namun, metode ini terbukti kurang efisien, dan kemudian Viola-Jones memperluas pendekatan ini menjadi *Haar-like features*. *Haar-like feature* memproses gambar dalam bentuk kotak-kotak yang mencakup beberapa piksel. Setiap kotak ini dianalisis dan menghasilkan perbedaan nilai (*threshold*) yang menunjukkan area gelap dan terang. Nilai-nilai ini menjadi dasar dalam proses *image processing* [10].

#### 2.4.1 Perhitungan Haar-like Features dengan integral image.

Dengan menerapkan *integral image*, kita dapat mencapai evaluasi waktu yang konstan untuk *Haar-like features*. Fitur tepi (*edge features*) atau 2 fitur persegi panjang hanya memerlukan 6 operasi pencarian dalam memori. Fitur garis (*line features*) atau 3 fitur persegi panjang hanya membutuhkan 8 operasi pencarian dalam memori. Sementara fitur diagonal (*diagonal features*) atau 4 fitur persegi panjang hanya membutuhkan 8 operasi pencarian dalam memori. Rumus untuk masing-masing tipe fitur adalah sebagai berikut: 2 Rectangle =  $A-2B+C-D+2E-F$ . 3 Rectangle =  $A-B-2C+2D+2E-2F-G+H$ . 4 Rectangle =  $A-2B+C-2D+4E-2F+H-2I+J$  [11].

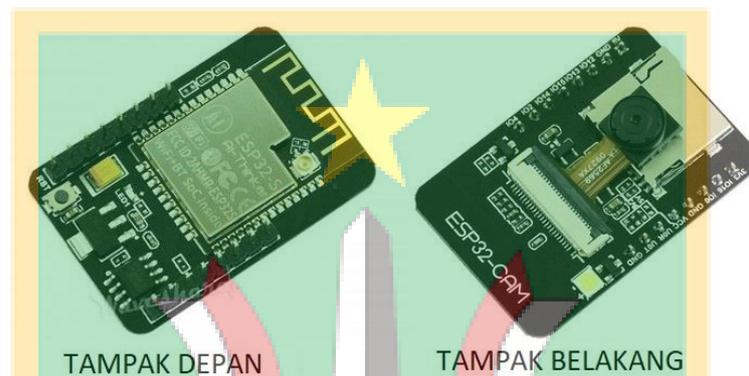


Gambar 2. 12 Fitur persegi panjang Haar-Like [11]

## 2.6 Modul ESP32-CAM

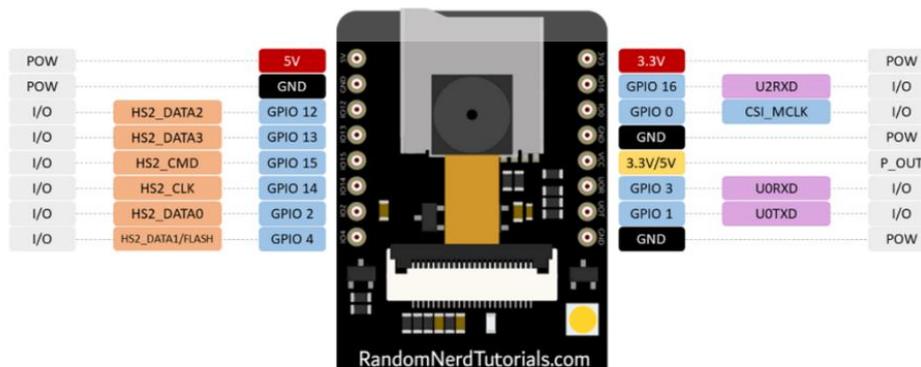
ESP32-CAM adalah mikrokontroler yang memiliki fitur kamera, *Wifi*, *Bluetooth*, dan slot *microSD*. Mikrokontroler ini sering digunakan dalam proyek IoT (Internet of Things) yang memerlukan kemampuan kamera. Modul ESP32-

CAM memiliki jumlah pin I/O yang lebih sedikit dibandingkan dengan modul ESP32 sebelumnya. Ini disebabkan oleh penggunaan banyak pin secara internal untuk fungsi kamera dan slot kartu microSD. Selain itu, modul ESP32-CAM tidak memiliki port USB yang khusus digunakan untuk mengirim program dari komputer melalui USB. Oleh karena itu, untuk memprogram modul ini, perlu menggunakan perangkat USB to TTL (*time to live*) atau dapat dengan menambahkan modul downloader khusus yang dirancang untuk ESP32-CAM.



Gambar 2. 13 ESP32-CAM [13]

Modul ESP32CAM memiliki dua sisi dalam perangkatnya. Di sisi atasnya terdapat modul kamera yang dapat dilepas dan dipasang kembali, microSD yang dapat diisi dengan data, serta lampu flash tambahan yang dapat digunakan sebagai sumber cahaya tambahan untuk kamera bila diperlukan. Di bagian belakang modul, terdapat antena internal, konektor untuk antena eksternal, pin male yang digunakan untuk I/O, dan komponen ESP32S sebagai otak pengolahnya. Spesifikasinya meliputi: Bluetooth 4.2 dengan BLE (Bluetooth Low Energy), Wi-Fi 802.11b/g/n, UART, SPI, I2C, dan PWM *interfaces*, daya komputasi hingga 600 DMIPS, kecepatan clock hingga 160 MHz, RAM sebesar 520 KB ditambah 4 MB



Gambar 2. 14 Pin ESP32-CAM [14]

PSRAM, mendukung unggah gambar melalui Wi-Fi, beberapa mode tidur, kemungkinan pembaruan firmware secara nirkabel (Firmware Over the Air/FOTA), 9 port GPIO (General Purpose Input/Output), LED Flash terintegrasi, dan modul kamera [20].

## 2.7 Sensor Kamera OV2640

Sensor Kamera OV2640 adalah sensor kamera yang umumnya digunakan pada ESP32-CAM yang berfungsi untuk menangkap tampilan gambar dari wajah pengguna. Kamera ini memiliki resolusi 2MP yang membutuhkan tegangan DC sebesar 1.3V. Memiliki ukuran array 1600 x 1200 (UXGA) dengan ukuran pixel 2.2 x 2.2 um dan luas citra 3590 x 2684 um [16].



Gambar 2. 15 Sensor Kamera OV2640 [16]