

BAB II

TINJAUAN PUSTAKA

2.1 Kajian Teori

Penelitian yang sudah ada mengenai sistem absensi menggunakan barcode untuk memantau kehadiran siswa dan aset dilakukan oleh Salah Elaskari dkk pada jurnal “*Using Barcode to Track Student Attendance and Asset in Higher Education Institutions*” tahun 2021. Sistem presensi ini dengan memasang barcode pada kartu identitas atau tanda pengenal mahasiswa. Penelitian ini menunjukkan bahwa teknologi barcode efektif dalam meminimalisir kesalahan penginputan data dan meningkatkan efisiensi dalam biaya penggunaan. Meskipun penelitian ini dianggap efektif penggunaan barcode untuk sistem presensi, namun masih banyak kekurangan dibandingkan dengan teknologi lainnya seperti penggunaan sistem absensi berbasis biometrik atau berbasis kamera. Selain itu, penggunaan barcode dapat memakan waktu yang lama untuk memindai dan memproses data. Teknologi barcode dan RFID juga memiliki batasan dalam hal keamanan karena dapat dengan mudah dipalsukan dan direkayasa ulang[5]

Penelitian mengenai pengembangan sistem absensi tanpa kontak langsung dengan menggunakan kamera yang dilengkapi dengan teknologi anti-spoofing untuk mencegah terjadinya tindakan pemalsuan atau penipuan dalam merekam kehadiran oleh Deepti saraswati, dkk pada jurnal “*Anti-spoofing Enabled Contactless Attendance Monitoring Sistem in the COVID-19 pandemic*” tahun 2023. Sistem ini mengatasi keterbatasan sistem absensi yang memungkinkan satu entri oleh beberapa orang yang mengikuti pelajaran atau kegiatan yang sama. Sistem ini menggunakan google vision API untuk mendeteksi wajah dan mekanisme anti-

spoofing untuk mencegah aktivitas curang. Server memproses dan melatih data, mencatat informasi individu dalam database, dan menandai kehadiran. Hasil pengujian menunjukkan bahwa sistem yang diusulkan memiliki akurasi 95,85% dan mengurangi biaya penyimpanan sebesar 33,52% melalui database firebase, tetapi penelitian ini belum dilengkapi dengan integrasi fitur notifikasi kehadiran secara real-time selain database[10].

Penelitian yang sudah dilakukan dalam sistem presensi dengan pengenalan wajah oleh Arif Ainur, dkk pada tahun 2022 "*Employee Presence Using Body Temperature Detection and Face Recognition*" membahas penggunaan teknologi deteksi suhu tubuh dan pengenalan wajah untuk melacak kehadiran dan keberadaan karyawan ditempat kerja. Sistem pengenalan wajah yang dapat mendeteksi dan mengenali wajah secara real-time menggunakan pendekatan haar cascade *classifier*. Penelitian tersebut memiliki keterbatasan pada kondisi cahaya yang rendah, perubahan pose, dan variasi lainnya dan memerlukan training data yang cukup besar dan representatif untuk dapat mengenali wajah dengan tepat[11]. Maka perlu dikembangkan untuk meningkatkan kinerja pengenalan wajah pada teknologi saat ini.

Selanjutnya penelitian mengenai penggunaan teknologi pengenalan wajah untuk memantau kehadiran siswa dikelas selanjutnya dari andre dkk yang berjudul "*Student Attendance with Face recognition (LBPH or CNN) Sitematic Literature Review*" tahun 2023 membahas implementasi teknologi pengenalan wajah menggunakan algoritma LBPH atau *Convolutional Neural Network (CNN)*. Hasil dari studi ini menunjukkan bahwa teknologi pengenalan wajah dapat digunakan sebagai alternatif metode pemantauan kehadiran siswa yang lebih akurat dan

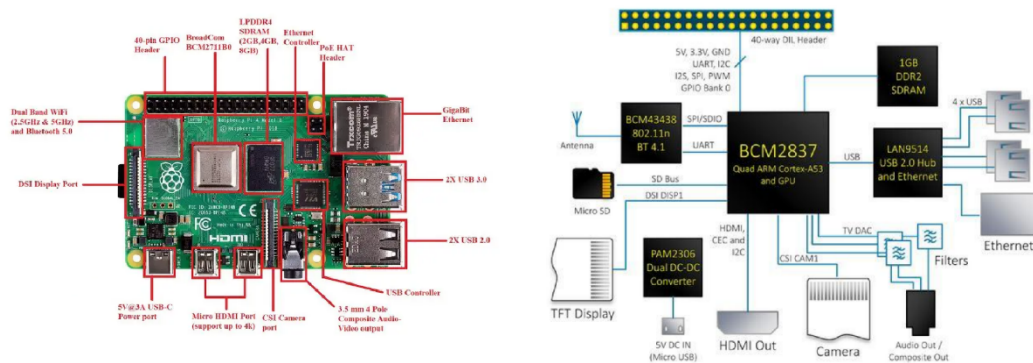
efisien. Akurasi yang didapatkan untuk *Face Recognition* pada sistem absensi kelas dari beberapa makalah adalah untuk LBPH mendapatkan akurasi 77,55%, Eigenfaces mendapatkan akurasi 85%, PCA mendapatkan akurasi 80%, Haar Cascade mendapatkan akurasi 92,9%, dan CNN mendapatkan akurasi 95%[8]. Namun dalam penelitian tersebut belum dilengkapi dengan fitur notifikasi keamanan untuk mendapatkan informasi secara *real-time*.

Maka dalam penelitian ini melakukan pengembangan teknologi dari penelitian sebelumnya, sistem presensi menggunakan pengenalan wajah (*face recognition*) berbasis *INTERNET OF THINGS* di SMK Cibening. Metode yang digunakan dalam sistem presensi menggunakan metode *deep learning*. *Deep learning* dapat digunakan untuk membangun model pengenalan wajah yang dapat mempelajari ciri-ciri wajah yang lebih kompleks daripada metode konvensional seperti Eigenfaces atau haar cascade. CNN merupakan salah satu cara penerapan *deep learning* yang memberikan hasil yang baik karena memiliki banyak layer yang bisa mendapatkan fitur terbaik dari sebuah gambar melalui proses pembelajaran, sehingga tidak perlu memasukkan fitur tersebut secara manual untuk mengekstraksi fitur dari input data yaitu gambar. Dengan mempelajari karakteristik gambar menggunakan kotak kecil dari gambar input, CNN mempertahankan distribusi spasial antar piksel, seperti rotasi, translasi, kompresi, dan transformasi geometris. Semakin banyak dataset yang digunakan maka semakin besar pula tingkat pembelajaran yang dilakukan oleh algoritma tersebut sehingga akurasi didapatkan juga dapat bertambah. Hasil pengenalan wajah dapat diintegrasikan dengan sistem absensi seperti mencatat kehadiran guru dan siswa dan mengirim notifikasi ke orang tua siswa melalui aplikasi Telegram. Hanya kepala sekolah, wali kelas dan staff tata

usaha yang dapat mengakses sistem tersebut, sehingga data absensi menjadi lebih terjaga keamanannya.

2.2 Raspberry pi

Raspberry Pi merupakan komputer yang memiliki ukuran kecil. Spesifikasi Raspberry Pi terdiri atas: prosesor, memori 1062 MB untuk tipe B, memakai SD-Card sebagai pengganti hardisk, dan memakai daya 2,5W serta berdimensi 3,37” x 2,21” x 0.83” dan memiliki berat 45gr [8]. Berikut **Gambar 2.1** merupakan gambar fisik Raspberry pi



Gambar 2.1 Fisik Raspberry pi

Perangkat ini digunakan untuk mengelola data gambar yang diterima oleh kamera melalui sambungan USB dan memproses mengidentifikasi nama kemudian mengirimkan data nama dan jam ke server *cloud* menggunakan API yang telah disediakan untuk dilakukan presensi. Selain itu perangkat ini digunakan sebagai pemroses data tulisan yang akan di cetak pada layar *display* LCD 16X2, memproses trigger inputan dari tombol presensi yang di tekan untuk memulai presensi. Adapun spesifikasi yang digunakan adalah dengan RAM 4GB dan Storage 64GB.

2.3 Kamera Webcam

Webcam adalah sebuah kamera video yang menyuplai gambar *real-time* ke sebuah jaringan komputer melalui usb, Ethernet, atau Wi-Fi [8]. Berikut **Gambar 2.2** merupakan gambaran fisik dari kamera webcam

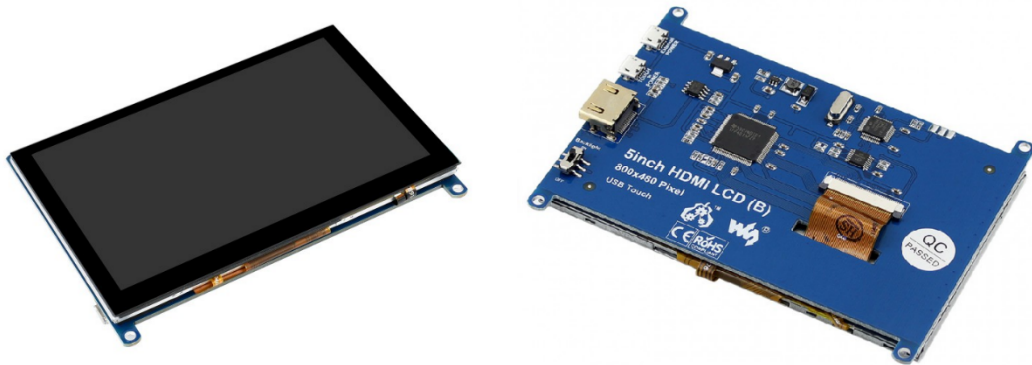


Gambar 2.2 Kamera Webcam

Kamera yang digunakan adalah model Logitech C230, Kamera ini digunakan untuk mengambil data gambar atau video yang diperlukan dalam proses pengenalan wajah atau pemrosesan citra. Pada kasus ini, kamera akan mengambil gambar wajah pengguna untuk kemudian diproses dan diidentifikasi.

2.4 Display 5 Inch Touch Screen LCD

Layar (*Liquid Crystal Display*) LCD 5 inci dengan layar sentuh untuk Raspberry Pi adalah perangkat tambahan yang dapat dihubungkan melalui input HDMI ke Raspberry Pi. Dengan layar sentuh, pengguna dapat berinteraksi secara langsung dengan Raspberry Pi, memudahkan operasi dan proyek elektronik. Layar ini menawarkan tampilan visual yang jelas dan cukup besar untuk berbagai aplikasi, membuatnya menjadi tambahan yang bermanfaat untuk meningkatkan pengalaman pengguna dengan Raspberry Pi [8]. Berikut **Gambar 2.3** merupakan gambaran fisik *Display 5 Inch Touch Screen*



Gambar 2.3 *Display Trough Screen 5 Inch*

2.5 Face Recognition

Face Recognition adalah salah satu teknik identifikasi teknologi biometrik dengan menggunakan wajah individu yang bersangkutan sebagai parameter utamanya. Pendekatan yang umum digunakan dalam *Face Recognition* adalah dengan menggunakan analisis pola-pola wajah, seperti bentuk wajah, posisi mata, hidung, mulut, dan fitur lainnya. Teknologi ini telah mengalami perkembangan signifikan dan digunakan dalam berbagai aplikasi seperti keamanan, manajemen kehadiran, identifikasi pribadi, dan pengawasan publik [11].

1. Metode Pengenalan Wajah:
 - a. Pengenalan Berdasarkan Fitur (*Feature-based Recognition*): Teknik ini melibatkan ekstraksi fitur-fitur kunci dari wajah, seperti bentuk, ukuran mata, hidung, mulut, dan rasio antarfitur. Fitur-fitur ini kemudian digunakan untuk membangun sebuah representasi numerik dari wajah yang dapat dibandingkan dengan data referensi.
 - b. Pengenalan Berdasarkan Pola (*Pattern-based Recognition*): Metode ini mengandalkan pada pengenalan pola global pada wajah secara

keseluruhan. Data gambar wajah yang ada dianalisis untuk mengenali pola-pola khas, seperti hubungan antara mata, hidung, dan mulut, serta distribusi warna kulit.

2. Proses Pengenalan:

- a. Pencocokan Wajah (*Matching*): Setelah fitur atau pola wajah diekstraksi, sistem *Face Recognition* akan mencocokkannya dengan data referensi yang sudah ada dalam basis data. Ini bisa melibatkan perhitungan perbedaan antara fitur-fitur atau perbandingan pola.
- b. *Training dan Learning*: Banyak sistem *Face Recognition* menggunakan teknik pembelajaran mesin (*machine learning*) untuk meningkatkan kualitas pengenalan. Dalam proses pelatihan, model diajarkan untuk mengenali pola-pola wajah dengan membandingkan input dengan label yang benar.

3. Keuntungan:

- a. Non-Intrusif: Teknologi ini tidak memerlukan interaksi fisik, seperti sidik jari atau iris mata, sehingga pengenalan dapat dilakukan tanpa kesadaran individu.
- b. Cepat dan Tidak Memerlukan Kontak Langsung: Proses pengenalan wajah dapat dilakukan dengan cepat dan tanpa perlu kontak langsung dengan perangkat, seperti pada pengenalan sidik jari.
- c. Digunakan dalam Berbagai Aplikasi: Selain keamanan, *Face Recognition* juga digunakan dalam manajemen kehadiran karyawan, identifikasi pribadi pada perangkat pintar, pengawasan peristiwa publik, dan bahkan pengenalan emosi pada wajah.

4. Tantangan dan Kepentingan Privasi:
 - a. Variabilitas Wajah: Wajah individu dapat berubah karena perubahan ekspresi, usia, cahaya, sudut pandang, dan lain-lain, yang dapat mempengaruhi akurasi pengenalan.
 - b. Kepentingan Privasi: Penggunaan teknologi ini telah menimbulkan keprihatinan terkait privasi dan penggunaan data wajah. Perlindungan data dan regulasi yang ketat diperlukan untuk memastikan informasi pribadi tidak disalahgunakan.
5. Pengembangan Terbaru:
 - a. Pengenalan 3D: Teknologi pengenalan wajah 3D telah berkembang, memungkinkan sistem untuk mengenali wajah bahkan dalam variasi sudut pandang yang lebih besar.
 - b. Kombinasi dengan Teknologi Lain: Beberapa sistem menggunakan *Face Recognition* bersama teknologi lain seperti pengenalan suara atau sidik jari untuk meningkatkan akurasi identifikasi.

2.6 Deep learning

Deep learning adalah subbidang dari machine learning yang berfokus pada pengembangan dan penerapan model neural network yang mendalam. *Deep learning* menggunakan arsitektur neural network yang memiliki banyak lapisan (layer) untuk mempelajari representasi fitur yang kompleks secara otomatis dari data mentah. Tujuan utama dari *deep learning* adalah untuk menghasilkan model yang mampu mempelajari dan mengekstraksi fitur yang semakin abstrak dan tingkat representasi yang semakin tinggi dari data yang diberikan [10]. Dalam *deep*

learning, neural network yang digunakan biasanya terdiri dari tiga jenis lapisan utama, yaitu:

1. Lapisan Input (*Input Layer*): Lapisan input adalah titik awal dari neural network. Ini menerima data mentah atau input dari dataset yang akan diproses oleh jaringan. Jumlah neuron dalam lapisan input sesuai dengan dimensi data input. Misalnya, dalam pengenalan citra, setiap neuron dalam lapisan input mungkin mewakili piksel-piksel citra atau fitur-fitur awal dari data input.
2. Lapisan Tersembunyi (*Hidden Layer*): Lapisan tersembunyi adalah tempat dimana sebagian besar komputasi dan pembelajaran terjadi dalam *deep learning*. Setiap lapisan tersembunyi terdiri dari banyak neuron yang saling terhubung. Mereka bekerja sama untuk mempelajari representasi-fitur yang semakin kompleks dari data. Setiap neuron dalam lapisan tersembunyi menghitung bobot terponderansi dari inputnya dan meneruskannya melalui fungsi aktivasi, yang dapat berupa fungsi sigmoid, ReLU (*Rectified Linear Unit*), atau fungsi aktivasi lainnya.
3. Lapisan Output (*Output Layer*): Lapisan output menghasilkan hasil akhir dari model *deep learning*. *Output* dari lapisan ini tergantung pada jenis tugas yang ingin diselesaikan. Misalnya, dalam tugas klasifikasi, lapisan *output* akan menghasilkan probabilitas atau prediksi kelas tertentu. Dalam tugas regresi, lapisan *output* dapat menghasilkan nilai numerik. Lapisan *output* juga bisa berjumlah lebih dari satu neuron, tergantung pada jumlah kelas yang ingin dihasilkan.

Penting untuk diingat bahwa dalam sebuah model *deep learning*, informasi mengalir dari lapisan input melalui lapisan-lapisan tersembunyi dan akhirnya

mencapai lapisan *output*. Selama proses ini, parameter atau bobot jaringan disesuaikan secara iteratif melalui algoritma pembelajaran seperti "backpropagation" untuk mengoptimalkan performa model terhadap tujuan tertentu.

Kekuatan utama dari *deep learning* adalah kemampuannya untuk secara otomatis mengekstraksi fitur yang semakin abstrak dan representasi yang semakin tinggi dari data. Dengan menggunakan banyak lapisan tersembunyi, neural network dapat belajar untuk mengenali pola yang kompleks dan sulit didefinisikan secara manual. Hal ini membuat *deep learning* sangat efektif dalam berbagai tugas seperti pengenalan gambar, bahasa alami, pengolahan bahasa alami, dan sebagainya.

2.7 CNN

Convolutional Neural Network (CNN) adalah jenis arsitektur *neural network* yang dirancang khusus untuk pengolahan data berstruktur grid, terutama dalam konteks pengolahan citra dan pengenalan pola visual. CNN telah mencatat kesuksesan yang signifikan dalam berbagai tugas pengolahan citra, termasuk pengenalan wajah, deteksi objek, pengenalan tulisan tangan, dan segmentasi citra. Pada dasarnya, CNN terdiri dari beberapa jenis lapisan yang bekerja secara bersama-sama untuk mengenali pola dan fitur pada citra [8].

Lapisan Ekstraksi Fitur merupakan salah satu komponen kunci dalam arsitektur *Convolutional Neural Network* (CNN). Tujuan utamanya adalah untuk mengenali dan mengekstraksi fitur-fitur penting yang relevan dari data input, seperti pola, tekstur, dan kontur. Lapisan ini biasanya terdiri dari beberapa lapisan konvolusi dan lapisan aktivasi, yang bekerja bersama-sama untuk menghasilkan representasi-fitur yang semakin kompleks dan semakin abstrak.

Dalam lapisan ekstraksi fitur, setiap lapisan konvolusi akan menerapkan filter atau kernel pada data input, yang menghasilkan peta fitur yang menyoroti kehadiran fitur-fitur tersebut. Filter ini mampu menangkap pola-pola visual yang lebih sederhana, seperti garis dan tepi. Lapisan aktivasi, seperti *Rectified Linear Unit* (ReLU), digunakan untuk memberikan respons yang kuat pada fitur yang diaktifkan dan menekan respons yang lemah atau nol.

Tujuan dari lapisan ekstraksi fitur adalah untuk memungkinkan model untuk secara bertahap memahami fitur-fitur yang semakin kompleks dari data input. Dalam konteks pengenalan gambar, lapisan ini dapat mengenali fitur-fitur seperti sudut, bentuk, atau tekstur. Proses ekstraksi fitur ini secara hierarkis membantu model untuk memahami pola-pola yang semakin kompleks dalam data dan menciptakan representasi-fitur yang lebih abstrak.

Hasil dari lapisan ekstraksi fitur akan menjadi input bagi lapisan berikutnya dalam CNN, seperti lapisan *pooling* atau lapisan *fully-connected*. Dengan mengenali fitur-fitur penting, lapisan ekstraksi fitur mempersiapkan data untuk pemahaman lebih lanjut dan pengambilan keputusan akhir oleh model, seperti klasifikasi objek atau pengenalan pola tertentu. Berikut adalah komponen utama dalam arsitektur CNN:

1. Lapisan Konvolusi (*Convolutional Layer*): Lapisan konvolusi adalah inti dari CNN. Pada lapisan ini, terdapat sekumpulan filter konvolusi yang digunakan untuk melakukan operasi konvolusi pada citra input. Setiap filter bergerak secara berulang kali pada seluruh citra input untuk menghasilkan peta fitur (*feature map*) yang menyoroti pola dan fitur-fitur penting. Proses ini

memungkinkan jaringan untuk menangkap fitur lokal yang relevan seperti tepi, garis, sudut, dan tekstur pada citra.

Di dalam matematika, konvolusi adalah sebuah operasi matematika dari dua fungsi f dan g yang menghasilkan fungsi ketiga h . Jenis konvolusi terdapat dua jenis yaitu konvolusi kontinu dan konvolusi diskrit. Didalam pengolahan citra diberlakukan pada domain spasial sehingga konvolusi yang 13 digunakan adalah jenis konvolusi diskrit dengan operasinya secara umum didefinisikan sebagai berikut:

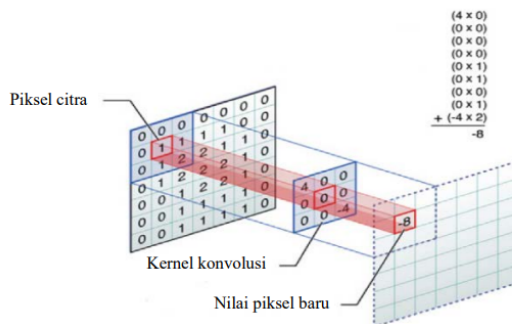
$$\begin{aligned} h(x, y) &:= (f * g)(x, y) \\ &:= \sum_{a=-\infty}^{\infty} \sum_{b=-\infty}^{\infty} f(a, b)g(x - a, y - b) \end{aligned} \dots\dots\dots (1)$$

Fungsi $f(x, y)$ adalah sebuah fungsi yang direpresentasikan sebagai citra dan fungsi $g(x, y)$ adalah kernel konvolusi atau kernel penapis (filter). Kernel $g(x, y)$ merupakan suatu jendela yang dioperasikan secara bergeser pada sinyal masukan $f(x, y)$, yang dalam hal ini, jumlah perkalian kedua fungsi pada setiap titik merupakan hasil konvolusi dinyatakan sebagai keluaran fungsi $h(x)$. Oleh karena ukuran citra terbatas pada ukuran $M \times N$ dengan kernel berukuran $m \times n$ maka operasi fungsi konvolusi didefinisikan sebagai berikut;

$$(f * g)(x, y) := \sum_{a=x-h}^{x+h} \sum_{b=y-w}^{y+w} f(a, b)g(x - a, y - b) \dots\dots\dots (2)$$

Dimana $m = 2h + 1$ adalah tinggi kernel dan $n = 2w + 1$ adalah lebar kernel. Operasi dilakukan dengan melakukan kombinasi linear dari mengambil bagian input citra yang sama dengan kernel dan nilai hasil operasi disimpan berupa elemen nilai matriks kemudian dilanjutkan dengan menggeser kernel

piksel per piksel sampai terhimpun keseluruhan nilai piksel baru. Proses konvolusi disajikan pada **Gambar 2.4** dengan citra berukuran $M = 7$ $N = 7$ dan kernel berukuran $m = 3$ $n = 3$. Operasi pertama dilakukan dengan mengalikan bagian dari elemen citra pertama berukuran 3x3 (sesuai dengan ukuran kernel) dengan tiap elemen kernel yang bersesuaian kemudian dilakukan penjumlahan pada akhir perhitungannya.



Gambar 2.4 Proses operasi konvolusi pada satu lokasi pada suatu citra

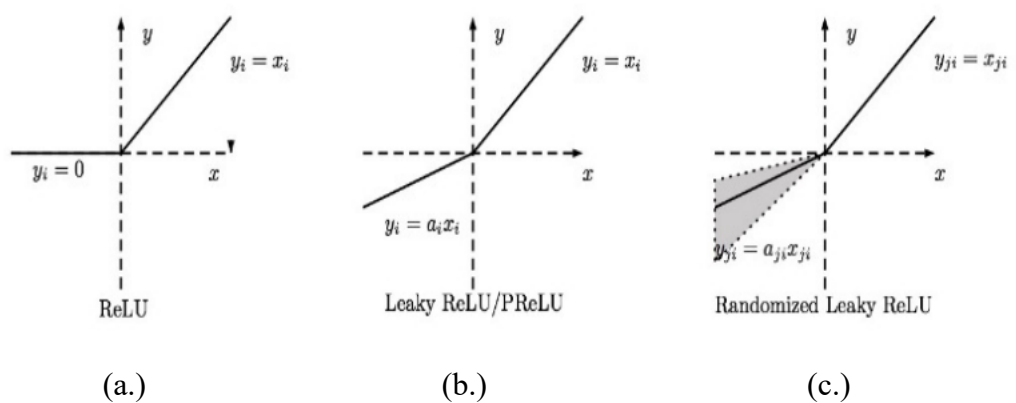
2. Lapisan ReLU (*Rectified Linear Unit*): Lapisan ReLU mengikuti lapisan konvolusi dan bertujuan untuk memperkenalkan non-linearitas dalam jaringan. Setiap piksel dalam peta fitur yang dihasilkan oleh lapisan konvolusi akan diaktivasi menggunakan fungsi ReLU, yaitu mengubah semua nilai negatif menjadi nol dan mempertahankan nilai positif. Definisi persamaan fungsi (Rectifiers Linear Unit) ReLU adalah sebagai berikut

$$f(x_i) = \begin{cases} x_i, & x_i \geq \epsilon \\ 0, & x_i < \epsilon \end{cases} \dots\dots\dots (3)$$

Dimana x_m adalah input dari fungsi aktivasi non-linier pada channel ke- I . Fungsi ReLU digunakan pada tahap forward pass CNN adalah $y = \text{Max}(\epsilon, x)$. Sedangkan untuk tahap backward pass digunakan fungsi turanannya yaitu:

$$f(x_i) = \begin{cases} 1, & x_i \geq \epsilon \\ 0, & x_i < \epsilon \end{cases} \dots\dots\dots (4)$$

Walaupun ReLU tidak terdiferensiasi di ϵ . Tetapi mempunyai subdiferensial pada selang $[0,1]$. Sembarang nilai pada subinterval dapat diterima sebagai subdiferensiasi sehingga dapat digunakan dalam SGD jika digeneralisasi dari gradient descent.

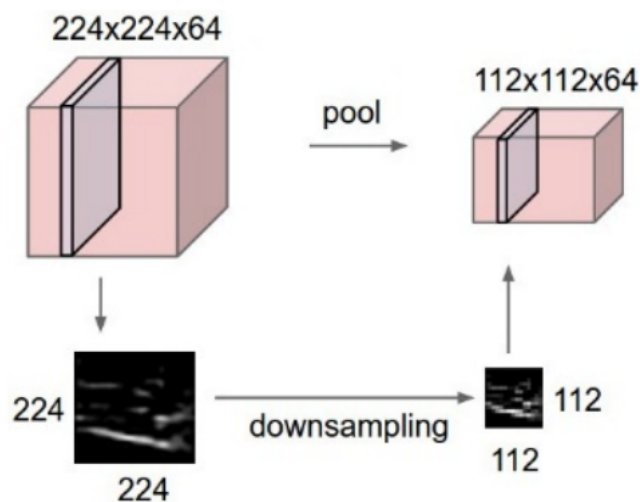


Gambar 2.5 Grafik fungsi ReLU(a.), fungsi LReLU(b.), dan fungsi PReLU(c.)

3. **Stride**: Stride, yang mengukur seberapa jauh filter digeser saat menggerakkan melintasi citra input, memainkan peran penting dalam mengatur resolusi spatia dan dimensi peta fitur yang dihasilkan. Stride yang lebih besar mengakibatkan pengurangan dimensi peta fitur, sementara stride yang lebih kecil mempertahankan informasi spasial secara lebih baik. Penggunaan stride bertujuan untuk mengontrol sejauh mana resolusi spatia dan ukuran output peta fitur yang diinginkan. Melalui penyesuaian stride yang tepat, CNN dapat mempercepat komputasi serta mengurangi dimensi data tanpa merugikan informasi yang penting.
4. **Padding**: Padding melibatkan penambahan piksel-piksel nol atau nilai yang ditentukan di sekitar tepi citra input sebelum operasi konvolusi dilakukan. Ini

bertujuan untuk menjaga integritas informasi tepi dalam data. Tujuan penggunaan padding adalah untuk mencegah penyusutan yang berlebihan dalam dimensi data selama operasi konvolusi dan untuk mempertahankan informasi tepi yang penting. Dengan demikian, padding membantu CNN mengatasi masalah seperti kehilangan informasi di tepi citra serta mempertahankan ciri khas fitur yang ada.

5. Lapisan *Pooling* (*Pooling Layer*): Lapisan *Pooling* memiliki tujuan utama untuk mengurangi dimensi spasial dari peta fitur yang dihasilkan oleh lapisan konvolusi sebelumnya. Proses ini dilakukan dengan mengambil nilai maksimum atau rata-rata dari suatu area tertentu pada peta fitur. Salah satu metode *pooling* yang umum digunakan adalah *Max pooling*, di mana nilai maksimum diambil dari area yang ditentukan. Contoh lainnya adalah *average pooling*, di mana nilai rata-rata diambil dari area yang sama.

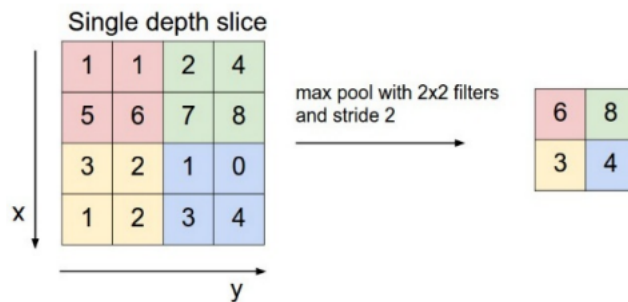


Gambar 2.6 Representasi *Pooling Layer*

Pooling layer bekerja dengan blok spasial yang bergerak sepanjang ukuran *feature pattern*. Ukuran pergeseran blok pada umumnya adalah ukuran pada dimensi blok ($H \times H$) itu sendiri sehingga tidak ada *overlapping* seperti pada

Convolutional Layer. Pergerakan blok diikuti dengan perhitungan *pooling* pada masukan pola fitur (u).

$$u_{i,j}^{(k)} = \max_{p,q \in P_{i,j}} z_{p,q}^{(k)} \dots\dots\dots (5)$$



Gambar 2.7 Perhitungan *Max pooling* dengan stride 2

Dalam *Max pooling*, untuk setiap area dalam peta fitur, hanya nilai maksimum yang diteruskan ke lapisan berikutnya, sementara informasi lain diabaikan. Ini membantu dalam mengurangi ukuran data dengan mereduksi resolusi spasial, sehingga memudahkan komputasi dan mengurangi kompleksitas model. Selain itu, *Max pooling* juga memiliki efek positif dalam mengurangi *overfitting*, karena mengurangi jumlah parameter dan menjadikan model lebih tahan terhadap variasi kecil dalam data. Tujuan utama dari lapisan *pooling* adalah untuk:

- a. Mengurangi Dimensi: Dengan mereduksi ukuran data, lapisan *pooling* membantu memangkas beban komputasi pada lapisan berikutnya.
- b. Mengurangi *Overfitting*: Dengan mengurangi jumlah parameter dan mempertahankan fitur penting, lapisan *pooling* membantu menghindari model yang terlalu kompleks dan mampu umum dalam mengenali pola.

- c. Mempertahankan Informasi Relevan: Walaupun mengurangi dimensi, lapisan *pooling* masih mempertahankan informasi yang dianggap penting, seperti keberadaan fitur penting dalam suatu area.

Dalam arsitektur CNN, lapisan *pooling* biasanya diaplikasikan setelah beberapa lapisan konvolusi. Ini membantu dalam hierarki ekstraksi fitur yang semakin kompleks dan abstrak, serta menghasilkan representasi-fitur yang lebih sederhana dan kompak sebelum dihubungkan ke lapisan-lapisan *fully-connected* untuk tujuan klasifikasi atau tugas lainnya.

6. Lapisan *Fully Connected (Fully Connected Layer)*: Setelah lapisan konvolusi dan *pooling*, representasi fitur yang dihasilkan diubah menjadi vektor dan diteruskan ke dalam lapisan *fully connected*. Lapisan *fully connected* mirip dengan lapisan neural network konvensional, di mana setiap neuron terhubung dengan setiap neuron pada lapisan sebelumnya dan setelahnya. Lapisan ini bertanggung jawab untuk mengklasifikasikan dan memproses informasi fitur yang lebih tingkat abstrak.

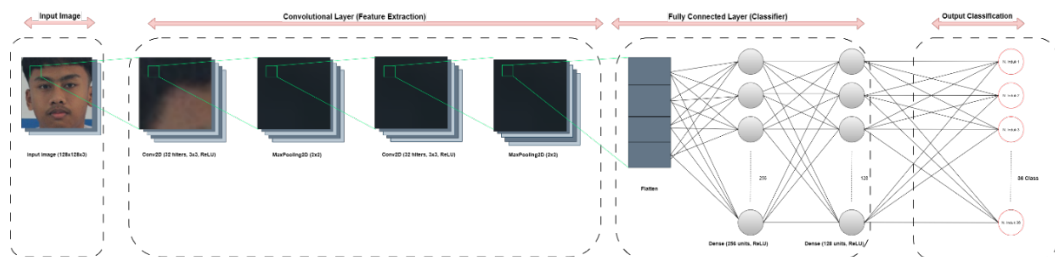
Ada beberapa pustaka (*libraries*) dalam bahasa pemrograman Python yang dapat digunakan untuk melakukan *Face Recognition*, *deep learning*, dan pengimplementasian CNN:

1. OpenCV (*OpenSource Computer Vision Library*): OpenCV adalah pustaka populer dalam dunia pengolahan citra dan visi komputer. Ini memiliki modul khusus untuk *Face Recognition* yang mencakup algoritma deteksi wajah dan fitur wajah, serta dukungan untuk melatih model pengenalan wajah. Selain

itu, OpenCV juga memiliki fitur yang mendukung pengolahan citra secara umum, yang sangat berguna dalam pra-pemrosesan data sebelum digunakan dalam model *deep learning*.

2. TensorFlow: TensorFlow adalah pustaka yang dikembangkan oleh Google untuk komputasi numerik dan *deep learning*. Ini memiliki antarmuka khusus untuk membangun dan melatih model CNN serta arsitektur *deep learning* lainnya. TensorFlow memiliki fitur yang kuat untuk pemrosesan tensor dan algoritma pelatihan yang canggih. TensorFlow juga menyediakan keras sebagai antarmuka tingkat tinggi untuk membangun dan melatih model *deep learning* dengan lebih mudah.
3. PyTorch: PyTorch adalah pustaka *opensource* yang juga populer untuk *deep learning*. Ini memiliki pendekatan yang lebih dinamis untuk definisi dan pelatihan model, membuatnya lebih mudah untuk eksperimen. PyTorch juga menyediakan modul torchvision yang berisi dataset populer dan arsitektur model, termasuk untuk pengenalan gambar dan *Face Recognition*.
4. dlib: dlib adalah pustaka yang memiliki banyak fitur dalam bidang visi komputer, termasuk pengenalan wajah. Ini memiliki algoritma yang kuat untuk deteksi wajah dan ekstraksi fitur wajah yang berguna dalam membangun model pengenalan wajah.
5. scikit-learn: scikit-learn adalah pustaka yang luas digunakan untuk *machine learning* di Python. Meskipun lebih fokus pada algoritma *machine learning* tradisional, scikit-learn memiliki beberapa fitur untuk ekstraksi fitur dan pengolahan data yang bermanfaat dalam persiapan data sebelum menggunakan model *deep learning*.

Selama proses pelatihan (*training*), CNN menggunakan algoritma backpropagation untuk mengoptimalkan bobot dan bias pada setiap lapisan. Ini dilakukan dengan meminimalkan fungsi kerugian (*loss function*) yang membandingkan prediksi CNN dengan label yang sesungguhnya pada data latih. Proses ini memungkinkan CNN untuk belajar secara mandiri dan menghasilkan representasi-fitur yang semakin baik dan akurat seiring dengan bertambahnya jumlah iterasi pelatihan. Pada **Gambar 2.8** merupakan desain diagram CNN yang digunakan pada penelitian ini.



Gambar 2.8 Model CNN yang digunakan

2.8 Model Warna RGB

RGB (Red, Green, Blue) dan *Grayscale* adalah dua model warna yang umum digunakan dalam pengolahan gambar dan grafika komputer. Model warna RGB merupakan pilihan utama dalam representasi warna di layar komputer dan perangkat digital lainnya. Dalam model ini, warna diciptakan dengan menggabungkan tiga warna dasar: merah (Red), hijau (Green), dan biru (Blue) dalam berbagai proporsi. Setiap piksel dalam gambar RGB memiliki tiga saluran warna: satu untuk merah, satu untuk hijau, dan satu untuk biru, dengan intensitas yang berkisar dari 0 hingga 255. Kombinasi intensitas saluran ini memungkinkan pembuatan berbagai warna, termasuk yang terang dan beragam.

Di sisi lain, model *Grayscale* mengandalkan tingkat kecerahan untuk merepresentasikan gambar. Dalam model ini, setiap piksel hanya memiliki satu saluran warna yang mencerminkan intensitas kecerahan atau tingkat abu-abu pada skala dari hitam hingga putih. *Grayscale* berguna ketika Anda hanya perlu mewakili gambar dalam skala keabuan tanpa memerlukan warna, seperti dalam mencetak dokumen, pemrosesan gambar medis, atau analisis citra berdasarkan tingkat kecerahan.

Merubah citra RGB menjadi citra *grayscale* adalah salah satu contoh proses pengolahan citra menggunakan operasi titik. Untuk mengubah citra RGB menjadi citra *grayscale* adalah dengan menghitung rata-rata nilai intensitas RGB dari setiap piksel penyusun citra tersebut. Rumus matematis yang digunakan adalah sebagai berikut:

$$f_0(x, y) = \frac{f_i^R(x, y) + f_i^G(x, y) + f_i^B(x, y)}{3} \dots\dots\dots (6)$$

Memiliki tiga komponen warna dalam model RGB, yaitu merah (R), hijau (G), dan biru (B). Untuk menghasilkan citra *grayscale*, kita mengambil rata-rata dari nilai intensitas ketiga komponen warna ini untuk setiap piksel. Hasil rata-rata ini menjadi nilai kecerahan piksel di citra *grayscale*. citra *grayscale* hanya memerlukan tingkat kecerahan atau abu-abu dari warna, dan rata-rata dari ketiga komponen warna memberikan representasi yang cukup baik tentang tingkat kecerahan tersebut [8].