

BAB 2

DASAR TEORI

2.1 KAJIAN PUSTAKA

Penelitian oleh Ananta Dwi Prayoga Alwy mengusulkan metode klasifikasi untuk menentukan kematangan daun selada berdasarkan fitur warna menggunakan algoritma KNN. Penelitian ini bertujuan untuk mengembangkan sistem untuk mengklasifikasikan kematangan daun selada dengan *accuracy* yang tinggi menggunakan teknik pengolahan citra. Metodologi penelitian terbagi menjadi tiga tahap utama diantaranya akuisisi citra, pra-pemrosesan, segmentasi, dan klasifikasi. Citra-citra diperoleh menggunakan kamera, dan ruang warna diubah dari *Red, Green, Blue* (RGB) menjadi *Lightness, a* dan *b* (LAB). Segmentasi dilakukan menggunakan *thresholding* pada ruang warna LAB, dan algoritma KNN digunakan untuk klasifikasi. Hasil penelitian menunjukkan *accuracy* sebesar 95% dengan harapan sistem ini dapat membantu petani selada dan menjadi standar untuk perbaikan di masa depan. Penelitian ini berfokus pada klasifikasi kematangan daun selada berdasarkan fitur warna menggunakan algoritma KNN. Proses ini melibatkan segmentasi citra, operasi morfologi, ekstraksi fitur, dan klasifikasi. Hasil penelitian menunjukkan tingkat *accuracy* yang tinggi, yaitu 95%, dalam mengklasifikasikan daun selada yang matang dan belum matang [8].

Penelitian yang dilakukan oleh Satrya Darmawan Putra Bahari mengusulkan menggunakan teknik *computer vision* sebagai salah satu metode klasifikasi buah apel, jeruk, dan pisang. Penelitian ini bertujuan untuk mengembangkan sistem otomatis yang dapat mengidentifikasi kategori buah serta menilai kualitas buah yang didasarkan pada gambar buah yang diambil. Gambar tersebut digunakan sebagai input dalam klasifikasi menggunakan algoritma *Deep Learning* dengan model CNN. Hasil dari penelitian menunjukkan bahwa metode yang digunakan mampu mengklasifikasikan buah-buahan segar seperti apel, jeruk, dan pisang dengan tingkat *accuracy* yang tinggi. Dari hasil pengujian didapatkan *accuracy* klasifikasi kualitas buah-buahan dengan nilai 98.17% dan *accuracy* klasifikasi kategori buah-buahan dengan nilai 95.38% [9].

Penelitian yang dilakukan oleh Febian Fitra Maulana membahas tentang klasifikasi gambar buah menggunakan CNN dalam bidang pengolahan citra. Arsitektur CNN yang digunakan dalam penelitian ini terdiri dari 3 lapisan konvolusi dan 2 lapisan terhubung penuh. Proses pembelajaran dilakukan untuk melatih model CNN, dan proses pengujian dilakukan untuk mengevaluasi *accuracy* model dalam mengklasifikasikan gambar buah. Proses pembelajaran bertujuan untuk menemukan model CNN terbaik dengan *accuracy* tertinggi dan kerugian terendah. Proses pengujian mengevaluasi kinerja model CNN menggunakan 345 gambar pengujian. *Accuracy* model dihitung berdasarkan klasifikasi yang benar dan salah. Penelitian ini menyimpulkan bahwa model CNN mencapai *accuracy* tinggi sebesar 97,97% dalam mengklasifikasikan gambar buah [10].

Penelitian yang dilakukan oleh Hanin Latif Fuadi membahas tentang klasifikasi kematangan tanaman hidroponik pakcoy menggunakan algoritma *Support Vector Machine* (SVM). Penelitian ini berfokus pada penggunaan pembelajaran mesin dan algoritma SVM untuk mengklasifikasikan tanaman Pakcoy hidroponik berdasarkan ukuran batang dan daun mereka. Dengan menerapkan algoritma SVM pada data citra digital tanaman Pakcoy, sebuah model pembelajaran mesin dapat mengklasifikasikan tanaman sebagai "kecil" atau "besar" untuk mewakili tanaman belum matang dan matang. *Dataset* dinormalisasi dan dibagi menjadi data latih dan data uji. Algoritma SVM digunakan untuk melatih dan menguji model. Kinerja model dievaluasi menggunakan *Confusion Matrix*. Hasil dari penelitian ini menunjukkan bahwa penggunaan algoritma SVM dalam klasifikasi kesiapan panen tanaman Pakcoy hidroponik berdasarkan ukuran batang dan daun dapat mencapai tingkat *accuracy* di atas 79%. Pengujian dari luar *dataset* juga menunjukkan performa yang layak, dengan tingkat *accuracy* yang tinggi lebih dari 95% pada setiap kelasnya [11].

Penelitian yang dilakukan oleh Bilawal Haesri membahas tentang implementasi algoritma *Machine Learning* khususnya algoritma KNN ke dalam sistem menggunakan Raspberry Pi4. Sistem ini dirancang untuk mengklasifikasikan tanaman bayam hidroponik ke dalam kategori siap panen dan belum siap panen. Penelitian ini fokus pada klasifikasi kesiapan panen bayam hijau menggunakan pemrosesan citra dan algoritma KNN. Waktu komputasi sistem,

mulai dari pengambilan gambar bayam hingga menampilkan hasil klasifikasi, rata-rata 1,4 detik. *Accuracy* klasifikasi ditemukan mencapai 100% menggunakan 12 data uji [12].

Penelitian yang dilakukan oleh Fakhri Habib Hawari membahas tentang pengklasifikasian penyakit pada tanaman padi menggunakan algoritma CNN. Tujuan dari penelitian ini adalah untuk membantu dan mendidik petani dalam mengklasifikasikan penyakit pada tanaman padi dan mengurangi risiko kegagalan panen. Penelitian ini menggunakan *dataset* berupa gambar daun padi dan menerapkan algoritma CNN untuk mengklasifikasikan penyakit. Hasil penelitian menunjukkan bahwa algoritma CNN mampu memberikan *accuracy* yang tinggi dalam mengklasifikasikan penyakit pada daun padi. *Accuracy training* mencapai 85%, *accuracy testing* mencapai 86%, dan *accuracy testing* mencapai 95%. Hal ini membuktikan bahwa metode CNN mampu diterapkan untuk mengidentifikasi citra daun padi yang terinfeksi penyakit dengan tingkat keakuratan yang tinggi [13].

Penelitian yang dilakukan oleh Aminatus Syarifah membahas tentang klasifikasi tingkat kematangan jambu bol merah menggunakan pengolahan citra digital dan metode KNN. Penelitian bertujuan untuk menciptakan sistem untuk mengklasifikasikan tingkat kematangan jambu bol merah berdasarkan fitur warna dan tekstur yang diekstraksi dari gambar. Aplikasi yang dibuat oleh peneliti mencakup pengolahan citra untuk klasifikasi tingkat kematangan jambu bol. Langkah-langkahnya meliputi input gambar, *resize*, konversi warna, ekstraksi fitur, klasifikasi, dan *output* hasil klasifikasi. Pengolahan citra melibatkan konversi warna RGB ke *Hue*, *Saturation* dan *Value* (HSV) dan ekstraksi fitur menggunakan metode *Gray Level Co-occurrence Matrix* (GLCM). Penelitian ini menggunakan metode KNN untuk mengklasifikasikan tingkat kematangan jambu bol berdasarkan data *training* dan *testing*. Hasil pengujian menunjukkan tingkat *accuracy* sebesar 93%. Metode ini menggunakan *confusion matrix* untuk mengevaluasi ketepatan hasil klasifikasi [14].

Tabel 2.1 Penelitian Terdahulu

Penulis Penelitian	Judul Penelitian	Hasil Penelitian	Perbedaan
Ananta Dwi Prayoga Alwy (2022)	Klasifikasi Kematangan Daun Selada Berdasarkan Fitur Warna Menggunakan KNN	Penelitian ini berfokus pada klasifikasi kematangan daun selada berdasarkan fitur warna menggunakan algoritma KNN. Hasil penelitian menunjukkan tingkat <i>accuracy</i> yang tinggi, yaitu 95%, dalam mengklasifikasikan daun selada yang matang dan belum matang.	Metode penelitian menggunakan algoritma KNN. Sedangkan pada penelitian yang akan dilakukan menggunakan CNN.
Satrya Darmawan Putra Bahari (2023)	Klasifikasi Buah Segar Menggunakan Teknik <i>Computer Vision</i> Untuk Pendeteksian Kualitas Dan Kesegaran Buah	Metode VGG16, yang merupakan arsitektur CNN, digunakan untuk melatih model yang dapat mengklasifikasikan jenis buah jeruk, apel, dan pisang. Metode VGG16, yang merupakan arsitektur CNN, digunakan untuk melatih model yang dapat mengklasifikasikan jenis buah jeruk, apel, dan pisang.	Objek penelitian berupa klasifikasi jenis buah. Sedangkan objek pada penelitian yang akan dilakukan yaitu klasifikasi keisapan panen selada keriting.
Febian Fitra Maulana (2019)	Klasifikasi Citra Buah Menggunakan CNN	Model CNN yang menggunakan perpaduan 3 <i>covolutional layer</i> dan 2 <i>hidden layer</i> mampu	Objek penelitian berupa klasifikasi gambar buah. Sedangkan objek

		mengklasifikasi citra buah dengan <i>accuracy</i> yang tinggi. Penelitian ini menghasilkan <i>accuracy</i> sebesar 97,97% yang didapatkan selama proses pelatihan.	pada penelitian yang akan dilakukan yaitu klasifikasi kesiapan panen selada keriting.
Hanin Latif Fuadi (2021)	Klasifikasi Kematangan Tanaman Hidroponik Pakcoy Menggunakan Metode SVM	Pengujian model <i>Machine Learning</i> dengan metode SVM menghasilkan tingkat <i>accuracy</i> sebesar 100% pada proses <i>training</i> dalam mengklasifikasi kematangan tanaman Pakcoy. Selama proses evaluasi, model ini mampu mengklasifikasi citra acak di luar <i>dataset</i> dengan tingkat <i>accuracy</i> lebih dari 79%.	Penelitian menggunakan <i>machine learning</i> dengan metode SVM. Sedangkan pada penelitian yang akan dilakukan menggunakan CNN.
Bilawal Haesri (2022)	Klasifikasi Kesiapan Panen Tanaman Hidroponik Bayam Hijau menggunakan Metode Pengolahan Citra dan KNN berbasis Raspberry Pi	Pengujian <i>accuracy</i> sistem dalam melakukan klasifikasi kesiapan panen menggunakan KNN dan <i>dataset</i> citra sebanyak 27 gambar dengan pembagian 9 data <i>training</i> , 6 data <i>validation</i> , dan 12 data <i>testing</i> mendapatkan <i>accuracy</i> 100% pada klasifikasi bayam hijau belum panen,	Penelitian tersebut menggunakan metode KNN untuk klasifikasi kesiapan panen bayam hijau. Sedangkan pada penelitian yang akan dilakukan menggunakan CNN.

		dan 100% <i>accuracy</i> pada klasifikasi bayam hijau siap panen.	
Fakhri Habib Hawari (2022)	Klasifikasi Penyakit Padi Menggunakan Algoritma CNN	Dari hasil penelitian ini, dapat disimpulkan bahwa metode CNN dapat diterapkan untuk mengklasifikasikan citra daun padi yang berpenyakit. <i>Accuracy</i> tertinggi yang dicapai adalah 85% untuk data <i>training</i> , 86% untuk data validasi, dan 95% untuk data <i>testing</i> .	Objek penelitian berupa klasifikasi penyakit pada padi. Sedangkan objek pada penelitian yang akan dilakukan yaitu klasifikasi kesiapan panen selada keriting.
Aminatus Syarifah (2022)	Klasifikasi Tingkat Kematangan Jambu Bol Berdasarkan Pengolahan Citra Digital Menggunakan Metode KNN	Berdasarkan pemaparan mengenai hasil pengujian yang telah dilakukan aplikasi klasifikasi tingkat kematangan jambu bol ini memiliki tingkat <i>accuracy</i> sebesar 93% dari data <i>testing</i> sejumlah 30, data <i>training</i> sebanyak 60 data, dan menggunakan nilai ketetanggaan $k=1$.	Metode KNN untuk klasifikasi tingkat kematangan jambu bol. Sedangkan penelitian yang akan dilakukan menggunakan CNN untuk klasifikasi kesiapan panen selada keriting

2.2 DASAR TEORI

Penelitian ini memerlukan dasar teori yang mencakup pemahaman mengenai pengertian dari objek penelitian, metode yang akan digunakan, serta hal-hal lain yang berkaitan dalam penelitian ini.

2.2.1 Selada Keriting

Selada atau yang juga dikenal sebagai *Lactuca sativa* dalam bahasa Latin, merupakan tanaman dari famili *Compositae*. Selada adalah sayuran populer yang sering dikonsumsi sebagai pelengkap hidangan karena memiliki warna, tekstur, dan aroma yang menyegarkan. Daunnya kaya akan vitamin A, vitamin B, dan vitamin C yang banyak manfaat untuk kesehatan tubuh [15].



Gambar 2.1 Gambar Selada [16]

Tanaman selada (*Lactuca sativa* L.) memiliki jenis yang beragam. Selada dikelompokkan berdasarkan kelompok kultivarnya yaitu *loose leaf*, *romaine/cos*, *crisphead*, *butterhead*, *summercrisp*, *stem*, *oilseed* [17].

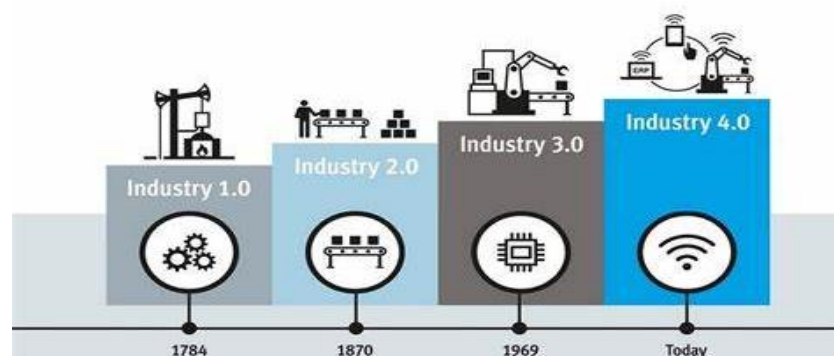
- 1) *Loose Leaf* (Selada Daun) tidak membentuk kepala, dan daun-daunnya bisa berbentuk dan berwarna beragam.
- 2) *Romaine/Cos* (Selada Tutup): memiliki daun lebih keras dan berwarna hijau gelap yang membentuk kepala yang lebih panjang dan rapat.
- 3) *Crisphead* (Selada Kepala) memiliki kepala yang sangat padat dan keras.
- 4) *Butterhead* (Selada Mentega) memiliki daun yang lembut, berwarna hijau muda, dan membentuk kepala longgar.

- 5) *Summercrisp* adalah kultivar selada yang tahan panas dan memiliki daun yang tumbuh dalam bentuk daun longgar dengan tekstur sedang antara selada mentega dan selada tutup.
- 6) *Stem* (Selada Batang) memiliki ciri khas tangkai atau batang yang lebih besar dan keras yang tumbuh di tengah daun-daunnya.
- 7) *Oilseed* (Selada Biji Minyak): Selada biji minyak adalah jenis selada yang ditanam terutama untuk ekstraksi minyak dari bijinya [17].

Dari tujuh kelompok tersebut, tiga diantaranya merupakan yang paling banyak dibudidayakan yaitu selada daun, selada tutup, dan selada kepala, sebagai contoh selada keriting lokal termasuk dalam jenis selada daun [15]. Selada keriting dapat tumbuh di dataran tinggi maupun dataran rendah. Selada keriting umumnya ditanam menjelang akhir musim hujan karena termasuk tanaman yang tidak tahan terhadap hujan. Pada musim kemarau, tanaman ini memerlukan penyiraman yang teratur. Selain tidak tahan terhadap hujan selada keriting juga tidak tahan terhadap cuaca yang terlalu panas. Sehingga selada keriting dinilai menjadi salah satu pilihan terbaik jika ingin menanam sayuran dengan sistem hidroponik [16].

2.2.2 Revolusi Industri

Revolusi industri adalah perubahan besar pada manufaktur dengan mengubah metode produksi tradisional yang digantikan oleh mesin. Perubahan tersebut pertama kali terjadi dalam periode antara tahun 1750 sampai 1850 yang menyebabkan terjadinya perubahan secara besar-besaran di bidang pertanian, manufaktur, pertambangan, transportasi, dan teknologi [18]. *The Fourth Industrial Revolution* menyatakan bahwa dunia telah mengalami empat tahapan revolusi [19].



Gambar 2.2 Perkembangan Revolusi Industri [18]

Tahap pertama dikenal sebagai Revolusi Industri 1.0 terjadi pada abad ke 18 melalui penemuan mesin uap oleh James Watt. Mesin uap menggantikan tenaga manusia dan hewan sebagai sumber utama energi, memungkinkan produksi barang secara masif di pabrik-pabrik. Ini membawa perubahan besar dalam sektor pertanian, manufaktur, dan transportasi [19].

Revolusi Industri 2.0, yang terjadi pada abad ke-19 hingga awal abad ke-20, menandai tonggak sejarah dengan penerapan listrik dan perkembangan jalur kereta api. Periode ini ditandai oleh transformasi besar dalam sektor industri yang didukung oleh penggunaan listrik secara luas dalam proses produksi. Penggunaan listrik membawa perubahan paradigma dalam cara pabrik-pabrik mengelola energi, mengurangi biaya produksi, dan secara signifikan meningkatkan efisiensi operasional. Tidak hanya dalam konteks produksi, Revolusi Industri 2.0 juga memberikan dampak yang signifikan pada sektor transportasi. Munculnya jalur kereta api dan kendaraan bermesin menjadi kunci dalam meningkatkan efisiensi dan konektivitas antar lokasi produksi. Transportasi yang lebih efisien ini memfasilitasi distribusi barang dengan cepat dan ekonomis, membuka peluang perdagangan yang lebih besar [19].

Revolusi Industri 3.0 yang terjadi pada sekitar tahun 1970 dengan ditandai oleh pengenalan teknologi komputer dan otomatisasi. Komputer mulai digunakan dalam berbagai aspek kehidupan, termasuk proses produksi di pabrik-pabrik. Sistem komputer terhubung menjadi jantung operasi industri. Dalam periode ini, terjadi pergeseran paradigma yang signifikan di mana komputerisasi tidak hanya menggantikan pekerjaan manusia secara mekanis, tetapi juga memungkinkan otomatisasi tugas-tugas yang sebelumnya memerlukan intervensi manusia secara terus-menerus. [19].

Terakhir, Revolusi Industri 4.0 yang terjadi pada tahun 2010-an dengan penggabungan teknologi digital, *artificial intelligence*, dan IoT ke dalam proses industri. Interkoneksi dan pertukaran data yang lebih canggih memungkinkan produksi yang lebih efisien dan cerdas. Pabrik-pabrik yang terhubung secara digital dapat mengoptimalkan produksi, memantau inventaris secara *real-time*, dan merespons dengan cepat terhadap perubahan pasar [19].

Setiap tahap Revolusi Industri membawa dampak besar pada masyarakat, ekonomi, dan cara kerja industri. Masing-masing periode ini dicirikan oleh adopsi teknologi baru yang mendefinisikan ulang cara produksi, distribusi, dan konsumsi barang dan jasa. Revolusi industri terus berkembang seiring waktu, dan dampaknya masih dirasakan dalam transformasi industri dan kehidupan sehari-hari [19].

2.2.3 Smart Agriculture

Smart agriculture atau bisa juga disebut *smart farming* merupakan penerapan teknologi dan inovasi dalam sektor pertanian dengan menerapkan perkembangan teknologi informasi dan komunikasi (TIK) dengan maksud untuk meningkatkan produktivitas, nilai tambah, daya saing dan keuntungan secara berkelanjutan. *Smart agriculture* menggunakan sebuah metode pertanian dengan memanfaatkan teknologi seperti *artificial intelligence* untuk memudahkan petani melakukan pekerjaan. Konsep *smart agriculture* bertujuan untuk memperbaiki proses produksi pertanian dengan memanfaatkan teknologi seperti *big data*, *machine learning*, IoT, dan teknologi lainnya dengan tujuan untuk mencapai hasil pertanian yang lebih efisien dan berkelanjutan [20].

Dalam praktiknya, *smart agriculture* dapat mencakup penggunaan drone untuk pemetaan lahan dan pemantauan tanaman, otomatisasi pertanian dengan traktor dan mesin yang dapat dikendalikan secara otonom, serta penggunaan robot dalam pemeliharaan tanaman dan panen. Integrasi *artificial intelligence* juga memainkan peran penting dalam menganalisis data pertanian yang besar dan kompleks untuk mengidentifikasi tren dan pola yang dapat membantu petani mengoptimalkan hasil panen mereka [21].

Beberapa elemen kunci dari *smart agriculture* melibatkan penggunaan teknologi canggih dan integrasi data untuk meningkatkan efisiensi pertanian. Pertama, sensor yang terpasang di lahan pertanian mengukur berbagai parameter seperti suhu, kelembaban tanah, dan tingkat nutrisi. Data ini dikumpulkan secara *real-time* dan dikirimkan ke platform analitik yang dapat diakses oleh petani. Kedua, teknologi IoT yang memungkinkan pengoptimalan penggunaan alat-alat pertanian secara otomatis. Selanjutnya, *drone* dan satelit digunakan untuk pemantauan udara, pemetaan lahan, dan analisis citra. Ini membantu petani mengidentifikasi masalah tanaman, seperti serangan hama atau penyakit pada tahap

awal sehingga dapat dilakukan tindakan pencegahan. Penggunaan *artificial intelligence* dan *machine learning* untuk menganalisis data besar memungkinkan prediksi hasil panen dan memberikan rekomendasi yang lebih baik terkait pemeliharaan tanaman. Otomatisasi pertanian merupakan salah satu elemen kunci lainnya, di mana traktor otonom dan robot dapat digunakan untuk melakukan tugas-tugas seperti penyemprotan pestisida, pemupukan, dan bahkan panen tanaman. Secara keseluruhan, integrasi elemen-elemen tersebut dapat menciptakan ekosistem *smart agriculture* yang mampu memberikan solusi terhadap tantangan pertanian modern dan meningkatkan produktivitas serta keberlanjutan pertanian secara keseluruhan [21].

2.2.4 Hidroponik

Hidroponik berasal dari bahasa Yunani, *hydro* memiliki arti air dan *ponos* berarti daya atau kerja. Sehingga secara harfiah hidroponik dapat diartikan sebagai air yang bekerja. Hal tersebut merujuk pada budidaya tanaman dengan memanfaatkan pengelolaan air tanpa menggunakan tanah sebagai media tanam dan menerima nutrisi mereka melalui larutan air yang kaya akan unsur hara [22]. Pertanian hidroponik menekankan pemenuhan kebutuhan nutrisi dengan air sebagai sumber nutrisi utama. Meskipun tanpa melibatkan tanah, tanaman hidroponik mampu tumbuh dan bahkan dapat memiliki kualitas yang lebih baik daripada tanaman yang tumbuh di tanah [23].



Gambar 2.3 Budidaya Hidroponik [23]

Sistem hidroponik ini memiliki beberapa komponen utama yang terdiri dari larutan nutrisi, media tanam, serta sistem irigasi dan drainase. Larutan nutrisi merupakan campuran antara air dan nutrisi yang dibutuhkan untuk pertumbuhan tanaman. Nutrisi ini dapat disesuaikan kebutuhannya sesuai dengan jenis tanaman yang dibudidayakan. Media tanam merupakan tempat tanaman ditanam dan akar mereka menyebar, media tanam dapat berupa arang sekam, *spons*, *rockwool*, *cocopeat*, dan lain sebagainya. Selain itu terdapat pula sistem irigasi dan drainase yang digunakan untuk menjaga suplai air dan nutrisi yang stabil serta menghilangkan kelebihan air yang dapat merusak tanaman [24].

Dalam budidaya tanaman hidroponik ada beberapa tahapan yang harus dilalui, yakni tahap penyemaian, penanaman, dan pembesaran. Tahap pertama dalam menanam hidroponik adalah penyemaian bibit atau biji tanaman. Biji ditanam dalam media penyemaian yang memungkinkan benih tumbuh. Selama tahap penyemaian, bibit membutuhkan perawatan khusus termasuk penyiraman yang cermat, kelembaban yang terjaga, dan suhu yang sesuai. Setelah bibit mencapai ukuran yang cukup untuk ditransplantasikan ke sistem hidroponik utama, tahap penanaman dimulai. Bibit yang sudah tumbuh dari media penyemaian akan ditempatkan dalam media tumbuh utama di sistem hidroponik. Media tumbuh ini dapat berupa substrat inert seperti serat kelapa atau *rockwool*, atau dapat berupa air dengan larutan nutrisi. Tahap terakhir adalah pembesaran, di mana tanaman hidroponik akan tumbuh dan berkembang hingga siap untuk panen [22].

Penggunaan metode hidroponik dalam budidaya tanaman sendiri memiliki beberapa keuntungan diantaranya [23].

- 1) Menciptakan sayuran sehat tanpa pestisida
- 2) Pemakaian pupuk atau unsur hara lebih efisien.
- 3) Umur panen lebih cepat dan penanganan pasca panen lebih mudah.
- 4) Metode hidroponik dapat diterapkan pada lahan atau ruangan terbatas.
- 5) Perawatan lebih praktis dan bebas dari masalah seperti hama dan penyakit.
- 6) Tidak ada risiko banjir, erosi, kekeringan atau ketergantungan dengan cuaca sehingga tanaman dapat dibudidayakan di luar musim.
- 7) Harga jual tanaman hidroponik lebih tinggi dibandingkan tanaman non hidroponik [23].

2.2.5 Citra Digital

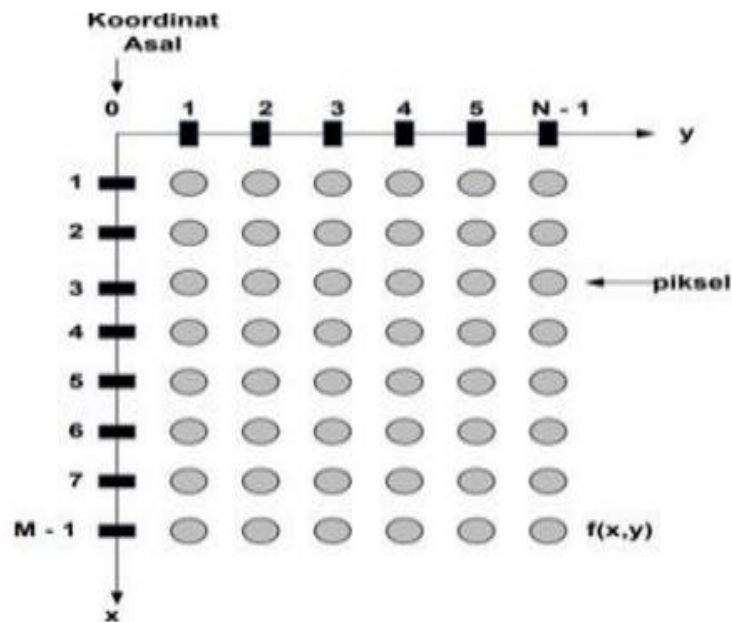
Citra merupakan salah satu bentuk informasi penting yang diperlukan manusia selain teks, suara, gambar dan video. Setiap manusia dapat merepresentasikan informasi yang terkandung dalam sebuah citra dengan cara yang berbeda. Citra sendiri dapat dibedakan menjadi dua yaitu citra analog dan citra digital. Citra analog dihasilkan dari alat akuisisi citra analog, Citra analog dihasilkan melalui alat seperti kamera analog atau mata manusia. Misalnya, gambar yang diambil oleh kamera analog atau gambaran yang diterima oleh mata manusia termasuk dalam kategori citra analog [25]. Di sisi lain, citra digital adalah gambar dua dimensi yang terbentuk dari citra analog kontinu melalui proses sampling. Proses ini membagi citra analog menjadi baris dan kolom, menghasilkan citra diskrit. Citra digital dapat diolah oleh komputer dan disimpan sebagai angka yang menunjukkan intensitas di setiap piksel [26].

Saat ini terdapat banyak perubahan yang berkaitan dengan kualitas citra, termasuk citra beresolusi rendah, warna yang memudar, dan berbagai masalah lainnya. Untuk membuktikan perubahan yang terjadi pada citra digital dapat dilakukan dengan melakukan penyamaan isi menggunakan pendekatan metadata dan proses pengolahan citra. Pengolahan citra adalah proses mengolah piksel-piksel dalam citra digital untuk meningkatkan kualitas suatu gambar. Proses ini mencakup peningkatan kontras, perubahan warna, restorasi citra, serta berbagai transformasi seperti translasi, rotasi, dan skala. Selain itu, pengolahan juga melibatkan pemilihan fitur-fitur khusus dalam citra, yang juga disebut sebagai citra ciri atau *feature images* [25]. Secara umum pengolahan citra digital adalah proses penggunaan komputer untuk memproses gambar dua dimensi. Citra digital merupakan suatu *array* (larik) yang berisi nilai-nilai *real* atau kompleks yang dapat direpresentasikan dalam bentuk deretan bit tertentu [26].

Dalam melakukan pemrosesan citra, dibutuhkan identifikasi dan klasifikasi ciri dari gambar sebagai langkah untuk membedakan dan mengelompokkannya. Berikut merupakan beberapa karakteristik mendasar dari citra [27].

- 1) Warna: Ilustrasi dari karakteristik warna ini dapat dilihat dalam bentuk histogram dengan tiga komponen warna, yaitu RGB.

- 2) Bentuk: Dalam mengekstrak ciri bentuk, informasi dapat diperoleh melalui tepi citra atau dengan memanfaatkan momen dari citra. Contoh besaran momen yang umum dalam ciri bentuk yaitu transformasi fourier. Untuk mendapatkan ciri bentuk, proses ini melibatkan beberapa langkah, seperti penggunaan *threshold*, segmentasi, deteksi tepi, dan perhitungan momen, termasuk mean, median, serta standar deviasi dari setiap bagian citra.
- 3) Tekstur: Ciri tekstur dalam citra berkaitan dengan karakteristik seperti tingkat kekasaran, granularitas, dan keteraturan dari susunan struktural piksel. Ciri-ciri dimanfaatkan dalam tahap klasifikasi, segmentasi, dan interpretasi citra[27].



Gambar 2.4 Koordinat dalam Citra Digital [26]

Dapat dilihat pada gambar 2.5, sebuah citra digital direpresentasikan dalam matriks dua dimensi dengan M kolom dan N baris. Sedangkan setiap perpotongan antara kolom dan baris tersebut disebut piksel (*picture element*) yang merupakan elemen terkecil dari sebuah citra. Setiap piksel mempunyai dua parameter, yaitu koordinat (x, y) dan intensitas (warna). Dalam representasi ini, nilai $f(x, y)$ mencerminkan besar intensitas yang menunjukkan warna dari piksel yang terletak pada koordinat (x, y). Dengan demikian, citra berukuran m x n piksel direpresentasikan dengan koordinat (m-1, n-1), contohnya pada koordinat (0,0) [26].

2.2.6 Artificial Intelligence (Kecerdasan Buatan)

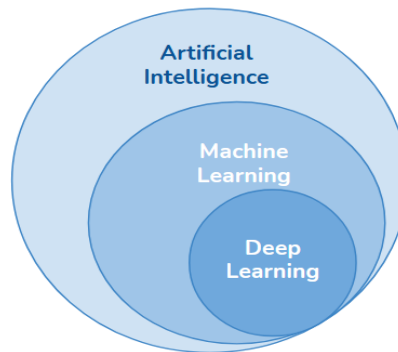
Artificial Intelligence (Kecerdasan buatan atau AI) adalah upaya untuk meniru kecerdasan manusia dalam mesin seperti komputer atau robot. AI mencakup berbagai teknik yang bertujuan untuk membuat komputer dapat meniru fungsi kognitif manusia, seperti pembelajaran, penalaran, komunikasi, dan pengambilan keputusan. AI diprogram untuk menjalankan tugas yang biasanya memerlukan kecerdasan manusia, seperti memecahkan masalah dan belajar. Sistem AI bisa berbentuk perangkat lunak atau perangkat keras yang dibuat oleh manusia untuk beroperasi di dunia fisik atau digital. Ini melibatkan memahami lingkungan melalui pengumpulan data, menafsirkan data yang terstruktur atau tidak terstruktur, menggunakan pengetahuan untuk penalaran, memproses informasi, dan membuat keputusan untuk mencapai tujuan tertentu [28].

Menurut Coppin, AI merujuk pada kemampuan mesin untuk beradaptasi dengan situasi baru, menyelesaikan masalah, memberikan jawaban atas pertanyaan, merencanakan tindakan, dan melakukan fungsi lain yang seringkali memerlukan tingkat kecerdasan yang biasanya terlihat pada manusia. Adapun beberapa jenis *artificial intelligence* secara garis besar terbagi menjadi tujuh bagian, diantaranya *Machine Learning*, *Natural Language Processing*, *Expert System*, *Vision Speech*, *Planing*, dan *Robotics*. Pembagian jenis *artificial intelligence* tersebut dimaksudkan untuk mempersempit ruang lingkup saat pengembangan atau belajar *artificial intelligence*, karena pada dasarnya *artificial intelligence* memiliki ruang lingkup yang sangat luas. Sedangkan pada penelitian ini *deep learning* berada pada subbidang dari *machine learning* yang menggunakan arsitektur *neural networks* yang mendalam untuk mengatasi tugas-tugas kompleks, seperti pengenalan gambar, pengenalan suara, dan pemrosesan bahasa alami [29].

2.2.7 Deep Learning

Deep memiliki arti dalam dan *learning* yang berarti pembelajaran, sehingga secara umum *deep learning* dapat diartikan sebagai pembelajaran mendalam. *Deep learning* adalah cabang dari *artificial intelligence* yang menggunakan jaringan saraf tiruan dengan banyak lapisan untuk memodelkan dan memahami data. Arsitektur

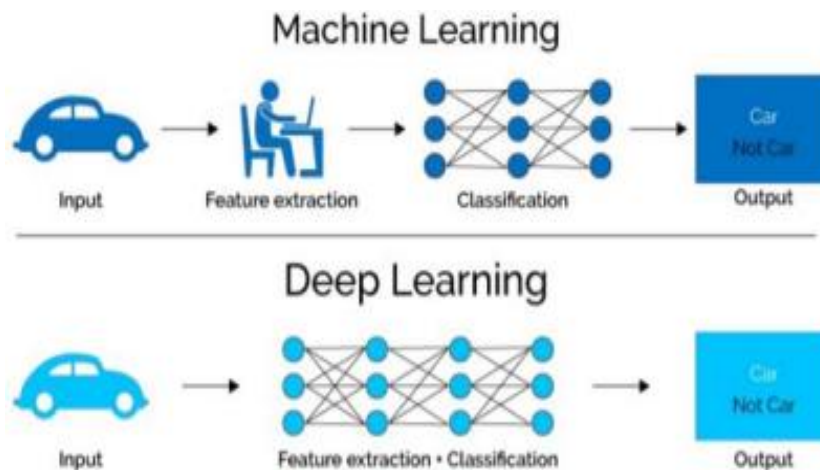
secara umum *artificial intelligence*, *machine learning*, dan *deep learning* dapat dilihat pada gambar berikut [30].



Gambar 2.5 Posisi *Deep Learning* [30]

Gambar 2.6, menunjukkan bahwa *artificial intelligence* melingkupi *machine learning* dan *deep learning*, sedangkan *machine learning* melingkupi *deep learning*. Gambar tersebut menunjukkan bahwa *deep learning* menjadi bagian terdalam *machine learning* sehingga menyebabkan pembelajaran *deep learning* lebih mendalam. Model *deep learning* terdiri dari lapisan kompleks yang memungkinkan mereka untuk meniru proses kerja otak manusia yang memungkinkan untuk mengklasifikasikan data yang besar dan kompleks, seperti gambar, teks, suara, dan video [30]. *Deep learning* ini digunakan untuk mengembangkan model tingkat tinggi yang memiliki kompleksitas tinggi dengan struktur yang rumit, terdiri dari serangkaian transformasi non-linear.

Dengan menggunakan *deep learning* memungkinkan ekstraksi informasi dari data melalui lapisan-lapisan unit. *Deep learning* dipilih karena kemampuannya dalam mengadaptasi dan mempelajari fitur-fitur dengan cepat, sementara fitur-fitur yang dikembangkan secara manual seringkali terlalu spesifik, tidak lengkap, dan sulit untuk diterapkan. *Deep learning* memiliki keunggulan sebagai kerangka pembelajaran yang dapat disesuaikan, *universal*, mudah dipelajari, dan dapat merepresentasikan dunia, informasi visual, dan bahasa. Selain itu *deep learning* juga penting untuk mencapai representasi yang tersebar secara luas untuk generalisasi non-lokal. Meskipun *deep learning* termasuk dalam kategori algoritma *machine learning*, cara kerjanya cukup berbeda. Perbedaannya dapat dilihat dalam gambar 2.7 [31].

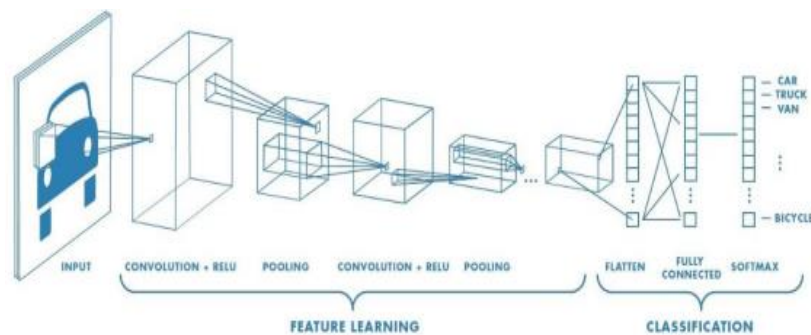


Gambar 2.6 Perbedaan *Machine Learning* dan *Deep Learning* [31]

Pada Gambar 2.7 ilustrasi perbedaan *machine learning* dan *deep learning* dapat dengan jelas diamati. Pada *machine learning* terdapat tahap *feature extraction* yang bertujuan untuk mengekstrak ciri-ciri unik dari data. Namun pada *deep learning* tahap *feature extraction* dan klasifikasi dilakukan bersama tidak seperti *machine learning* yang dilakukan terpisah. Klasifikasi adalah proses mengidentifikasi atau menentukan kelas yang sesuai dengan suatu input yang diberikan [31]. *Feature extraction* atau ekstraksi fitur merupakan proses pengurangan dimensi dengan mengelompokkan *dataset* awal menjadi kelompok yang lebih mudah dikelola dan diproses. Secara sederhana, ekstraksi fitur menggabungkan variable dari data ke dalam fitur yang pada akhirnya mengurangi jumlah data yang perlu diproses. Dalam penggunaan *deep learning* sering timbul permasalahan yang dihadapi yaitu semakin besar ukuran *neural network*, maka semakin besar pula beban komputasi yang dibutuhkan. Namun dengan adanya *Graphic Processing Unit* (GPU) masalah tersebut dapat terselesaikan dan membuat proses *training* data menjadi lebih cepat [32]. Dengan adanya perbedaan antara *machine learning* dan *deep learning* menyebabkan penggunaannya disesuaikan dengan kebutuhan tertentu. Penggunaan *deep learning* umumnya dipilih saat data yang akan diolah memiliki ukuran yang besar dan memerlukan infrastruktur kelas atas untuk melatih model dalam waktu yang wajar. Metode ini juga berguna ketika fitur-fitur yang dianalisis kompleks seperti klasifikasi gambar, *natural language processing*, dan *speech recognition* [31].

2.2.8 Convolutional Neural Network (CNN)

CNN adalah tipe arsitektur *neural network* yang dikembangkan khusus untuk mengolah data seperti data citra. CNN menggunakan lapisan konvolusi untuk mengekstrak fitur dari data input, yang memungkinkan model untuk memahami pola spasial [33]. CNN sendiri merupakan sebuah arsitektur yang dapat dilatih yang terbagi menjadi tiga bagian yaitu konvolusi *layer* ditambah fungsi aktivasi *Rectified Linear Unit (ReLU)*, *pooling layer*, dan *fully connected layer* digunakan untuk menerima nilai dan mengklasifikasikan nilai tersebut pada salah satu kelas tertentu. Arsitektur ini sangat efektif dalam tugas pengenalan gambar, pengolahan citra, dan tugas-tugas lain yang melibatkan data *grid*. CNN telah berhasil diterapkan dalam berbagai aplikasi, seperti pengenalan wajah, klasifikasi objek, dan pemrosesan gambar medis. Berikut adalah jaringan arsitektur CNN [34]:



Gambar 2.7 Arsitektur CNN [33]

2.2.9 Convolutional Layer

Convolutional layer merupakan bagian dari tahap pada arsitektur CNN. Fungsi utamanya adalah mengekstraksi fitur-fitur dari input menggunakan operasi konvolusi. Lapisan ini terdiri dari sejumlah filter atau *kernel*, di mana setiap *kernel* melakukan operasi konvolusi pada input dan menghasilkan satu *channel* dalam *output* yang disebut sebagai *Feature Map*. Filter ini bergerak melintasi input, mengalikan nilai-nilai piksel dengan bobot-bobot (*weights*) filter, dan menghasilkan representasi-fitur yang berfokus pada pola-pola lokal dalam citra. Pada *convolutional layer*, ReLU adalah salah satu fungsi aktivasi yang umum digunakan dalam CNN [33].

2.2.10 Pooling Layer

Pooling adalah proses pengurangan ukuran matriks dengan menerapkan operasi *pooling* yang biasanya dilakukan dengan mengambil nilai maksimum atau rata-rata dari sekumpulan nilai piksel. *Pooling Layer* digunakan untuk mengurangi dimensi spasial yang dihasilkan oleh lapisan konvolusi. Dua jenis *pooling* yang umum digunakan adalah *Max Pooling* dan *Average Pooling*. Dalam *Max Pooling*, nilai maksimum diambil dari setiap area beririsan di *Feature Map*, sementara dalam *Average Pooling*, rata-rata nilai diambil. Tujuan dari lapisan *pooling* adalah untuk mengurangi jumlah parameter dan komputasi, menjaga objek tetap sama meskipun posisinya bergeser secara translasi, dan memberikan kemampuan model untuk memfokuskan pada fitur-fitur penting dalam citra. Lapisan *pooling* membantu menjaga informasi penting dan mempercepat proses pelatihan [33].

2.2.11 Fully Connected Layer

Fully Connected Layer adalah sebuah lapisan yang terdiri dari neuron-neuron yang terhubung sepenuhnya satu sama lain. Perbedaan lapisan *Fully Connected* dengan lapisan konvolusi yaitu neuron di lapisan konvolusi terhubung hanya ke daerah tertentu pada input, sementara lapisan *Fully Connected* memiliki neuron yang secara keseluruhan terhubung. Lapisan ini umumnya ditempatkan di bagian akhir arsitektur CNN. Input dari lapisan ini berasal dari *flatten* atau hasil *pooling* yang kemudian dihubungkan ke setiap neuron dalam lapisan ini. Lapisan ini berfungsi untuk menyatukan informasi dari fitur-fitur yang telah diekstraksi sebelumnya dan melakukan tugas klasifikasi atau regresi. Penggunaan fungsi aktivasi *binary step* pada lapisan terakhir ini lebih cocok untuk tugas klasifikasi biner (dua kelas). Fungsi *binary step* menghasilkan *output* antara 0 dan 1 [33].

2.2.12 Fungsi Aktivasi

Fungsi aktivasi menggambarkan hubungan antara tingkat aktivitas internal (*summation function*) yang dapat berbentuk linear atau non-linear. Fungsi ini digunakan untuk menentukan apakah neuron akan diaktifkan atau tidak. Dalam konteks *deep learning* dan *neural network*, fungsi aktivasi digunakan untuk memperkenalkan non-linearitas pada model. Tanpa fungsi aktivasi, lapisan-lapisan dalam *neural network* akan bersifat linear, dan rangkaian tersebut hanya akan dapat

merepresentasikan transformasi linear dari input ke *output*. Dengan adanya fungsi aktivasi, model dapat menangkap dan mempelajari pola-pola yang lebih kompleks dalam data. Beberapa fungsi aktivasi umum yang digunakan dalam *deep learning* diantaranya ReLU, *Binary Step*, *Sigmoid*, *Tanh*, dan lain sebagainya [35].

2.2.13 Fungsi Aktivasi *Rectified Linear Unit* (ReLU)

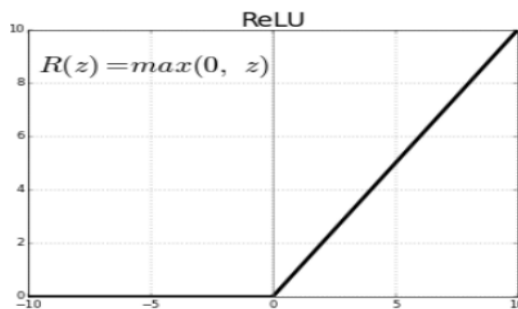
Fungsi aktivasi yang cukup terkenal adalah ReLU. Pada dasarnya fungsi ReLU membatasi nilai-nilai input dari 0 hingga tak terhingga. Berikut ini adalah persamaan dari fungsi aktivasi ReLU [33].

$$f(x) = \max(0, x) \quad (2.1)$$

Dimana, $f(x)$ = fungsi ReLU

x = nilai batas ReLU

$\max(0, x)$ = jika $x \leq 0$ maka $x = 0$ dan jika $x > 0$ maka $x = x$



Gambar 2.8 Fungsi Aktivasi ReLU [33]

Dalam fungsi ini jika masukan dari neuron bernilai negatif, maka nilai tersebut akan diubah kedalam nilai 0, dan jika masukan dari neuron bernilai positif maka keluaran dari neuron berupa nilai aktivasi itu sendiri. Fungsi aktivasi ini memiliki keunggulan dalam mempercepat proses konfigurasi jika dibandingkan dengan fungsi *sigmoid* dan *tanh*. Namun kelemahan dari aktivasi ini adalah kemungkinan aktivasi menjadi tidak stabil selama proses pelatihan, dan ini dapat menyebabkan unit menjadi tidak aktif [33].

2.2.14 Fungsi Aktivasi *Binary Step*

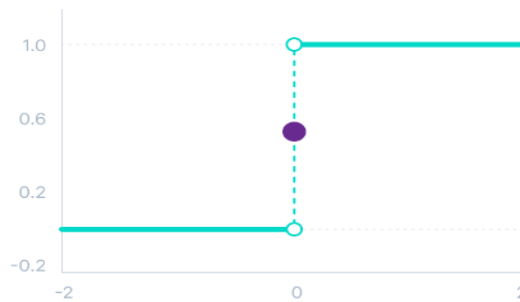
Fungsi aktivasi *binary step* adalah jenis fungsi aktivasi yang sederhana dan umumnya digunakan pada masalah klasifikasi biner di dalam *deep learning*. Fungsi

ini mengubah input menjadi dua nilai keluaran yaitu 0 atau 1. Berikut ini adalah persamaan fungsi aktivasi *Binary Step* [36]:

$$f(x) = \begin{cases} 0, & \text{for } x < 0 \\ 1, & \text{for } x \geq 0 \end{cases} \quad (2.2)$$

Dimana, $f(x)$ = fungsi *Binary Step*

x = nilai input *Binary Step*



Gambar 2.9 Fungsi Aktivasi *Binary Step* [36]

Pada fungsi ini jika masukan dari neuron bernilai negatif, maka nilai tersebut akan diubah kedalam nilai 0, dan jika masukan dari neuron bernilai positif maka keluaran nilainya adalah 1. Kelemahan dari fungsi ini yaitu *binary step* tidak dapat memberikan *output* dengan nilai ganda contohnya, tidak dapat digunakan untuk masalah klasifikasi multikelas. Namun kelebihan dari fungsi aktivasi ini dapat menormalkan keluaran dari tiap neuron dan memberikan prediksi yang jelas yaitu antara 0 atau 1 [36].

2.2.15 *Training Set, Validation Set, dan Testing Set*

Terdapat dua istilah penting dalam pengembangan model *deep learning* yaitu pelatihan (*training*) dan pengujian (*testing*). Pelatihan merupakan proses membangun model, sementara pengujian adalah langkah untuk mengevaluasi performa model pembelajaran. *Dataset* merupakan kumpulan data yang berfungsi sebagai sampel dalam statistik. Sampel ini digunakan untuk membentuk model dan mengevaluasi model *deep learning*. *Dataset* dapat dikelompokkan menjadi tiga jenis yang tidak saling tumpang tindih, yang berarti satu sampel pada satu set tidak ada dalam set lainnya. Tiga jenis tersebut diantaranya terdiri dari [37]:

- 1) *Training set* adalah kumpulan data yang digunakan untuk melatih atau membangun model.

- 2) *Validation set* adalah kumpulan data yang digunakan untuk mengoptimalkan model selama pelatihan. Model dilatih menggunakan *training set* dan dan kinerjanya dievaluasi menggunakan *validation set*.
- 3) *Testing set* adalah kumpulan data yang digunakan untuk mengevaluasi model setelah proses latihan selesai. Penting dicatat bahwa *testing set* adalah data yang belum pernah dilihat sebelumnya. Sehingga baik model maupun manusia tidak seharusnya melihat sampel ini selama proses pelatihan karena dapat menyebabkan bias [37].

2.2.16 Accuracy dan Loss

Accuracy dan *Loss* adalah dua metrik yang paling dikenal dan dibahas dalam pembelajaran mesin. *Accuracy* dan *Loss* menjadi salah satu poin penting dalam klasifikasi citra. *Accuracy* merupakan metode untuk mengukur kinerja model klasifikasi yang hasilnya dapat dinyatakan dalam persentase. *Accuracy* adalah jumlah prediksi di mana nilai prediksi sama dengan nilai sebenarnya. Ini bersifat biner (benar/salah) untuk sampel tertentu. *Loss* atau kesalahan diperoleh dari perhitungan pada *training* dan *testing* yang hasilnya mencerminkan seberapa baik model saat melakukan *training* dan *validation*. Nilai *loss* menunjukkan ukuran yang menunjukkan seberapa besar kesalahan dalam klasifikasi citra oleh model, dimana semakin rendah nilai *loss* maka semakin baik model yang dihasilkan. Namun hal tersebut tidak berlaku jika model mengalami *overfitting*. Nilai *loss* tidak dinyatakan dalam persen seperti *accuracy* karena menggambarkan jumlah kesalahan dari setiap contoh dalam validasi. Tujuan utama dari proses pembelajaran model adalah untuk mengurangi nilai *loss* dengan menyesuaikan *weight vector* melalui berbagai metode optimasi yang berbeda seperti *backpropagation* pada *neural network* [38].

2.2.17 Confusion Matrix

Confusion Matrix adalah tabel yang digunakan untuk menganalisis kinerja model klasifikasi dengan membandingkan prediksi model terhadap nilai sebenarnya dari data uji. Tabel ini mencatat jumlah prediksi yang benar (*true positive* dan *true negative*) serta jumlah prediksi yang salah (*false positive* dan *false negative*). *Confusion matrix* berfungsi untuk mengetahui tentang seberapa baik suatu model dapat mengklasifikasikan data pada kelas tertentu [39].

Tabel 2.2 Confusion Matrix

	<i>Predicted Positive</i>	<i>Predicted Negative</i>
<i>Actual Positive</i>	<i>True Positive (TP)</i>	<i>False Negative (FN)</i>
<i>Actual Negative</i>	<i>False Positive (FP)</i>	<i>True Negative (TN)</i>

Informasi yang terdapat dalam tabel *confusion matrix* di atas dapat diuraikan sebagai berikut [39].

- 1) *True Positive (TP)*: Jika hasil klasifikasi menghasilkan kelas prediksi benar (*true*), dan pada pengujian sebenarnya kelas tersebut benar (*positive*). Ketika keduanya benar, disebut sebagai *True Positive*.
- 2) *False Positive (FP)*: Jika hasil klasifikasi menghasilkan kelas prediksi yang salah (*false*), tetapi pada pengujian sebenarnya kelas tersebut benar (*positive*).
- 3) *False Negative (FN)*: Jika hasil klasifikasi menghasilkan kelas prediksi yang salah (*false*), dan pada pengujian sebenarnya kelas tersebut salah (*negative*). Ketika keduanya salah, disebut sebagai *False Negative*.
- 4) *True Negative (TN)*: Jika hasil klasifikasi menghasilkan kelas prediksi benar (*true*), tetapi pada pengujian sebenarnya kelas tersebut salah (*negative*) [39].

Proses klasifikasi melibatkan perhitungan matriks data dengan memanfaatkan *confusion matrix* yang merepresentasikan hasil dari model klasifikasi yang digunakan. Pada tahap evaluasi akan menampilkan nilai-nilai seperti *accuracy*, *precision*, dan *recall*[34].

1) *Accuracy*

Matriks ini menunjukkan sejauh mana model berhasil memprediksi kelas dengan benar. Ini adalah metode yang paling umum digunakan, tetapi dapat menjadi tidak akurat ketika data tidak seimbang. Persamaan untuk matriks ini adalah sebagai berikut [34]:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{2.3}$$

2) *Precision*

Matriks ini mengukur sejauh mana model secara akurat memprediksi kelas positif, namun tidak memberikan informasi tentang berapa banyak kasus positif

yang diabaikan oleh prediksi model. Persamaan untuk matriks ini adalah sebagai berikut [34]:

$$Precision = \frac{TP}{TP+FP} \quad (2.4)$$

3) *Recall*

Matriks ini menunjukkan seberapa baik model memprediksi kelas positif yang sebenarnya. Namun, *recall* tidak memberikan gambaran tentang seberapa baik model memprediksi kasus positif secara keseluruhan. Persamaan untuk matriks ini adalah sebagai berikut [34]:

$$Recall = \frac{TP}{TP+FN} \quad (2.5)$$