

BAB II

TINJAUAN PUSTAKA

2.1 Identifikasi

Identifikasi menurut Kamus Besar Bahasa Indonesia (KBBI) adalah tindakan mengenali atau memberikan bukti identitas seseorang, benda, atau badan lain. Identifikasi berasal dari istilah “identifikasi”, yang mengacu pada proses penyelidikan atau pemeriksaan persyaratan yang ada dalam domain tertentu. Menurut Wikipedia, identifikasi dapat digolongkan menjadi dua jenis yaitu kebutuhan mendesak dan kebutuhan tidak mendesak (Ilahiyah & Nilogiri, 2018).

Identifikasi yang disimpulkan, sebagaimana didefinisikan, mengacu pada prosedur penelitian yang dilakukan oleh seorang individu untuk mengumpulkan data yang berharga bagi orang lain.

2.2 Umur

Menurut Kementerian Kesehatan Indonesia, usia adalah metrik temporal yang mengukur durasi keberadaan suatu entitas, yang mencakup makhluk hidup dan mati. Misalnya, usia manusia sering kali ditentukan dengan mengukur waktu yang telah berlalu sejak ia lahir hingga usianya dihitung, yang biasanya dinyatakan dalam lima belas tahun. Jadi, usia diukur sebagai durasi dari saat lahir hingga saat ini.

Usia mengacu pada durasi tertentu dalam kehidupan seseorang yang dapat diukur berdasarkan pertumbuhan kronologis, anatomi, dan fisiologis. Usia diartikan sebagai lamanya waktu yang telah berlalu sejak seseorang dilahirkan (S. Rahmanto, 2019).

2.1.1 Klasifikasi Umur

Kementerian Kesehatan RI mengklasifikasikan usia manusia ke dalam berbagai kategori, antara lain sebagai berikut:

1. Berusia 0 sampai 5 tahun merupakan masa balita.
2. Usia 5 sampai dengan 11 tahun merupakan masa kanak-kanak.
3. Usia 12 sampai dengan 16 tahun merupakan masa remaja awal.

4. Usia 17 sampai dengan 25 tahun merupakan masa remaja akhir.
5. Usia 26 sampai dengan 35 tahun merupakan masa dewasa awal.
6. Usia 36 sampai dengan 45 tahun merupakan masa dewasa akhir.
7. Usia 46 sampai dengan 55 tahun merupakan masa lansia awal.
8. Usia 56 sampai dengan 65 tahun merupakan masa lansia akhir.
9. Seseorang dengan usia 65 tahun keatas akan dikategorikan manula.

2.1.3 Umur dan Wajah

Hubungan usia dapat diamati melalui pengenalan wajah pada manusia. Identifikasi wajah atau yang sering disebut dengan *face recognition* merupakan suatu kegiatan yang lumrah dalam kehidupan manusia sehari-hari. Pengenalan wajah merupakan salah satu cabang dari *computer vision* yang melibatkan analisis citra digital untuk mengidentifikasi wajah seseorang (Budi et al., 2018).

Perkembangan manusia dikategorikan menjadi empat tahap usia: anak-anak, remaja, dewasa, dan lanjut usia. Seiring bertambahnya usia, wajah manusia mengalami transformasi yang signifikan, termasuk bertambahnya kerutan, perubahan struktur tulang pipi, dan perubahan jarak antar fitur utama wajah seperti mata, hidung, dan mulut (Melangi, 2020).

2.2 Manusia

Manusia dianggap makhluk sempurna karena bentuk fisiknya yang estetik dan memiliki jiwa atau roh. Kesempurnaan seorang manusia dapat dilihat melalui keterpaduan yang harmonis antara aspek jasmani dan rohani dalam tubuh manusia. Sebagai contoh, mari kita periksa kombinasi aspek fisik dan metafisik melalui kerja panca indera pada kulit. Mencubit kulit menimbulkan rasa sakit, yang merupakan pengalaman sensorik dan emosional. Setelah diperiksa lebih dekat, ada baiknya mempertimbangkan apakah tubuh atau jiwalah yang mengalami penyakit. Manusia yang telah meninggal dapat memperlihatkan tubuh tak bernyawa tanpa jiwa. Jika seseorang yang meninggal dicubit, ia tidak akan merasakan sensasi apa pun akibat mati rasa pada organ tubuhnya. Demikian pula, sifat roh yang tidak berwujud menjadikannya kebal terhadap sensasi fisik seperti dicubit, karena ia tidak berwujud dan tidak terlihat oleh mata manusia. Namun Allah SWT memberikan penderitaan kepada manusia ketika mereka mengalami

ketidaknyamanan fisik. Sensasi nyeri pada panca indera kulit bila diremas merupakan akibat dari menyatunya aspek jasmani dan rohani dalam tubuh manusia.

Menurut keyakinan Islam, manusia dianggap sebagai wakil Tuhan di muka bumi, yang diberi tanggung jawab memelihara, mengembangkan, dan memajukan dunia sesuai dengan bimbingan Allah Ta'ala. Dalam Al-Qur'an, khususnya surat al-An'am ayat 165 disebutkan *"Dan Dia lah yang menjadikan kamu penguasa-penguasa di bumi dan Dia meninggikan sebahagian kamu atas sebahagian (yang lain) beberapa derajat, untuk mengujimu tentang apa yang diberikan-Nya kepadamu. Sesungguhnya Tuhanmu amat cepat siksaan-Nya dan sesungguhnya Dia Maha Pengampun lagi Maha Penyayang"*. (Q.S. al-An'am: 165). Dalam surat Hud ayat 61, Allah SWT berfirman *"Dan kepada Tsamud (Kami utus) saudara mereka Shaleh. Shaleh berkata: "Hai kaumku, sembahlah Allah, sekali-kali tidak ada bagimu Tuhan selain Dia. Dia telah menciptakan kamu dari bumi (tanah) dan menjadikan kamu pemakmurnya, karena itu mohonlah ampunan-Nya, kemudian bertobatlah kepada-Nya, Sesungguhnya Tuhanku amat dekat (rahmat-Nya) lagi memperkenankan (do'a hamba-Nya)"* (Q.S. Hud: 61). Jika mencermati perkataan yang disampaikan sebelumnya mengenai manusia, jelaslah bahwa manusia adalah makhluk ketuhanan yang diciptakan oleh Tuhan. Penciptaan mereka melampaui ciptaan lainnya, karena manusia memiliki tubuh fisik yang terdiri dari banyak komponen organik dan jiwa yang tidak dapat dipisahkan. Ada dasar rasional untuk percaya. Manusia tidak ada bandingannya dengan binatang, apalagi setara. Manusia berbeda dengan malaikat dan binatang. Namun, ada kalanya, manusia mungkin menunjukkan sifat-sifat kebinatangan ketika kita melihat sifat dan perilaku dasar mereka, terutama ketika mereka gagal mematuhi hukum dan konvensi masyarakat dan agama yang mereka ikuti. Pernyataan para filosof bahwa manusia setara dengan binatang atau monster memang kurang tepat, namun tidak bisa dianggap sepenuhnya salah. Namun, terbukti bahwa manusia memiliki ciri-ciri yang berbeda dibandingkan dengan hewan atau makhluk hidup, baik dalam aspek biologis maupun psikologis (Kurniawati & Bakhtiar, 2018).

2.3 Wajah

Wajah merupakan komponen penting tubuh manusia yang memiliki fungsi penting seperti menentukan penampilan, menyampaikan ekspresi wajah, dan membangun identitas diri. Setiap wajah, seperti wajah kembar identik, akan memiliki bentuk yang berbeda dan sama sekali tidak dapat dibedakan satu sama lain. Dalam ranah ilmiah, terdapat berbagai definisi wajah, khususnya: wajah adalah salah satu komponen kepala; ekspresi wajah; wajah, sesuai uraian yang diberikan Kamus Besar Bahasa Indonesia (KBBI). Wajah merupakan bagian anterior kepala manusia yang terbentang dari dahi hingga dagu. Area wajah meliputi dahi, alis, mata, hidung, mulut, pipi, bibir, gigi, kulit, rambut, dan dagu.

Wajah manusia merupakan ciri khas yang dapat diidentifikasi melalui penggunaan teknologi biometrik. Sistem biometrik memanfaatkan karakteristik biologis berbeda yang ditemukan pada manusia, dengan penekanan khusus pada fitur wajah, untuk memberikan informasi yang unik dan individual. Atribut khas ini mungkin terwujud dalam bentuk pola wajah yang spesifik pada setiap orang. Pola wajah memiliki sifat yang dapat diukur dan dianalisis yang dapat digunakan untuk tujuan deteksi atau otentikasi. Dengan demikian, wajah berfungsi sebagai alat identifikasi seseorang (Anggraini, 2020).

Perubahan wajah selanjutnya yang terjadi pada seseorang akibat penuaan adalah sebagai berikut:

1) **Wajah bayi (umur 1-2 Tahun)**

Penampilan wajah bayi terus mengalami perubahan sehingga sulit untuk menentukan apakah ia mirip dengan ayah atau ibunya. Periode ketidakpastian ini biasanya berlangsung selama 1-2 tahun pertama kehidupan (Sumber: DetikHealth diakses tanggal 10 November 2021).

Variasi dari wajah bayi yang khas adalah:

- a. Bentuk mata yang lebih besar.
- b. Wajah yang cenderung bulat.
- c. Ukuran kepala yang lebih besar.
- d. Alis yang pendek dan datar.
- e. Bentuk hidung yang relatif kecil dan pendek.

2) Wajah anak-anak (di atas 2 tahun sampai menjelang remaja)

Seiring bertambahnya usia anak-anak, daya tarik estetika fitur wajah mereka biasanya berkurang dibandingkan saat mereka masih bayi. Tipikal wajah anak-anak menunjukkan perpaduan antara ciri-ciri wajah kekanak-kanakan dan dewasa (Sumber: DetikHealth diakses tanggal 10 November 2021).

Transformasi Wajah anak-anak, yaitu:

- a. Ada beberapa gigi yang hilang atau jaraknya berjauhan.
- b. Alis yang lebih lengkap dibandingkan saat bayi.
- c. Telinga yang mulai membesar.
- d. Hidung yang tadinya kecil, mulai melebar.
- e. Lemak yang timbul umumnya berada di seluruh wajah yang bisa mengurangi kekhasannya.

3) Wajah dewasa muda (umur 18-25)

Struktur wajah ini biasanya tidak berubah dan sangat berbeda dengan struktur wajah bayi atau anak kecil. Mayoritas orang menganggap wajah mereka paling menarik pada tahap kehidupan ini. (Sumber: DetikHealth diakses tanggal 10 November 2021).

Transformasi yang terjadi pada kaum muda adalah :

- a. Berkurangnya elastisitas kolagen pada wajah sehingga menyebabkan jaringan lunak wajah menjadi kendur.
- b. Jaringan adiposa wajah biasanya berkurang, mengakibatkan berkurangnya ketebalan jaringan lunak wajah.
- c. Pada akhirnya, pengurangan jumlah jaringan lunak di wajah akan meningkatkan visibilitas tulang wajah.

4) Wajah dewasa (25-49 tahun)

Selama tahap kehidupan ini, individu biasanya mengalami transformasi wajah yang signifikan dan laju perkembangannya mulai melambat. Morfologi wajah pada tahap ini ditentukan oleh berbagai faktor, antara lain usia, jenis kelamin, dan latar belakang etnis atau ras (Sumber: DetikHealth diakses tanggal

10 November 2021). Ketika seseorang memasuki masa kedewasaan dan menjalani proses penuaan, mungkin akan muncul ciri-ciri umum tertentu yaitu:

- a. Kontur pipi yang menurun sehingga mengakibatkan rahang menonjol
- b. Sudut-sudut mulut, jaringan di sekitar mata, kelopak mata atas dan bawah serta jaringan di dahi menurun dengan sedikit berkerut.
- c. Ujung hidung bisa membesar dan menjadi bulat.

5) **Wajah Tua (50 tahun keatas)**

Seiring bertambahnya usia, wajah manusia mengalami transformasi yang signifikan, termasuk munculnya kerutan tambahan, perubahan struktur tulang pipi, dan perubahan posisi mata, hidung, dan mulut. Ciri-ciri khas wajah yang menua antara lain:

- a. Rambut mulai mengputih
- b. Rambut diwajah mulai mengputih
- c. Sudah muncul kerutan dengan jelas
- d. Mulai keriput
- e. Rambut mulai mengputih
- f. Muncul kerutan diwajah
- g. Muncul kusam

2.4 Citra

Citra (*Image*) adalah suatu bidang citra dua dimensi yang terdiri dari banyak piksel, dimana piksel-piksel tersebut mewakili bagian terkecil dari citra tersebut. Citra dihasilkan dan disusun dari kotak persegi panjang standar, sehingga jarak *horizontal* dan *vertikal* antar piksel sama. Citra dapat dipisahkan menjadi dua bagian yaitu citra diam (*still image*) dan citra bergerak (*moving image*). Citra diam diproyeksikan secara progresif, memberikan tampilan citra bergerak pada mata. Setiap citra dalam rangkaian disebut bingkai. Citra-citra yang dilihat di film layar lebar atau televisi umumnya terdiri dari ribuan *frame*.

Elemen piksel pada citra merupakan matriks indeks baris dan kolom yang mewakili suatu titik pada citra dan penyusunnya. Tahapan transformasi citra analog menjadi citra digital sering disebut dengan proses digitalisasi. Digitalisasi adalah proses transformasi citra, teks, dan suara dari objek yang dapat diamati

menjadi data elektronik yang dapat disimpan dan diproses untuk kegunaan lain. Citra digital pada komputer disusun dalam bentuk elemen grid atau piksel dalam bentuk matriks 2 dimensi. Untuk merepresentasikan warna pada setiap citra, elemen piksel mempunyai nilai tersendiri, dimana nilai dalam angka pada setiap piksel akan dicatat secara berurutan oleh komputer (Anggraini, 2020).

2.5 Citra Digital

Citra adalah citraan visual suatu benda atau sejumlah benda. Tentu saja, citra bisa bermacam-macam bentuknya, mulai dari foto individu, citra, foto rontgen, hingga citra satelit. Jenis citra terbagi menjadi tiga macam, yaitu citra biner (citra monokrom), citra skala abu-abu (*grayscale*), dan citra berwarna. Suatu citra digital dapat direpresentasikan dengan matriks berukuran $M \times N$, dimana setiap elemen dalam matriks tersebut mewakili sebuah piksel yang merupakan satuan terkecil dari suatu citra. Piksel memiliki dua sifat berbeda, khususnya koordinat dan intensitas atau warna. Koordinat (x, y) memuat nilai $f(x, y)$ yang merepresentasikan intensitas atau warna piksel pada tempat tertentu (A. Fadjeri, 2020). Suatu citra digital dapat direpresentasikan dalam bentuk matriks sebagai berikut:

$$f(x, y) = \begin{bmatrix} f(0,0) & f(0,1) & \dots & f(0, N-1) \\ f(1,0) & f(1,1) & \dots & f(1, N-1) \\ \vdots & \vdots & \dots & \vdots \\ f(M-1,0) & f(M-1,1) & \dots & f(M-1, N-1) \end{bmatrix} \dots \dots \dots (2.1)$$

Sumber : (A. Fadjeri, 2020)

Secara matematis, citra digital dapat direpresentasikan dengan matriks. Setiap elemen dalam matriks berhubungan dengan posisi dalam gambar, dengan nilai baris dan kolom menunjukkan koordinatnya. Fungsi intensitas $f(x, y)$ memberikan nilai pada setiap titik (x, y) pada gambar, yang menunjukkan intensitas, tingkat keabuan, atau warna piksel pada posisi tersebut. Proses digitalisasi melibatkan pengambilan sampel dan kuantifikasi gambar untuk menghasilkan matriks berukuran $M \times N$, di mana M mewakili jumlah baris dan N mewakili jumlah kolom. Selain itu, prosedur tersebut menentukan jumlah tingkat keabuan untuk setiap piksel pada gambar (Fadjeri, 2020).

2.5.1 Representasi Warna Digital

Warna digital merupakan hasil penggabungan tiga warna primer, merah, hijau, dan biru (RGB). Model warna RGB mewakili setiap warna sebagai triplet nilai intensitas, dengan masing-masing rentang intensitas dari 0 hingga 255 (Andono *et al.*, 2017), yaitu:

$$\text{Red} : \text{RGB} (255, 0, 0) \dots\dots\dots(2.2)$$

$$\text{Green} : \text{RGB} (0, 255, 0) \dots\dots\dots (2.3)$$

$$\text{Blue} : \text{RGB} (0, 0, 255) \dots\dots\dots (2.4)$$

Dari nilai triplet RGB persamaan (2.2) sampai (2.4) di atas dapat dikonversikan ke dalam nilai desimal seperti dibawah ini:

$$\text{Red} : 255*256^0 + 0*256^1 + 0*256^2 = 255 + 0 + 0 = 255 \dots\dots\dots (2.5)$$

$$\text{Green} : 0*256^0 + 255*256^1 + 0*256^2 = 0 + 65,280 + 0 = 65,280 \dots\dots\dots (2.6)$$

$$\text{Blue} : 0*256^0 + 0*256^1 + 255*256^2 = 0 + 0 + 16,711,680 = 16,711,680\dots\dots (2.7)$$

Rumus dasar mencari nilai RGB citra adalah:

$$R = \text{COLOR And RGB}(255, 0, 0) \dots\dots\dots (2.8)$$

$$G = (\text{COLOR And RGB}(0, 255, 0)) / 256 \dots\dots\dots (2.9)$$

$$B = ((\text{COLOR And RGB}(0, 0, 255)) / 256) / 256 \dots\dots\dots (2.10)$$

Dari persamaan (2.4) sampai (2.6) diatas, rumus RGB pada persamaan (2.11) sampai (2.9) menjadi:

$$\text{Nilai R} = c \text{ and } 255 \dots\dots\dots (2.12)$$

$$\text{Nilai G} = (c \text{ and } 65,280)/256 \dots\dots\dots (2.13)$$

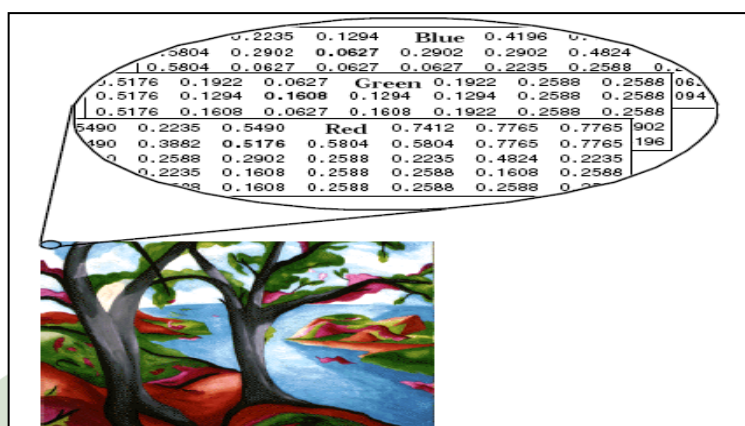
$$\text{Nilai B} = ((c \text{ and } 16,711,680)/256)/256 \dots\dots\dots (2.14)$$

2.5.2 Citra RGB

Model warna RGB terdiri dari tiga komponen warna dasar: merah (R/*red*), hijau (G/*green*), dan biru (B/*blue*). Model warna ini mengikuti pendekatan aditif, artinya jika beberapa warna ditambahkan maka akan menghasilkan warna putih. Foto RGB terkadang disebut sebagai gambar warna asli. Citra RGB merupakan citra digital yang terdiri dari tiga *layer* yang terdiri dari matriks data berukuran $m \times n \times 3$. Matriks ini berisi warna merah, hijau, dan biru untuk setiap pikselnya. Setiap *layer* mempunyai intensitas kecerahan warna yang berbeda-beda. Ketika

ketiga *layer* tersebut digabungkan, maka terciptalah kombinasi warna baru berdasarkan tingkat kecerahan warna yang diberikan oleh masing-masing *layer*.

Setiap *layer* memiliki kapasitas 8 bit, memungkinkan maksimal 256 tingkat kecerahan warna. Setiap *layer* warna mempunyai kemampuan untuk memberikan tingkat kecerahan yang berkisar antara 0 hingga 255. 0 melambangkan warna hitam dan 255 melambangkan warna putih. Gambar RGB terlihat pada Gambar 2.1 di bawah.



Gambar 2. 1 Citra RGB (Andono et al., 2017)

Citra digital dapat dikategorikan menjadi tiga kelompok berdasarkan nilai pikselnya, yaitu:

Model warna RGB adalah model warna aditif yang menggabungkan warna merah, hijau, dan biru dengan cara berbeda untuk menghasilkan beragam warna. Nama model diambil dari inisial tiga warna primer: merah, hijau, dan biru. Tujuan utama model warna RGB adalah untuk mendeteksi, menggambarkan, dan menampilkan citra dalam sistem elektronik, seperti televisi dan komputer. Namun, hal ini juga dapat diterapkan dalam fotografi konvensional (Fadjeri, 2020).

Setiap piksel dalam gambar RGB memiliki intensitas warna yang dihasilkan dari penggabungan tiga nilai intensitas pada saluran R, G, dan B. Misalnya, jika sebuah piksel memiliki nilai intensitas warna 255 pada saluran merah, 102 pada saluran hijau, dan 0 pada saluran biru, maka akan menghasilkan warna oranye. Jumlah total kemungkinan warna piksel potensial pada gambar

RGB truecolor 24-bit dihitung $256 \times 256 \times 256 = 16.777.216$ (Gusti Salamah & Ekawati, 2021).

2.5.3 Citra Biner

Konversi citra biner melibatkan ekstraksi sebagian citra skala abu-abu yang hanya memiliki dua nilai abu-abu berbeda: 0 untuk hitam dan 1 untuk putih. Dalam proses ini, setiap piksel pada citra biner secara konsisten diwakili oleh angka 1. Citra biner yang diperoleh setelah diubah menjadi hitam putih akan dianalisis kembali untuk mengidentifikasi posisi tepat objek yang bersangkutan dan menghilangkan nilai piksel yang tidak signifikan (Fadjeri, 2020).

Citra biner adalah citra dengan kedalaman bit piksel 1 bit sehingga hanya menghasilkan dua nilai intensitas warna. Citra skala *grayscale* dapat diubah menjadi citra biner dengan menerapkan prosedur *thresholding*. Selama prosedur *threshold*, nilai *threshold* diperlukan sebagai nilai pembatas untuk konversi. Nilai intensitas piksel yang lebih besar atau sama dengan nilai *threshold* akan diubah menjadi 1. Sebaliknya, nilai intensitas piksel yang berada di bawah ambang batas akan diubah menjadi nol (Gusti Salamah & Ekawati, 2021).

2.6 JPEG

Format JPEG, yang dirancang oleh *Joint Photographic Experts Group*, telah menjadi format foto digital yang paling banyak digunakan. Ini pertama kali dikembangkan pada tahun 1982 dan kemudian diterima sebagai standar kompresi internasional ISO/IFC 10918-1 untuk foto kontinu, termasuk citra skala *grayscale* dan berwarna. Standar ini dirancang khusus untuk memfasilitasi pemanfaatan citra dan grafik berkualitas unggul pada perangkat digital. Setiap file JPEG berisi penanda awal atau header yang dikenal sebagai "*start of image*" (SOI) dan penanda atau cuplikan penutup yang dikenal sebagai "*End of image*" (EOI).

Pemanfaatan format citra JPEG sedang meningkat karena meningkatnya keluaran foto digital, yang didorong oleh munculnya *smartphone* dan kemajuan media sosial. Menurut prediksi yang dibuat pada tahun 2017, diperkirakan setiap individu akan menghasilkan 1.600 foto digital dalam satu tahun. Hal ini menyebabkan semakin majunya format JPEG (Ardiansyah et al., 2020).

2.7 Deep Learning

Deep learning (pembelajaran mendalam) adalah bagian dari *machine learning* yang menggunakan *Deep Neural Network* untuk mengatasi masalah dalam *domain machine learning*. *Deep learning* meniru proses kognitif berpikir manusia (Saiful Islam, 2020). *Deep learning* adalah teknik pembelajaran yang menggunakan *artificial neural network* berlapis-lapis (*multilayer*). *Artificial Neural Network* dirancang untuk meniru struktur otak manusia, dengan neuron-neuron yang saling berhubungan membentuk jaringan yang sangat rumit (Nugroho et al., 2020).

Deep learning dikategorikan menjadi tiga kelompok berbeda berdasarkan arsitektur dan teknik yang digunakan.

1. Deep Network for Unsupervised or Generative Learning

Kategori ini mencakup identifikasi dan sintesis pola dalam data yang terlihat atau tersembunyi pada tingkat tinggi, tanpa mengetahui nama kelas target.

2. Deep Network for Supervised Learning

Kategori-kategori ini secara langsung berkontribusi pada kemampuan membedakan pola untuk tujuan klasifikasi, biasanya dengan mendeskripsikan kemungkinan kelas yang berbeda berdasarkan data yang diamati. Data label target dapat diakses secara konsisten baik secara langsung maupun tidak langsung untuk jenis pembelajaran yang diawasi ini. Kelas ini juga dikenal sebagai *discriminative deep networks*.

3. Hybrid Deep Network

Kategori ini berfokus pada diskriminasi yang sering kali difasilitasi oleh hasil dari jaringan dalam yang bersifat *generative* atau *unsupervised deep networks*. Tujuan lain tercapai ketika kriteria diskriminatif digunakan untuk memperkirakan parameter di salah satu jaringan *deep generative* atau *unsupervised deep networks* yang disebutkan di atas. Proses evaluasi kinerja model *deep learning* terdiri dari empat langkah yang diuraikan oleh A. Retnowardhani pada tahun 2019. Proses-proses tersebut adalah:

1. *Data Integration*

Data integration metode yang digunakan untuk menggabungkan data dari beberapa database menjadi satu database.

2. *Data Cleansing*

Data cleansing adalah langkah penting dalam fase *preprocessing* data yang melibatkan identifikasi dan perbaikan nilai yang hilang serta menyelesaikan perbedaan data.

3. *Data Reduction*

Data reduction adalah metode yang digunakan untuk mengurangi jumlah informasi dalam kumpulan data. Proses *data reduction* dapat menggunakan teknik *feature selection*.

4. *Feature Selection*

Feature selection adalah prosedur memilih *subset* fitur yang paling sesuai berdasarkan kriteria tertentu. Tujuan dari strategi ini adalah untuk menghilangkan fitur-fitur yang tidak relevan dan berlebihan, sehingga mengurangi jumlah fitur dalam model. *Feature selection* adalah proses yang berupaya menurunkan metrik kinerja (seperti interval kecepatan, daya prediktif, dan kesederhanaan model) dan menurunkan jumlah dimensi.

Menurut penelitian Lu Yifei (2017), metode deep learning memanfaatkan parameter sebagai berikut:

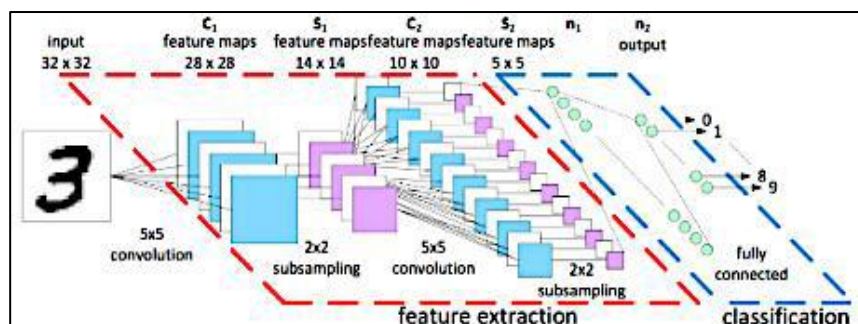
1. *Hidden Layer*, mengacu pada proses menentukan jumlah optimal lapisan tersembunyi dan jumlah *neuron* dalam setiap *layer* dalam arsitektur pembelajaran mendalam.
2. *Epochs*, menunjukkan jumlah iterasi yang diperlukan untuk memproses pengumpulan data. *Epochs* mewakili satu iterasi sistem *deep learning* yang belajar dari kumpulan data pelatihan lengkap. Era pembelajaran mendalam mengacu pada proses di mana suatu algoritma belajar dari kumpulan data pelatihan yang lengkap.
3. *Learning rate*, merupakan parameter pelatihan yang digunakan untuk menghitung nilai koreksi bobot selama proses pelatihan. Nilai *learning rate* dibatasi dalam interval dari nol (0) hingga satu (1). Ketika nilai *learning*

rate meningkat, akurasi jaringan menurun. Sebaliknya, jika kecepatan pembelajaran lebih kecil, akurasi jaringan meningkat, namun hal ini menyebabkan periode pelatihan lebih lama.

4. *Evaluate performance*, melibatkan evaluasi model berdasarkan *epochs* dan parameter *learning rate* yang telah ditentukan. Tujuannya adalah untuk memperoleh hasil terbaik dalam hal akurasi, *TPR/sensitivity* dan kurva *receiver operating characteristic (ROC)*.

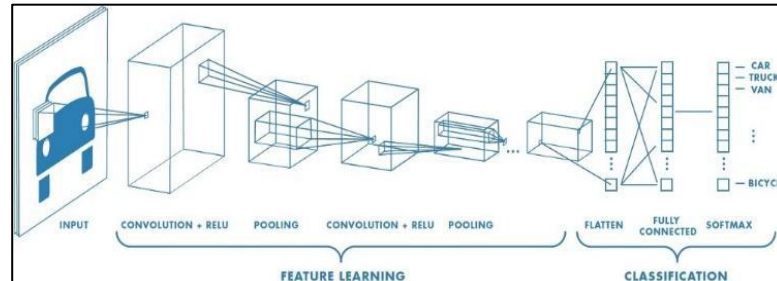
2.8 Teori Matematis Convolutional Neural Network

Convolutional Neural Network (CNN) adalah kemajuan dari *multilayer perceptron (MLP)* yang termasuk dalam kategori *neural network* bertipe *feed forward* (tidak berulang). *Convolutional Neural Network (CNN)* adalah jenis *neural network* yang secara khusus dimaksudkan untuk menganalisis dan memanipulasi masukan dua dimensi. CNN diklasifikasikan sebagai bentuk *Deep Neural Network* karena kedalaman jaringannya yang tinggi dan penggunaannya yang luas dalam memproses data gambar. CNN, atau *Convolutional Neural Network*, adalah model komputasi yang dirancang khusus untuk menganalisis citra visual. Ia mampu mendeteksi dan mengenali objek dalam gambar-gambar ini. CNN bekerja dengan merepresentasikan citra sebagai vektor berdimensi tinggi, yang berisi banyak parameter yang menggambarkan jaringan. Dari segi struktur dan fungsi, CNN mirip dengan *neural network* biasa (Nugroho et al., 2020). Di bawah ini adalah desain struktur *Convolutional Neural Network*:



Gambar 2.2 Arsitektur *Convolutional Neural Network* (Sumber : R. Meihendra Prasmatio, 2020)

Dan juga, berikut adalah penjelasan dari masing-masing proses dari *Convolutional Neural Network*:

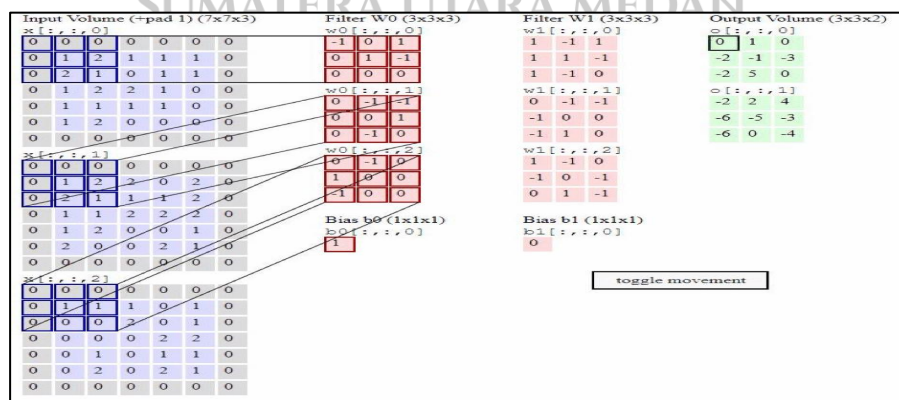


Gambar 2.3 Proses *Convolutional Neural Network* (Sumber : R. Meihendra Prasmatio, 2020)

Grafik tersebut mengilustrasikan proses sekuensial yang terjadi, dimana gambar masukan mengalami fase *feature learning* yang terdiri dari *convolutional layer*, *pooling* dan *ReLU*. Setelah itu, kita masuk ke tahap *classification*. Penjelasan rinci masing-masing fase akan diberikan pada paragraf berikutnya.

2.8.1 Convolutional Layer

Convolutional Layer: Setiap masukan yang melewati lapisan konvolusional akan menjalani operasi konvolusi. *Layer* tersebut akan mengubah setiap filter menjadi beberapa komponen data yang masuk dan menghasilkan *activation map* atau *feature map 2D*. Teknik konvolusi diilustrasikan pada Gambar 2.4.

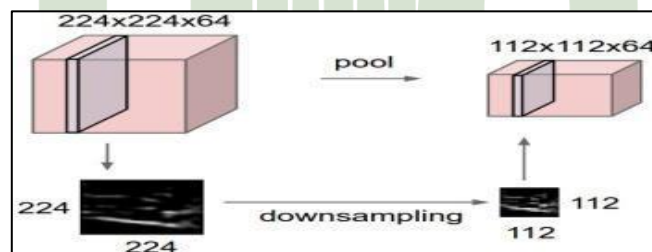


Gambar 2.4 Proses Konvolusi untuk menghasilkan *activation map* (Sumber : R. Meihendra Prasmatio, 2020)

Filter yang terdapat di *Convolutional Layer* memiliki dimensi panjang, tinggi, dan ketebalan yang sesuai dengan *channel* data masukan. Setiap *filter* akan mengalami pergeseran dan operasi "*dot*" menggunakan data masukan dan nilai *filter*. *Layer* konvolusional secara efektif mengurangi kompleksitas model dengan mengoptimalkan outputnya. Optimasi dicapai dengan menyesuaikan tiga parameter: *depth*, *stride*, dan *zero padding* (Santoso & Ariyanto, 2018).

2.8.2 Pooling Layer

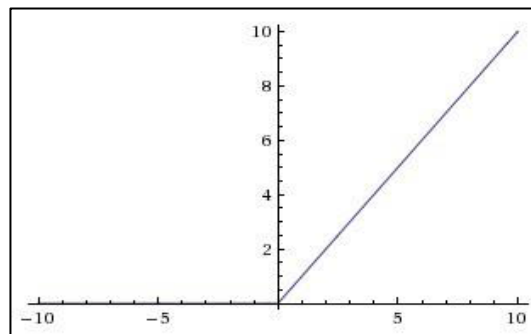
Pooling Layer mengikuti *Convolutional Layer* dalam arsitektur *neuron network*. *Pooling Layer* terdiri dari *filter* dengan ukuran dan langkah tertentu. Jumlah *stride* di setiap *stride* akan ditentukan oleh jumlah langkah yang akan diterapkan di seluruh *feature map* atau *activation map*. Biasanya, *pooling layer* yang digunakan dalam praktiknya adalah *max pooling* dan *average pooling*. Misalnya, saat menggunakan *max pooling* 2x2 dengan *stride* 2, nilai yang dipilih pada setiap pergeseran *filter* adalah nilai maksimum dalam wilayah 2x2. Sedangkan *average pooling* menghitung nilai rata-rata dalam wilayah yang sama (A. Santoso, 2018).



Gambar 2. 5 Contoh Diagram MAX Pooling Layer (Sumber : R. Meihendra Prasmatio, 2020)

2.8.3 Aktivasi ReLu

Aktivasi *ReLu* (*Rectified Linear Unit*) merupakan *layer* pada model CNN yang menerapkan fungsi $f(x) = \max(0, x)$. Fungsi ini menyelesaikan *thresholding* dengan mengatur semua nilai piksel di bawah nol hingga nol pada citra masukan. Fungsi aktivasi ini menetapkan semua nilai piksel dalam citra yang kurang dari nol hingga nol (Ilahiyah & Nilogiri, 2018).



Gambar 2. 6 Aktivasi *ReLU* (Sumber: Sarirotul Ilahiyah dan Agung Nilogiri, 2018)

Citra aktivasi *ReLU* memberikan gambaran menyeluruh tentang proses yang akan berlangsung pada lapisan ini. *Layer* ini berfungsi sebagai tahapan yang menentukan nilai piksel pada citra.

Fungsi aktivasi *ReLU* digunakan untuk menormalkan nilai keluaran *convolutional Layer*. Secara khusus, ini memastikan bahwa semua nilai di bawah 0 menggunakan fungsi $\max(0, x)$. Nilai *ReLU* dapat diturunkan untuk matriks masukan x menggunakan persamaan berikut:

$$f(x) = \max(0, x) \dots\dots\dots (2.16)$$

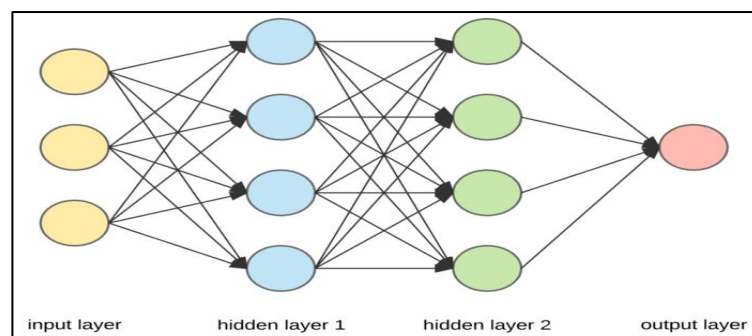
Pada tahap *backpropagation* CNN, digunakan turunan fungsi aktivasi untuk memperoleh nilai gradien input (Achmad et al., 2019). Berdasarkan persamaan diatas maka didapatkan turunan pada persamaan dibawah ini :

$$f(x) = \begin{cases} 1, & x > 0 \\ 0, & x \leq 0 \end{cases} \dots\dots\dots (2.17)$$

2.8.4 Fully Connected Layer

Feature map yang dihasilkan oleh *feature extraction layer* tetap berupa *multidimensional array*. Untuk menggunakannya sebagai masukan bagi lapisan yang terhubung sepenuhnya, Anda perlu ***flatten*** atau ***reshape feature map*** menjadi vektor. *Layer Fully-connected* yang sepenuhnya terhubung adalah *layer* yang setiap *neuron* aktivasi dari lapisan sebelumnya terhubung ke setiap *neuron* pada *layer* berikutnya, mirip dengan jaringan syaraf tiruan konvensional. Sebelum terhubung ke semua *neuron* pada *layer*, setiap aktivasi dari *layer* sebelumnya harus diubah menjadi data satu dimensi. *Layer fully-connected* sepenuhnya

biasanya digunakan dalam algoritma perceptron *multilayer* dengan tujuan memproses data untuk klasifikasi. Perbedaan antara *layer Fully-Connected* dan *layer* konvolusi biasa terletak pada pola konektivitas neuronnya. Pada *layer* konvolusi, neuron terhubung secara selektif ke bagian masukan tertentu, namun pada *layer Fully-Connected*, neuron membangun koneksi dengan setiap elemen masukan. Meskipun demikian, kedua lapisan tersebut masih melakukan perkalian titik, yang menunjukkan bahwa peran mereka agak mirip (Pujoseno, 2018).

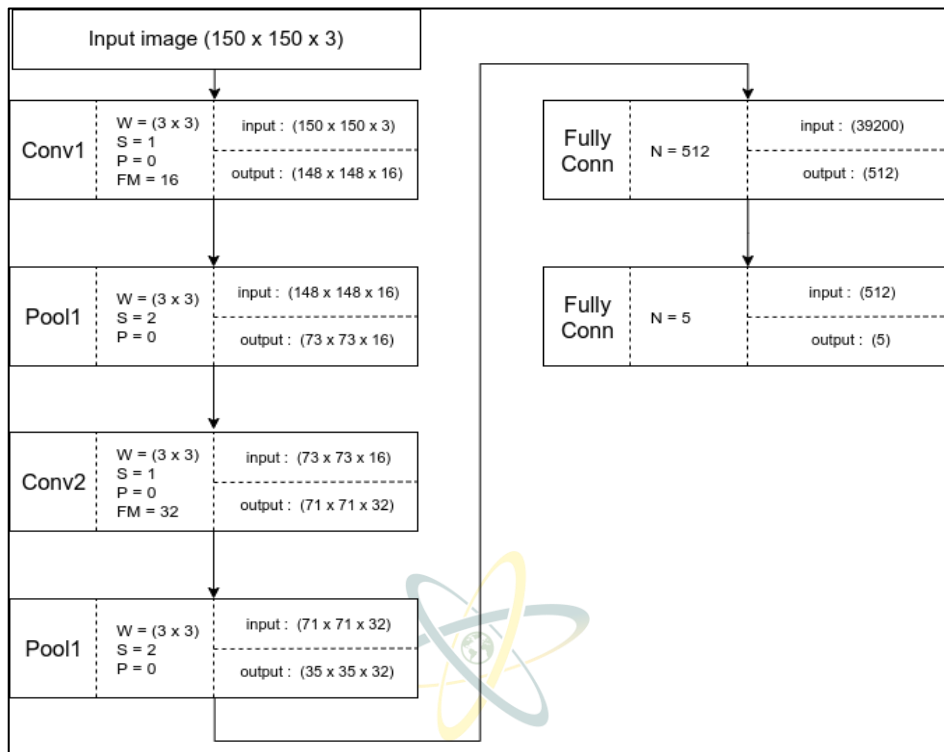


Gambar 2.7 *Fully Connected Layer* (Pujoseno, 2018).

Grafik yang tersedia menggambarkan *input layer*, *hidden layer 1*, *hidden layer 2* dan *output layer* yang saling berhubungan. Yang merupakan pemaparan dari *fully connected layer*.

2.8.5 Perhitungan Manual Secara Komputasi CNN

Neural network melibatkan penerapan penghitungan manual dalam program, yang berfungsi sebagai referensi untuk penghitungan sistem. Di bawah ini adalah ilustrasi perhitungan manual untuk kategorisasi gambar menggunakan Convolutional Neural Networks (CNN), dengan parameter yang tepat untuk setiap lapisan seperti terlihat pada gambar 2.7.



Gambar 2. 8 Contoh Perhitungan CNN Dalam Klasifikasi

Keterangan:

W = Window size

S = Stride

P = Padding

FM = Future Map

N = Jumlah Neuron

Pada contoh tersebut, input citra yang dimasukkan memiliki proses konvolusi atau *pooling* yang dapat dihitung dengan menggunakan rumus berikut:

$$Ukuran\ Akhir = \frac{Ukuran\ Awal - W + 2P}{s} + 1 \dots\dots\dots (2.18)$$

2.8.6 Perhitungan Manual Backpropagation

Backpropagation merupakan metode yang memanfaatkan aturan **delta** atau **gradient descent** untuk menentukan nilai minimal *loss function* dalam bobot. Bobot yang meminimalkan *loss function* dianggap sebagai jawaban atas permasalahan pembelajaran. *Backpropagation* adalah metode pelatihan yang umum digunakan untuk jaringan saraf. Karakteristik utamanya adalah

minimalisasi kesalahan pada keluaran jaringan. Algoritma *backpropagation* berfungsi dengan menyesuaikan nilai bobot *neuron* lapisan tersembunyi secara berulang. Pendekatan *Backpropagation* menyesuaikan bobot dengan memanfaatkan nilai *error* yang diperoleh dari perbandingan nilai *output* dengan target *output* yang diinginkan (Suhardin et al., 2021). Teknik pelatihan melalui *Backpropagation* biasanya terdiri dari dua tahap:

1. Tahap *Forward* (Maju)

Selama tahap *forward* (maju), perkalian *dot* akan dilakukan antara nilai bobot node masukan dan node *layer* sebelumnya. Tujuan dari operasi ini adalah untuk memperoleh nilai *output*, yang disebut dengan *forward pass*. Selanjutnya, hasil *output* yang diproyeksikan akan disandingkan dengan hasil yang diinginkan menggunakan fungsi yang biasa disebut sebagai *loss function*. Sederhananya, *loss function* adalah fungsi matematika yang mengukur keakuratan prediksi *neural network* kita mengenai hasil yang diinginkan. Tersedia beberapa *loss function*, termasuk kesalahan kuadrat, yang biasanya digunakan dalam regresi linier. Sedangkan proses penentuan *loss function* dengan menggunakan *squared error* dapat digambarkan sebagai berikut:

$$Loss = (prediction - target)^2 \dots\dots\dots (2.19)$$

Tahap Pertama : Propagasi maju (*Forward*)

- a. Langkah 0 : Inisialisasi semua bobot dengan bilangan kecil
- b. Langkah 1 : Jika kondisi berhenti belum terpenuhi, lakukan langkah 2-9
- c. Langkah 2 : Untuk setiap pasangan data pelatihan lakukan langkah 3-8
- d. Langkah 3 : Tiap unit masukan menerima sinyal dan meneruskannya ke unittersembunyi di atasnya
- e. Langkah 4 : Hitunglah semua *output* di unit tersembunyi tersebut $z_j (j=1,2,\dots,p)$. Perhatikan rumus dibawah ini :

$$z_{netj} = v_{0j} + \sum_{i=1}^n x_i v_{ij} \dots\dots\dots (2.20)$$

$$z_j = f(z_{netj}) = \frac{1}{-z_{netj}}$$

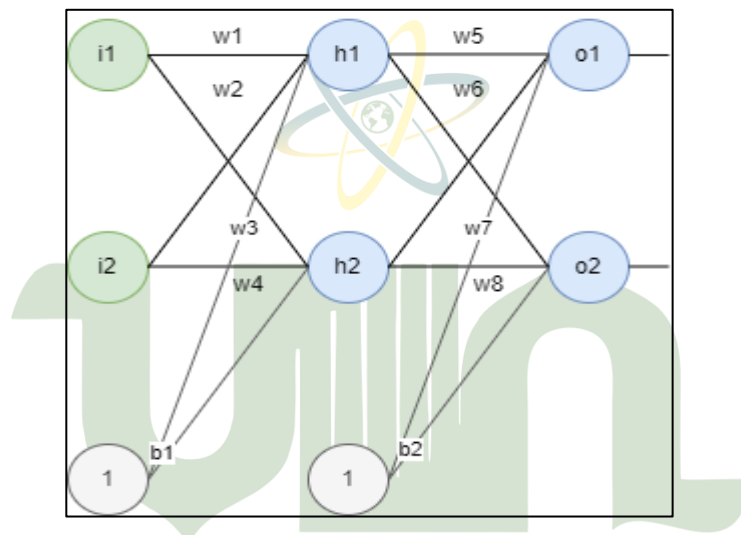
- f. Langkah 5 : Hitunglah semua keluaran jaringan di unit $y_k (k = 1,2, \dots, m)$ perhatikan rumus dibawah ini :

$$Y_{netk} = w_{k0} + \sum_{i=1}^n z_i w_{ki} \dots\dots\dots (2.21)$$

$$y_k = f(y_{netk}) = \frac{1}{1+e^{-y_{netk}}} \dots\dots\dots (2.22)$$

2. Tahap *Backward* (Mundur)

Pada tahapan *backward* (mundur), tahapannya sangat mirip dengan tahapan pada tahap *forward pass*. Selama proses *backward pass*, nilai *loss* yang diperoleh akan disebar dan berfungsi sebagai pengganda bobot untuk semua node lapisan sebelumnya untuk menghitung gradien. Sebagai ilustrasi, jika Anda ingin mengubah parameter W_{k10} , Anda harus terlebih dahulu memastikan besarnya perubahan. Untuk menentukan *loss* yang bergantung pada *output*, hitung turunan parsial *loss function* terhadap *output*.



Gambar 2. 9 Struktur Tahap Backpropagation (Medium.com, 2019)

Setelah nilai *gradient loss* dari parameter target diperoleh, nilai parameter tersebut akan dimodifikasi menggunakan proses *Stochastic Gradient Descent* (SGD). Algoritme SGD berupaya meminimalkan fungsi kerugian suatu parameter dengan menggunakan sebagian kecil dari *gradient loss function* yang telah dihitung sebelumnya. Nilai kecepatan pembelajaran mewakili beberapa di antaranya. Algoritme *Stochastic Gradient Descent* (SGD) akan meminimalkan *loss function* secara iteratif dengan terus memperbarui nilai parameter hingga mencapai titik minimum ideal.

Tahap kedua : Propagasi Mundur

Langkah 6 : Hitung factor δ unit keluaran berdasarkan error pada setiap y_k

($k=1,2, \dots, m$) Perhatikan rumus berikut ini:

$$\delta_k = (t_k - y_k) f'(y_{net_k}) = (t_k - y_k) y_k (1 - y_k)$$

δ_k merupakan unit kesalahan yang akan dipakai dalam perubahan bobot *layer* di bawahnya.

Langkah 7 : Hitunglah suku perubahan bobot w_{kj} dengan laju percepatan α .

Perhatikan rumus berikut ini :

$$\Delta w_{kj} = \alpha \delta_k z_j \quad ; k = 1,2, \dots, m ; j = 0,1,2, \dots, p$$

2.8.7 Akurasi

Akurasi didefinisikan sebagai persentase dari data uji yang diklasifikasikan ke kelas yang benar. Akurasi dapat dinyatakan dalam persamaan berikut:

(Qolbiyatul Lina, 2019),

$$Akurasi = \frac{TP+TN}{TP+TN+FP+FN} \dots\dots\dots (2.23)$$

Keterangan :

TP = *True Positive*

TN = *True Negative*



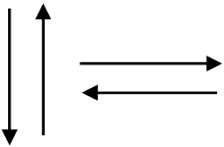
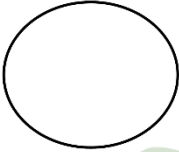
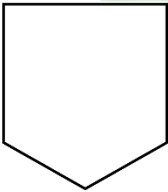
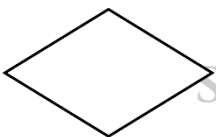

FP = *False Positive*

FN = *False Negative*

2.9 Flowchart

Flowchart adalah representasi metodis dari proses dan alasan di balik operasi penanganan informasi, atau representasi visual dari langkah-langkah dan urutan prosedur dalam suatu program. *Flowchart* membantu analis dan pemrogram dalam menguraikan tantangan menjadi bagian-bagian yang lebih kecil dan memfasilitasi analisis berbagai pendekatan dalam operasi (Murdock, 2018).

Tabel 2. 1 Simbol-simbol *flowchart* (Murdock, 2018)

Simbol	Arti	Keterangan
	<i>Input/Output</i>	Simbol input/output digunakan untuk mewakili data input/output
	Proses	Simbol proses digunakan untuk mewakili suatu proses
	Garis Alir	Simbol garis alir (<i>flow lines</i>) digunakan untuk menunjukkan arus dari proses.
	Penghubung	Simbol penghubung (<i>connector symbol</i>) digunakan untuk menunjukkan sambungan dari bagan alir yang terputus di halaman yang masih sama.
	Penghubung	Simbol penghubung (<i>connector symbol</i>) digunakan untuk menunjukkan sambungan dari bagan alir yang terputus di halaman lainnya.
	Keputusan	Simbol keputusan (<i>decision symbol</i>) digunakan untuk suatu penyeleksian kondisi di dalam program.
	Proses Terdefinisi	Simbol proses terdefinisi (<i>predefined process symbol</i>) digunakan untuk menunjukkan suatu operasi yang rinciannya ditunjukkan di tempat lain

2.10 Web

Jaringan situs web adalah kumpulan situs web yang menyajikan informasi tekstual, foto atau film, animasi, suara, atau kombinasi elemen-elemen ini. Komponen-komponen ini mungkin bersifat statis atau dinamis, bergantung pada

karakteristiknya. Suatu kompleks struktur yang saling berhubungan berkaitan dengan setiap elemen yang terkait dengan situs web (hyperlink) (Hasan & Muhammad, 2020).

2.11 *Flask*

Flask merupakan sebuah web *framework* yang memanfaatkan bahasa pemrograman. *Flask* bermanfaat untuk mengembangkan kerangka aplikasi dan membuat antarmuka web. Pengembang dapat memanfaatkan *Flask*, kerangka kerja Python, untuk membangun situs web yang terorganisir dengan baik dan memfasilitasi pengelolaan situs web. *Flask* diklasifikasikan sebagai *microframework* karena independensinya dari alat atau perpustakaan tertentu untuk penggunaannya. *Flask* tidak dilengkapi dengan fungsi dan komponen standar seperti validasi formulir dan database yang sudah diinstal sebelumnya. Hal ini disebabkan oleh fakta bahwa fungsi dan komponen ini disediakan oleh entitas eksternal, dan *Flask* memiliki kemampuan untuk memanfaatkan ekstensi yang memberikan kesan bahwa fitur dan komponen ini sebenarnya adalah bagian dari implementasi *Flask* sendiri. Meski disebut sebagai *microframework*, *Flask* tidak kekurangan kemampuannya. *Flask* disebut sebagai *microframework*, yang berarti tujuan utamanya adalah menjaga inti aplikasi tetap minimal dan mudah diperluas (Irsyad, 2018).

2.12 *Keras*

Keras adalah *framework deep learning* berbasis Python yang menawarkan pendekatan langsung untuk mendefinisikan dan melatih berbagai model *deep learning*. *Keras* ini awalnya dirancang oleh para peneliti dengan tujuan untuk mempercepat pengujian. *Keras* saat ini merupakan kerangka kerja yang paling banyak digunakan di Kaggle dan juga sangat lazim dalam kontes pembelajaran mendalam di seluruh dunia. *Framework* ini kompatibel dengan dua bahasa pemrograman yang paling banyak digunakan untuk ilmu data, khususnya R dan Python. *Keras* memiliki beberapa manfaat, termasuk eksekusi kode yang efisien pada CPU dan GPU, API intuitif untuk membuat prototipe *deep learning*, dan kompatibilitas dengan model RNN (Sen et al., 2020).

2.13 *Python*

Python adalah bahasa pemrograman dinamis yang memungkinkan pemrograman berbasis objek. *Python* telah diberikan banyak lisensi dengan versi berbeda. Namun, pada dasarnya dapat diakses dan digunakan secara bebas, bahkan untuk alasan komersial. Perbedaan *Python* dengan bahasa pemrograman lainnya terletak pada sintaksis dan konvensi yang mengatur susunan kode program. *Python* adalah bahasa pemrograman serbaguna yang kompatibel dengan berbagai sistem operasi, termasuk Linux. Ini disertakan di hampir semua distribusi Linux. Pemrogram dapat memprioritaskan pengembangan aplikasi dengan memanfaatkan kode yang tidak rumit dan lugas. Selain itu, *Python* merupakan perangkat lunak serbaguna dan tersedia secara gratis yang dapat dijalankan di banyak sistem operasi (Saiful Islam, 2020)



Gambar 2. 10 Logo *Python*

(Sumber : <https://www.python.org/community/logos>)

2.14 *Tensorflow*

TensorFlow adalah *library* perangkat lunak yang dibuat oleh Tim Google Brain, sebuah divisi dari Google, dengan tujuan memfasilitasi *machine learning* dan penelitian jaringan saraf (Wulan Anggraini, 2020). *Tensorflow* mengintegrasikan aljabar komputasi dengan teknik kompilasi, memungkinkan komputasi yang efisien dari berbagai ekspresi matematika. *TensorFlow* mencakup berbagai atribut yang menonjol (Anggraini, 2020).

1. Memberikan definisi yang jelas dan tepat, meningkatkan efisiensi, dan menghitung ekspresi wajah secara matematis menggunakan *array multidimension (tensors)*.
2. Pemrograman yang memfasilitasi pemanfaatan jaringan saraf dalam dan teknik *machine learning*.

3. Memanfaatkan GPU (*Graphics Processing Unit*) secara efektif dengan mengotomatiskan manajemen memori dan mengoptimalkan penggunaan kembali data. *Tensorflow* memiliki kemampuan untuk menulis kode yang dapat dieksekusi pada CPU dan GPU. Lebih tepatnya, *TensorFlow* memiliki kemampuan untuk menentukan bagian komputasi mana yang harus ditransfer ke GPU.
4. Mesin memiliki skalabilitas komputasi tingkat tinggi ketika berhadapan dengan kumpulan data yang besar.



Gambar 2. 11 Logo *Tensorflow* (Sumber : <https://www.tensorflow.org/community/3p-logo-generator>)

2.15 Penelitian terdahulu

Para peneliti telah melakukan banyak penelitian tentang identifikasi citra. Berikut ini adalah:

1. Melakukan penelitian tentang “Implementasi Metode *Convolutional Neural Network* (CNN) Untuk Klasifikasi Jenis Bunga Anggrek”. Penelitian dilakukan oleh Mohammad Zainuri dan Danar Putra Pamungkas (Zainuri, Mohammad. Pamungkas, Danar Putra. 2020). Temuan penelitian ini menunjukkan probabilitas kemiripan sebesar 0,872.
2. Melakukan penelitian mengenai “*Algoritma Convolutional Neural Network* untuk Pengenalan Wajah”. Kajian tersebut dilakukan oleh Sriyati, Arief Setyanto, dan Emha Taufiq Luthfi (Sriyati., Setyanto, Arief., Luthfi, Emha Taufiq. 2020). Penelitian ini menghasilkan tingkat akurasi 98,8% dari 15 penelitian, dengan mempertimbangkan arsitektur dan faktor relevan lainnya.
3. Melakukan penelitian mengenai “*Convolutional Neural Network* untuk memprediksi usia dan jenis kelamin secara akurat”. Ditulis oleh Afrizal Zein (Zein, Afrizal. 2020). Penelitian tersebut menghasilkan tingkat akurasi

keseluruhan sebesar 90%, dengan klasifikasi gender mencapai tingkat akurasi 96%. Hal ini dicapai dengan menggunakan kumpulan data pelatihan sebanyak 1000 sampel dan kumpulan data pengujian sebanyak 100 sampel.

4. Melakukan penelitian mengenai “Implementasi *Deep Learning* Dalam Identifikasi Jenis Tumbuhan Berdasarkan Citra Daun Menggunakan *Convolutional Neural Networks*”. Dilakukan oleh Sarirotul Divineyah dan Agung Nilogiri pada tahun 2018 (Sarirotul Divineyah., Agung Nilogiri. 2018). Temuan penelitian menunjukkan bahwa presisi terbaik dicapai pada percobaan pertama dengan menggunakan *fold cross* 10 kali lipat, khususnya pada *fold* ke-7, dengan tingkat akurasi 90%. Tingkat akurasi rata-rata dalam kategorisasi sistem yang dicapai melalui eksperimen validasi silang sepuluh kali lipat adalah 85,21%. Sistem yang dibangun dapat mengidentifikasi jenis genus tanaman secara akurat dengan tingkat presisi sebesar 90,8%.
5. Melakukan penelitian mengenai “*Deep Learning* Deteksi Wajah Hijab Menggunakan *Algoritma Convolutional Neural Network (CNN)* dengan *Tensorflow*”. Dilakukan oleh Wulan (Wulan Anggraini. 2020). Temuan penelitian menunjukkan bahwa model CNN mencapai tingkat akurasi sebesar 92% pada data pelatihan dan 87% pada data pengujian.
6. Melakukan penelitian mengenai “Kategorisasi Citra Porno Menggunakan Teknik *Convolutional Network* pada Perangkat *Smartphone* Berbasis Android”. Dieksekusi oleh Heru (Heru Mulyana. 2020). Temuan penelitian menunjukkan bahwa ketika model terbaik diuji dengan *dataset* KIA, diperoleh kinerja akurasi sebesar 91,67%, tingkat presisi 91%, tingkat recall 92%, dan waktu komputasi 6 *ms*.
7. Melakukan penelitian mengenai "Klasifikasi Citra Menggunakan *Convolutional Neural Network (CNN)* di *Caltech 101*". Dieksekusi oleh Wayan Suartika, Arya Yudhi, dan Rully Soelaiman. Temuan penelitian menunjukkan bahwa pemanfaatan *Convolutional Neural Network* bersama dengan pendekatan *preprocessing* sangat dapat diandalkan dalam menilai keakuratan klasifikasi *citra* objek secara akurat. Hal ini ditunjukkan dengan hasil akurasi yang berkisar antara 20% hingga 50%.

8. Melakukan penelitian mengenai “Penerapan *Convolutional Neural Networks* untuk Mengkategorikan Alat Tulis Menggunakan *Deep Learning*”. Dieksekusi oleh Jimmy Pujoseno. Temuan penelitian menunjukkan bahwa model yang dibangun telah mencapai tingkat akurasi 95% dalam mengkategorikan alat tulis berdasarkan data pengujian.
9. Melakukan penelitian mengenai “Implementasi Pengolahan Citra Dengan Menggunakan Teknik Konvolusi Untuk Pelembutan Citra (*Image Smoothing*) Dalam Operasi Reduksi *Noise*”. Dilakukan pada tahun 2017 oleh Tugiono, Hafizah, dan Asyahri Hadi Nasyuha. Hasil penelitian menunjukkan bahwa *filter median* merupakan *filter* yang paling efektif untuk menghilangkan noise pada gambar, terlihat dari nilai parameter MSE dan PSNR yang diperoleh dari hasil pengujian. Semakin besar nilai *Mean Squared Error* (MSE), maka disparitas antara kedua gambar yang dibandingkan juga semakin besar.
10. Melakukan penelitian tentang “Deteksi Usia Menggunakan Foto Menggunakan *Deep Learning*”. Dipandu oleh Ridhwan Saiful Islam pada tahun 2020. Penelitian tersebut menghasilkan tingkat akurasi yang signifikan selama proses pelatihan. Secara khusus, temuan tes menunjukkan akurasi sebesar 43,11% untuk anak usia 4-6 tahun, 92,48% untuk usia 8-12 tahun, 56,90% untuk usia 15-20 tahun, 68,32% untuk usia 25-32 tahun, dan 77,04% untuk usia 60-100 tahun.