

## BAB 2

### DASAR TEORI

#### 2.1 KAJIAN PUSTAKA

Penelitian [11] mengidentifikasi jenis penyakit pada daun tanaman padi dengan menggunakan *Adaptive Neuro Fuzzy Inference System* (ANFIS) berdasarkan tekstur. Penelitian ini membedakan tiga jenis penyakit, yaitu *bacterial leaf blight*, *brown spot*, dan *leaf smut*, dengan masing-masing kelas memiliki 40 citra berekstensi jpeg, sehingga totalnya terdapat 120 citra. Nilai akurasi rata-rata sebesar 98,5%.

Penelitian [12] mengenai klasifikasi penyakit tanaman padi dengan menerapkan metode *transfer learning* dengan *pretrained* model *layer*, dan *BatchNormalization layer*. Model yang digunakan adalah ResNet101, yang dimodifikasi dengan penambahan lapisan *fully connected* untuk meningkatkan hasil performa *accuracy* dalam klasifikasi penyakit padi. Data yang digunakan terdiri dari tiga kelas yaitu *Bacterial leaf blight*, *Brown spot*, dan *Leaf smut*, dengan setiap kelas terdiri dari 40 gambar, sehingga totalnya terdapat 120 gambar.

Penelitian [13] mengidentifikasi penyakit tanaman padi dengan menggunakan metode *K-Nearest Neighbor* (KNN) untuk klasifikasi, berdasarkan ekstraksi fitur *Gray Level Co-Occurrence Matrix* (GLCM) dengan citra yang telah dikonversi menjadi citra keabu-abuan (*grayscale*). Jumlah data yang digunakan berjumlah 240 citra yang diperoleh dari *UCI Machine Learning Repository*, yang mencakup tiga jenis penyakit padi diantaranya *bacterial leaf blight*, *brown spot*, dan *leaf smut*. Penelitian ini berhasil mencapai akurasi tertinggi sebesar 93.3%.

Penelitian [14] mengenai klasifikasi terhadap penyakit daun padi dengan menggunakan fitur dari *Color Histogram* diekstraksi dan kemudian klasifikasikan menggunakan algoritma *Random Forest*. Dataset pada penelitian ini terdiri dari tiga kelas penyakit daun padi yaitu *bacterial leaf blight*, *brown spot*, dan *leaf smut* dengan setiap kelas terdiri dari 40 citra sehingga totalnya terdapat 120 citra dalam dataset. Penelitian ini dilakukan dengan membandingkan berbagai metode ekstraksi fitur dan algoritma untuk mencapai hasil optimal dengan akurasi yang tinggi. Hasil

akurasi tertinggi diperoleh dari metode *Color Histogram* serta algoritma *Random Forest* dengan nilai akurasi 99.65%.

Penelitian [15] mengukur dan membandingkan performa kernel dengan menggunakan algoritma *Support Vector Machine (SVM)* untuk mengklasifikasikan penyakit tanaman padi. Dataset dalam penelitian ini terdapat tiga kelas yaitu *bacterial leaf blight*, *brown spot*, dan *leaf smut*. Akurasi tertinggi yang dicapai oleh algoritma SVM dalam klasifikasi penyakit tanaman padi terjadi pada dua skenario perbandingan data latih dan data uji, yaitu 70% : 30% dan 80% : 20%, menggunakan fungsi kernel linear dengan nilai mencapai 89%.

Penelitian [16] melakukan klasifikasi penyakit pada daun padi menggunakan *K-Nearest Neighbor (KNN)* yang merupakan klasifikasi bersumber pada informasi data pembelajaran dengan jarak yang terdekat. Pada penelitian ini menggunakan data yang berjumlah 120 citra yang terbagi menjadi tiga kelas jenis penyakit diantaranya yaitu *bacterial leaf blight*, *brown spot*, dan *leaf smut*. Sistem mengklasifikasi hasil akurasi terbaik dari klasifikasi ketika  $k = 1$  dan saat performa sistem menggunakan 6 fitur terbaik yaitu *hue*, *saturation*, *value*,  $h_2$ ,  $h_3$ , dan  $h_7$  menghasilkan dengan *confusion matrix* 81.66%.

**Tabel 2. 1 Perbandingan metode dan klasifikasi**

Pengarang	Tujuan	Metode	Perbedaan
Retno Nugroho Whidhiasih <i>et al</i> (2019)	Mengidentifikasi jenis penyakit pada daun tanaman padi	<i>Adaptive Neuro Fuzzy Inference System (ANFIS)</i> berdasarkan tekstur.	Penelitian rujukan pertama menggunakan metode ANFIS berdasarkan tekstur. Sedangkan, penelitian ini menggunakan metode algoritma CNN.

Pengarang	Tujuan	Metode	Perbedaan
Ulfah Nur Oktaviana <i>et al</i> (2021)	Klasifikasi penyakit tanaman padi	Metode <i>transfer learning</i> dengan model ResNet101	Penelitian rujukan kedua memanfaatkan metode <i>transfer learning</i> dengan model ResNet101. Sedangkan, penelitian ini menggunakan metode CNN.
Arif Akbarul Huda <i>et al</i> (2022)	Identifikasi jenis penyakit tanaman padi	Algoritma GLCM dan klasifikasi KNN	Penelitian rujukan ketiga menggunakan GLCM dan KNN. Sedangkan, penelitian ini menggunakan metode CNN.
Syarifah Agustiani <i>et al</i> (2022)	klasifikasi terhadap penyakit daun padi	Metode ekstraksi fitur <i>Color Histogram</i> dan menggunakan algoritma <i>Random Forest</i> .	Penelitian rujukan ketiga dengan metode ekstraksi fitur <i>Color Histogram</i> dan menggunakan algoritma <i>Random Forest</i> . Sedangkan, penelitian ini menggunakan metode CNN.

Pengarang	Tujuan	Metode	Perbedaan
Kurniawan Saputra <i>et al</i> (2023)	klasifikasi penyakit tanaman padi	algoritma <i>Support Vector Machine</i> (SVM)	Penelitian rujukan keempat menggunakan metode SVM. Sedangkan, penelitian ini menggunakan metode CNN.
Santika Tri Hapsari S (2023)	Klasifikasi penyakit pada daun padi.	Metode KNN	Penelitian rujukan kelima menggunakan metode KNN. Sedangkan, penelitian ini menggunakan metode CNN.

## 2.2 DASAR TEORI

### 2.2.1 TANAMAN PADI

Tanaman padi atau dikenal dengan nama lain *Oryza sativa L* adalah salah satu tanaman *spermatophyte* yang bersifat hidrofit. Beras yang dihasilkan dari tanaman padi adalah bahan pangan utama yang menjadi komoditas strategis dan mempunyai peran yang sangat penting [17]. Pertumbuhan daun yang satu dengan daun berikutnya atau daun baru mempunyai selang waktu tujuh hari, dimana setiap tujuh hari berikutnya akan muncul daun tambahan [18]. Dengan mengetahui strategi yang tepat sehingga dapat diimplementasikan untuk mencegah dan mengendalikan serangan dan penyakit yang menyerang tanaman di lapangan, sedini mungkin kerusakan dan kerugian akibat virus dan penyakit dapat dihindari, sehingga mempunyai kemampuan untuk meningkatkan produktivitas [19].

Padi mempunyai umur berbeda-beda. Padi bisa dipanen berumur kurang dari 90 hari, tanaman padi bisa dipanen ketika umur lebih dari 6 bulan. Umumnya,

padi dipanen saat sekitar umur 3-4 bulan setelah tanam. Tanaman padi dapat tumbuh dan beradaptasi hampir di berbagai wilayah, termasuk dataran rendah hingga tinggi 2000 meter di atas permukaan laut, dari wilayah tropis hingga subtropis, dan dari lahan basah (rawa-rawa) hingga lahan kering (padang pasir), serta lain-lain [20].

Tanaman padi memiliki batang yang beruas-ruas dan berbuku-buku. Ruas-ruas ini kosong selain empelur yang lunak dan berwarna putih di bagian atas dekat buku. Panjang ruas batang padi bervariasi. Secara umum, terdapat tujuh dan ruas yang teratas paling panjang dan besar, kebawah cenderung lebih pendek dan kecil. Batang padi bervariasi dalam tinggi tergantung jenisnya. Secara umum, tanaman yang lebih tua memiliki tinggi yang lebih besar daripada yang lebih muda, dengan kisaran rata-rata 80 hingga 120 cm dan paling tinggi 1,5 m [21]. Karakteristik batang yang pendek dan kokoh diinginkan dalam pengembangan varietas tanaman padi yang unggul karena dapat meningkatkan ketahanan roboh atau tumbang. Tumbuhan dapat dikatakan sehat jika tumbuhan mampu melakukan fungsi fisiologi dengan baik diantaranya fungsi perkembangan, pembelahan, dan diferensiasi sel, penyerapan air, hara dapat diserap dengan baik dari dalam tanah serta distribusinya, melakukan fotosintesis dengan baik, metabolisme senyawa sintesis yang lancar, kemampuan untuk menyimpan sumber daya makanan untuk kelangsungan hidup dan proses reproduksi yang baik [22].



**Gambar 2. 1 Padi sehat**

Gambar 2.1 merupakan daun padi yang sehat, padi yang berkualitas tinggi memiliki daun yang sehat. Kualitas padi sangat penting untuk meningkatkan produksi pada musim panen.

Beras menjadi pilihan utama untuk memenuhi kebutuhan gizi dari berbagai lapisan masyarakat di negara ini karena berperan sebagai makanan pokok. Di Indonesia, rata-rata konsumsi beras per tahun mencapai 139.15 kg, jauh melebihi negara-negara maju yang hanya mengonsumsi sekitar 80-90 kg beras per jiwa tahun. Berdasarkan hasil analisis, beras mengandung seperti karbohidrat, protein, lemak, air, magnesium, besi, fosfor, seng, kalium, vitamin B1, B2, B3, B6, B9, dan serat [23]. Dalam budidaya padi upaya pengendalian sangat penting karena menyebabkan menurunnya hasil panen yang signifikan yang diakibatkan karena serangan hama dan penyakit [19].

### 2.2.2 PENYAKIT DAUN PADI

Hama dan penyakit tumbuhan adalah organisme yang mengganggu tumbuhan. Petani dirugikan karena serangan ini menyebabkan banyak kerusakan dan kehilangan panen [24]. Jamur, bakteri, dan virus dapat menyebabkan penyakit padi. Selain itu, pemicu penyakit tanaman padi adanya 3 faktor utama yaitu patogen penyebab penyakit yang ganas, tanaman inang yang rentan, dan area pendukung [19].



**Gambar 2. 2 Segitiga penyakit [22]**

Jika patogen yang ganas bertemu dengan bagian tanaman yang rentan, penyakit tidak akan terjadi, namun lingkungan tidak akan mendukung perkembangan patogen tersebut. Faktor-faktor lingkungan seperti kelembaban, suhu, cahaya matahari, dan hara tanah juga berpengaruh terhadap tumbuhan inang maupun patogen. Karenanya, ketiga faktor tersebut saling berhubungan dan dapat menyebabkan timbulnya suatu penyakit pada tumbuhan, yang dikenal sebagai "segitiga penyakit" atau *diseases triangle* yang dapat dilihat pada Gambar 2.2.

Penyakit adalah suatu proses yang penyebabnya terjadi secara terus-menerus dalam waktu yang lama, memberikan dampak pada bagian tertentu atau seluruh tanaman tidak dapat berfungsi dengan optimal. Oleh karena itu, istilah bahwa “tanaman terkena penyakit” tidak sepenuhnya tepat, karena lebih tepat menggunakan istilah "terserang" ketika organisme seperti tanaman diserang oleh jamur atau bakteri. Proses penyakit pada tumbuhan terjadi secara terus-menerus dalam jangka waktu yang panjang, sehingga kerusakan yang disebabkan secara mekanis seperti daun berlubang akibat belalang tidak dianggap penyakit. Oleh karena itu, kondisi ini disebut "kerusakan". Kerusakan pada daun yang disebabkan oleh gigitan belalang tidak dianggap sebagai penyakit, melainkan sebagai kondisi rusak. Sementara itu, kondisi dimana terbentuk bintil-bintil karena adanya jamur disebut sebagai penyakit [22]. Semuanya terjadi pada waktu dan tempat yang sama. Beberapa penyakit padi yang mengurangi produksi:

#### 1. Blas Kolar

Penyakit *rice blast* atau yang sering disebut blas kolar, disebabkan oleh jamur *Pyricularia oryzae* (Cavara). Penyakit ini menyebar melalui angin dan air yang menginfeksi bagian daun, batang, kolar, dan malai selama tanaman tumbuh [25].



**Gambar 2. 3 Penyakit blas kolar**

Pada gambar 2.3 Gejala yang dapat ditemukan pada daun adalah bercak yang berbentuk kecil tapi melebar beberapa sentimeter dengan ujungnya yang meruncing. Bercak di bagian tengah berwarna keabu-abuan, dan pinggir bercak berwarna coklat sampai coklat kemerahan.

## 2. Hawar daun bakteri

Penyakit hawar daun bakteri (HDB), atau yang dikenal sebagai *bacterial leaf blight*, adalah penyakit serius dalam pertanian padi yang disebabkan oleh bakteri *Xanthomonas oryzae* pv. *oryzae*. Penyakit ini menjadi salah satu kendala utama dalam produksi padi di Indonesia [26].



**Gambar 2. 4 Penyakit hawar daun bakteri**

Gambar 2.4 menunjukkan penyakit hawar daun bakteri pada tanaman padi pada fase awal pertumbuhannya, yang mengakibatkan tanaman menjadi layu dan mati. Gejala penyakit hawar daun bakteri muncul ketika tepi daun berubah warna menjadi keabuan dan daun mengering. Bagian yang kering tersebut akan semakin menyebar ke arah tulang daun hingga seluruh daun akan mengering.

## 3. Tungro

*Rice tungro virus* atau tungro adalah penyakit yang disebabkan oleh interaksi kompleks antara dua jenis virus, yaitu *Rice Tungro Bacilliform Virus* (RTBV) dan *Rice Tungro Spherical Virus* (RTSV). Penyakit ini adalah salah satu penyakit penting pada tanaman padi di Indonesia [27]. Keduanya hanya dapat menyebar melalui wereng hijau *Nephotettix virescens* (*Distant*) [28].





**Gambar 2. 5 Penyakit tungro**

Pada gambar 2.5 merupakan contoh penyakit tungro yang mempunyai gejala diantaranya daun muda kuning hingga oranye di ujungnya, pertumbuhan tanaman yang kerdil, jumlah anakan yang berkurang, dan perkembangan akar yang lambat [29].

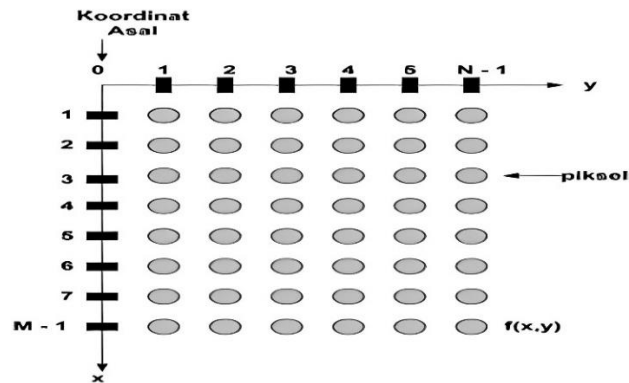
**Tabel 2. 2 Jenis penyakit padi dan ciri-cirinya**

Jenis Penyakit	Ciri Fisik
Padi Sehat	Daun berbentuk sempit dan memanjang, serta urat daun sejajar.
Blas Kolar	Terdapat bintik-bintik berbentuk belah ketupat
Hawar daun Bakteri	Dari warna hijau dengan garis coklat hingga abu-abu
Tungro	Kuning layu hingga oranye dengan bentuk tulang daun padi yang tidak beraturan

### 2.2.3 CITRA DIGITAL

Citra digital adalah gambar dua dimensi yang dimunculkan pada layar komputer dalam bentuk nilai digital atau piksel, sehingga komputer dapat memproses gambar atau citra secara langsung [30]. Citra merupakan representasi objek tiga dimensi yang dipresentasikan dalam format dua dimensi dengan kombinasi garis, titik, bentuk, dan warna. Dalam teknologi digital, citra atau gambar terbentuk dari gabungan dari piksel yang mengandung warna sedemikian

rupa hingga menciptakan imitasi objek dan menampilkan tekstur tertentu. Komputer menganalisis dan mengenali informasi dasar tersebut untuk mengidentifikasi peristiwa atau situasi tertentu [31].



**Gambar 2. 6 Koordinat pada citra digital [32]**

Dapat dilihat dari gambar 2.6, sebuah citra digital ditampilkan sebagai matriks dua dimensi dengan M kolom dan N baris. Titik dimana kolom dan baris bertemu disebut dengan piksel (*picture element*) yang merupakan elemen terkecil dari citra tersebut. Piksel tersebut memiliki dua parameter yaitu koorniat (x, y) dan intensitas warna. Dalam representasi ini, nilai  $f(x,y)$  merupakan besar intensitas warna pada piksel yang terletak pada koordinat (x,y). Representasi tersebut berupa koordinat (m-1 dan n-1). Dengan demikian, citra memiliki ukuran m x n piksel [32].

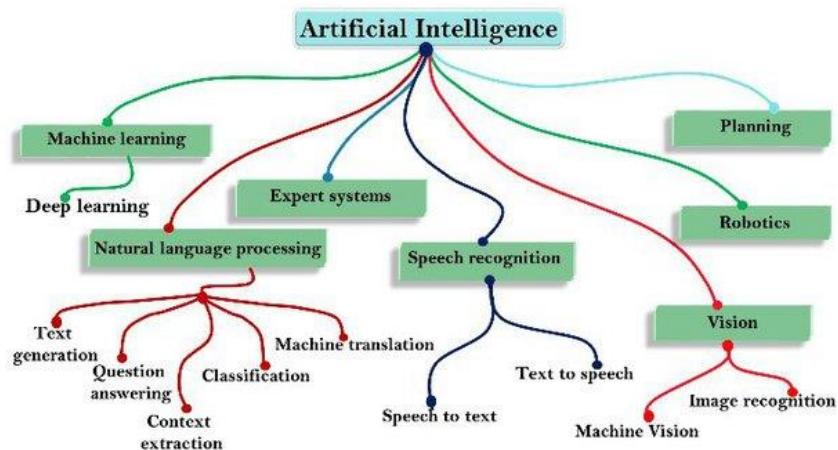
#### 2.2.4 PENGOLAHAN CITRA DIGITAL

Pengolahan citra (*image Processing*) adalah proses memproses piksel-piksel dalam citra digital untuk mencapai tujuan tertentu. Awalnya, pengolahan citra bertujuan untuk meningkatkan kualitas citra. Namun, seiring berkembangnya teknologi komputer yang meningkatkan kapasitas dan kemampuan komputasi, manusia dapat mengambil informasi lebih lanjut dari citra tersebut. Pengolahan citra digital adalah bidang ilmu yang mempelajari tentang pembentukan, pengolahan, dan analisis citra untuk memunculkan informasi yang dapat dimengerti manusia. Histogram citra merupakan grafik yang menunjukkan distribusi nilai-nilai warna atau intensitas piksel-piksel dalam citra. Pengolahan citra digital merupakan bidang studi yang memfokuskan pada metode untuk mengolah citra, yang dapat berupa foto atau video yang direkam. Arti digital merupakan pemrosesan citra atau gambar menggunakan komputer [33].

### 2.2.5 ARTIFICIAL INTELLIGENCE

*Artificial Intelligence* (AI), atau kecerdasan buatan adalah teknologi dalam bidang komputer yang meniru kemampuan intelektual manusia dalam mesin (komputer) agar menangani berbagai masalah dan berperilaku layaknya manusia, bahkan dengan potensi untuk melampaui kemampuan manusia. *Artificial Intelligence* (AI) adalah bidang penelitian yang menekankan pada penangkapan, pemodelan, dan penyimpanan kecerdasan manusia dalam sistem teknologi, memungkinkan sistem tersebut untuk mendukung proses pengambilan keputusan yang sering dilakukan oleh manusia [34].

Banyak teknologi AI yang diterapkan ke dalam banyak bidang. Referensi populer di dunia akademis membagi AI menjadi lima penerapan yang terkait dengan tujuan penggunaan kecerdasan buatan. Tentang bagaimana AI digunakan untuk pemecahan masalah, pengembangan informasi, penalaran dan perencanaan, resolusi dan penalaran ketidakpastian informasi, pembelajaran mesin, kemampuan komunikasi, memahami serta bertindak. Namun jika bisa diringkas, ilmu kecerdasan buatan saat ini dibagi menjadi tujuh sub-bidang. Tetapi, pembagian ini tidak mutlak dan bervariasi. Beberapa sub-bidang mungkin juga saling tumpang tindih di beberapa tempat karena luasnya penerapan AI ini yang dapat dilihat pada gambar 2.7. [35].

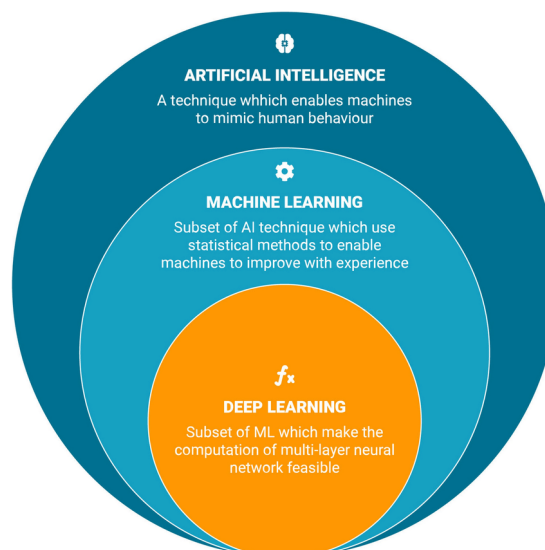


Gambar 2. 7 Sub-bidang keilmuan AI [36]

### 2.2.6 DEEP LEARNING

*Deep learning* merupakan subbidang dalam *machine learning* yang memanfaatkan Jaringan Syaraf Tiruan (JST) agar melatih sebuah mesin atau komputer dalam meniru kemampuan layaknya mirip manusia, seperti belajar dari contoh dan menerapkannya dalam pengambilan keputusan. Dalam *deep learning*, komputer mempelajari untuk mengklasifikasikan gambar, teks, atau suara secara langsung [37]. Memprediksi peluang atau peristiwa, mengenali objek, dan mendiagnosis penyakit adalah beberapa contoh pekerjaan dimana *deep learning* digunakan [38].

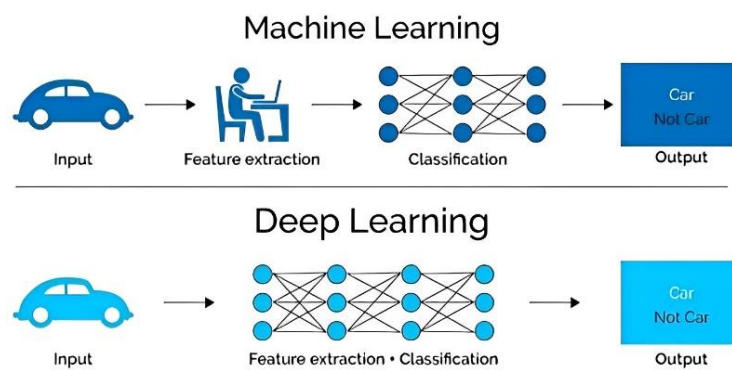
Dalam konteks ini, kata "*deep*" mengacu pada jumlah lapisan yang terdapat pada jaringan saraf, semakin banyak jumlah lapisan pada jaringan saraf, maka akan semakin dalam (*deep*) pula jaringan tersebut. *Deep neural network* terdiri dari banyak lapisan *neuron* yang memungkinkan mereka mempelajari representasi hierarki data karena setiap lapisan akan mengekstraksi fitur yang semakin kompleks dari data input, dimulai dengan fitur sederhana pada lapisan pertama, dan berkembang menjadi konsep yang lebih kompleks pada lapisan berikutnya. *Deep learning* telah berhasil mencapai prestasi luar biasa dalam berbagai aplikasi seperti pengenalan citra, identifikasi audio, dan pemrosesan bahasa alami dengan memanfaatkan fitur-fitur ini [39].



**Gambar 2. 8 Deep learning [40]**

Gambar 2.8 *deep learning* termasuk bagian dari AI. Ada beberapa pendekatan dalam menggunakan *deep learning* untuk menyelesaikan tugas,

masing-masing dengan kelebihan dan kekurangannya. *Deep learning* dipilih karena kemampuannya untuk belajar dengan mudah disesuaikan dan cepat dipelajari. Fitur yang dirancang secara manual seringkali terlalu spesifik, tidak lengkap, dan susah untuk diimplementasikan. Salah satu keunggulan lain dari *deep learning* adalah menyediakan susunan kerja yang sangat fleksibel, global, dan dapat dipelajari untuk menampilkan informasi visual, dan bahasa. Di samping itu, untuk mencapai generalisasi yang tidak terbatas pada lokasi tertentu, *deep learning* juga memerlukan representasi yang terdistribusi. Meskipun *deep learning* termasuk subbidang dari *machine learning*, cara kerja keduanya cukup berbeda [40]. Perbedaan dapat dilihat pada gambar berikut.



**Gambar 2. 9 Perbedaan *machine learning* dan *deep learning***

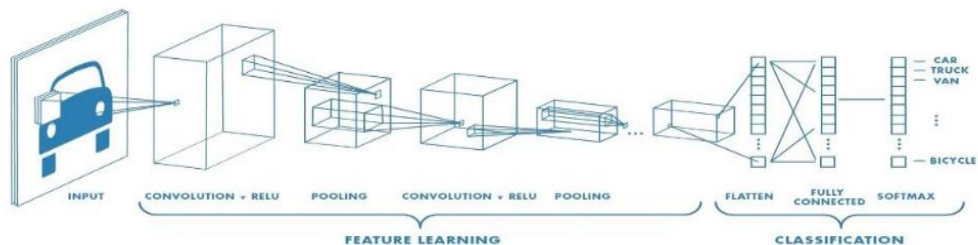
Pada gambar 2.9 dapat dilihat perbedaan yang cukup jelas antara *machine learning* dan *deep learning*. Dalam ilustrasi tersebut, *machine learning* membutuhkan *feature extraction* untuk mengidentifikasi karakteristik khusus dari data. *Feature extraction* nantinya menggabungkan variabel dari data ke dalam fitur, sehingga mengurangi jumlah data yang perlu diproses. Klasifikasi adalah langkah untuk menemukan kelas yang terkait dengan masukan yang diberikan. Berbeda dengan *machine learning*, pada *deep learning*, ekstraksi fitur dan klasifikasi dilakukan bersama dan tidak dipisahkan. Permasalahan yang sering muncul pada *deep learning* adalah bahwa semakin besar *neural network*nya, maka akan semakin besar pula beban komputasi yang dibutuhkan. Namun, dengan adanya *Graphic Processing Unit* (GPU) dapat mempercepat proses *training* data tersebut. Dengan terlihatnya perbedaan antara *machine learning* dan *deep learning*, penggunaannya bergantung dengan kebutuhan. Penerapan *deep learning* diperlukan saat data yang nantinya diproses memiliki ukuran cukup besar, membutuhkan infrastruktur tingkat

tinggi untuk *training* dalam waktu yang wajar, dan diterapkan pada kasus-kasus dengan fitur introspeksi yang kompleks, seperti klasifikasi gambar, pemrosesan bahasa alami, dan pengenalan ucapan.

### 2.2.7 CONVOLUTIONAL NEURAL NETWORK (CNN)

*Convolutional Neural Network* (CNN) merupakan algoritma dalam *deep learning* yang telah dikembangkan dari *Multilayer Perceptron* (MLP) dan dioptimalkan untuk memproses data berformat dua dimensi seperti gambar atau audio [41]. CNN dapat menemukan pola level atau tingkatan dalam data dan menyusun piksel-piksel kecil menjadi bentuk yang lebih kompleks dengan cara sederhana. Oleh karena itu, CNN menunjukkan performa yang baik dalam mengatasi keterhubungan dan kompleksitas piksel pada gambar.

Model *deep learning*, khususnya *Convolutional Neural Network* (CNN), terbukti efektif dalam mengidentifikasi dan pengolahan citra. CNN adalah jenis jaringan saraf tiruan yang umumnya dimanfaatkan sebagai penyelesaian masalah pengenalan pola seperti pengenalan suara dan *computer vision*. CNN merupakan jenis jaringan saraf tiruan yang sedikit berbeda karena setiap *neuronnya* digambarkan dalam struktur dua dimensi, berbeda dengan *Multilayer Perceptrons* (MLP) yang menggunakan struktur *neuron* satu dimensi. Perbedaan arsitektur ini membuat CNN lebih efisien untuk mengenali pola pada data citra. Pada penelitian ini, CNN digunakan untuk mengidentifikasi penyakit daun pada tanaman padi. Hal ini dipilih karena metode ini memiliki hasil yang baik, mudah digunakan dalam proses pelatihan, dan memiliki parameter yang lebih sedikit dibandingkan dengan metode lainnya. Selain itu, CNN berusaha meniru sistem pengenalan citra di korteks visual manusia, sehingga dapat memproses data citra dengan baik [42].

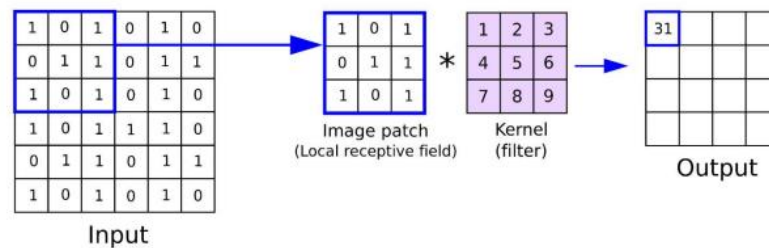


Gambar 2. 10 Arsitektur CNN [43]

Terlihat pada gambar 2. 10 terdapat arsitektur CNN tersusun dari beberapa lapisan, diantaranya *convolutional layer*, *pooling layer*, *fully-connected layer*, dan yang terakhir ada *output* lapisan yang merupakan hasil dari proses *Convolutional Neural Network*.

a. *Convolutional Layer*

Lapisan konvolusi adalah lapisan awal pada CNN yang merupakan inti dari proses perhitungan. Lapisan ini terdiri dari sejumlah filter atau kernel, dimana setiap kernel melakukan operasi konvolusi pada input dan menghasilkan satu channel dalam *output* yang disebut sebagai *feature map*. Salah satu fungsi aktivasi yang diterapkan di lapisan konvolusi adalah *ReLU (Rectified Linear Unit)*, fitur yang biasa digunakan dalam CNN [44].



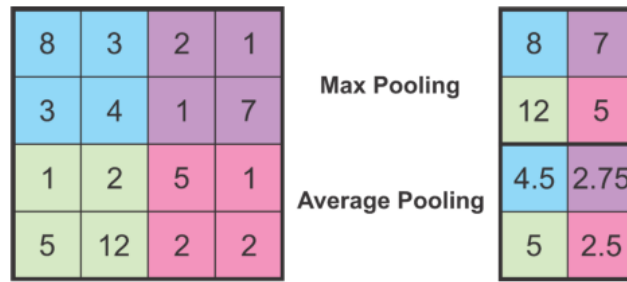
**Gambar 2. 11 Operasi konvolusi [45]**

Pada Gambar 2. 11 merupakan representasi operasi konvolusi dimana, filter dengan ukuran 3x3 bergeser sambil melakukan operasi dot pada data input dengan ukuran 6x6 untuk menghasilkan *feature map*. Konvolusi ini dapat dioperasikan dengan melakukan perkalian matriks pada baris dan kolom yang sama pada data input dan filter, lalu menjumlahkan hasilnya untuk menjadi satu nilai representatif pada peta fitur. Setelah operasi dot dilakukan, maka operasi berlanjut dengan bergeser sebesar 1 *stride* untuk melakukan operasi dot hingga seluruh matriks 6x6 sudah selesai.

b. *Pooling Layer*

*Pooling Layer* digunakan untuk mengurangi dimensi spasial dari peta fitur yang dibuat dari lapisan konvolusi. Tujuannya adalah untuk mengurangi jumlah parameter yang diperlukan, mengurangi beban komputasi, serta membantu mencegah *overfitting*. Umumnya, lapisan *pooling* diletakkan di antara lapisan konvolusi untuk mengurangi resolusi *feature map*.



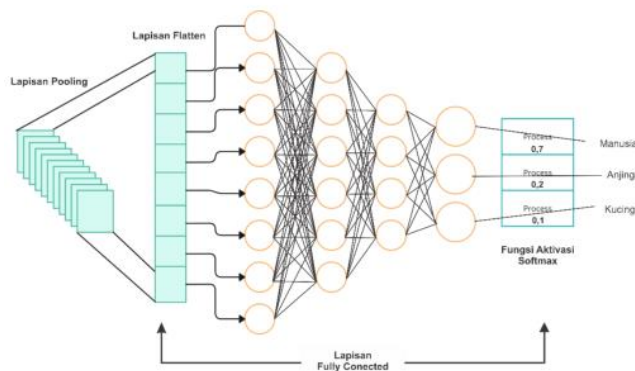


**Gambar 2. 12 Pooling layer [44]**

Pada gambar 2.12 merupakan jenis *pooling*. Ada dua jenis *pooling layer* yang umum digunakan, yakni *max pooling* dan *average pooling* [44]. Jenis *max pooling* yang paling umum adalah *max pooling*, karena pada *max pooling* nilai maksimumnya diambil dari sekumpulan nilai di dalam filter atau *kernel* berukuran 2x2. Sedangkan pada *average pooling*, nilai rata-rata diambil dari sekumpulan nilai didalam filter atau *kernel* tersebut.

c. Lapisan *Fully-Connected*

Lapisan ini sering disebut sebagai lapisan *dense* yang merupakan bagian akhir dari struktur *Convolutional Neural Network*. Lapisan ini beroperasi setelah menerima vektor hasil dari lapisan *pooling* yang telah diratakan dimensinya menggunakan teknik *flattening* atau *global average pooling*. Hasil ini kemudian dimasukkan ke dalam satu atau beberapa lapisan yang terhubung sepenuhnya (*fully connected*). Lapisan ini berfungsi untuk menyatukan informasi dari fitur-fitur yang telah diekstraksi sebelumnya dan melakukan tugas klasifikasi yang terlihat pada gambar 2.13.



**Gambar 2. 13 Lapisan *fully-connected* [46]**

Sebelum dimasukkan ke dalam lapisan sepenuhnya terhubung, setiap neuron di lapisan konvolusi perlu diubah menjadi bentuk satu dimensi terlebih dahulu. Namun, perlu diingat bahwa proses ini mengakibatkan hilangnya informasi



spasial yang tidak dapat dipulihkan. Oleh karena itu, *fully connected* hanya dapat diterapkan di bagian akhir jaringan [46].

### 2.2.8 FUNGSI AKTIVASI *RECTIFIED LINEAR UNIT (RELU)*

*Rectified Linear Unit* atau *ReLU* sering digunakan sebagai fungsi aktivasi dalam jaringan saraf tiruan, khususnya dalam arsitektur *Convolutional Neural Network* (CNN). Dalam CNN, fungsi aktivasi biasanya terletak setelah perhitungan konvolusi. Fungsi aktivasi ini digunakan untuk mengubah seluruh nilai piksel pada input gambar yang bernilai negatif yaitu kurang dari nol, menjadi nol dan memberikan nilai positif tidak berubah guna meminimalisir *error* dan menghilangkan nilai negatif dari lapisan sebelumnya [47].

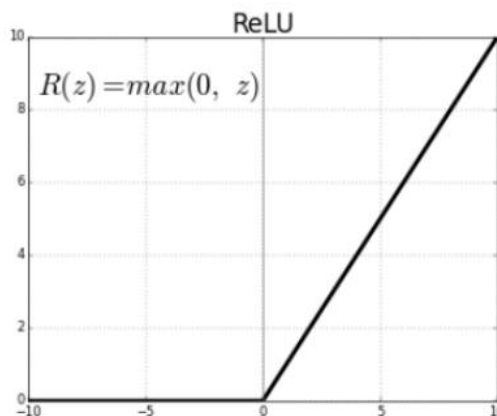
Fungsi aktivasi menunjukkan antara tingkat aktivasi internal (*summation function*) dengan pola linear atau non-linear. Kelebihan dari fungsi aktivasi ini adalah kemampuannya untuk meningkatkan kecepatan konfigurasi dengan *Stochastic Gradient descent* (SGD), melebihi fungsi sigmoid dan tanh [44]. Berikut adalah persamaan aktivasi *ReLU*.

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

Dimana,  $f(x)$  = fungsi *ReLU*

$x$  = nilai batasan *ReLU*

Di sini  $x$  adalah input ke fungsi *ReLU* dan  $f(x)$  adalah *output* yang dihasilkan setelah mengaplikasikan fungsi *ReLU* pada input  $x$ . Jadi, jika  $x$  positif atau 0, maka *output* akan tetap  $x$ , tetapi jika  $x$  bernilai negatif, maka *output* akan menjadi 0.



**Gambar 2. 14** Grafik aktivasi *ReLU* [42]

Pada Gambar 2.14 menunjukkan grafik fungsi aktivasi *ReLU*. Dari grafik tersebut, terlihat bahwa ketika nilai piksel yang bernilai negatif atau kurang dari 0, nilai tersebut diubah menjadi 0. Sedangkan ketika nilai masukan yang positif atau lebih besar dari 0, *outputnya* sama dengan nilai masukan tersebut. Beberapa fungsi aktivasi umum yang digunakan dalam *deep learning* diantaranya *ReLU*, *Binary Step*, *Sigmoid*, *Tanh*, dan lain sebagainya.

### 2.2.9 TRAINING, VALIDATION, DAN TESTING SET

Pada pengembangan model, ada dua istilah utama yaitu *training* dan *testing*. *Training* mempresentasikan proses pembangunan model, sedangkan *testing* digunakan untuk mengevaluasi kinerja model yang sudah dibuat. Dataset, yang merupakan kumpulan data atau sampel dalam statistik, diperlukan baik selama proses *training* maupun *testing* model. Kategori dataset terbagi dari data berikut merupakan yang tidak saling beririsan satu sama lain.

#### a. *Training set*

*Training set* adalah kumpulan data yang digunakan untuk melatih atau membentuk model.

#### b. *Validation set*

*Validation set* adalah kumpulan data yang digunakan untuk mengoptimalkan model selama pelatihan. Model dilatih menggunakan *training set* dan kinerjanya akan dievaluasi menggunakan *validation set*.

#### c. *Testing set*

Pada *testing set* ini dilakukan untuk menguji atau *testing* model setelah proses *training* selesai [48].

### 2.2.10 ACCURACY DAN LOSS

*Accuracy* dan *loss* menjadi elemen penting dalam pengklasifikasian citra. akurasi merupakan metode untuk mengukur kinerja model klasifikasi yang hasilnya dapat dinyatakan dalam presentase.

$$\text{Akurasi} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \times 100\% \quad (2.2)$$

Rumus 2.2 menjelaskan cara menghitung nilai *accuracy*. Pada perhitungan tersebut akan dilakukan pembagian antara jumlah *True Positive* (TP) dengan *True Negative* (TN), kemudian hasilnya dibagi dengan penjumlahan antara *True Positive* (TP), *True Negative* (TN), *False Positive* (FP), dan *False Negative* (FN) yang diperoleh dari pengujian citra. Hasil dari kelas prediksinya juga dibandingkan dengan kelas aktual atau kelas sebenarnya pada *confussion matrix*. Sementara itu, *loss* atau kerugian mengindikasikan seberapa besar *loss* pada klasifikasi citra, dimana semakin rendah *loss* maka semakin baik model yang dibuat [49].

### 2.2.11 CONFUSION MATRIX

*Confusion matrix* merupakan metode yang digunakan untuk mengukur dan mengevaluasi kinerja dari model dalam klasifikasi. *Confusion matrix* juga memberikan informasi mengenai kesesuaian hasil klasifikasi model dengan klasifikasi yang sebenarnya. *Confusion matrix* memiliki empat bagian utama yaitu *True Positive* (TP), *True Negative* (TN), *False Positive* (FP), dan *False Negative* (FN) [45]. Metode ini sering digunakan untuk menghitung akurasi, presisi, *recall*, dan *F1-Score*. Pada *confusion matrix* yang terpenting dapat memberikan informasi mengenai jumlah kelas prediksi yang diberikan, apakah prediksinya benar atau salah. *Confusion matrix* berbentuk matriks persegi dengan ordo  $n \times n$ , dimana  $n$  menunjukkan jumlah kelas target. Umumnya setiap kolom pada *confusion matrix* mewakili kelas aktual atau kelas sebenarnya sedangkan setiap baris pada *confusion matrix* mewakili kelas prediksi. Tabel 2.3 merupakan *confusion matrix* yang digunakan.

**Tabel 2. 3 Confusion matrix**

		Kelas Prediksi	
		Positif	Negatif
Kelas Aktual	Positif	TP	FP
	Negatif	FN	TN

Tabel 2.3 menampilkan hasil dari *confusion matrix* pada pengklasifikasian citra. Informasi yang terdapat pada tabel dapat diuraikan sebagai berikut:

a. TP (*True Positive*)

*True positive* menunjukkan pada kelas aktualnya positif dan pada prediksi juga positif (*true*).

b. FP (*False Positive*)

Pada FP ini akan menunjukkan jumlah data yang kelas aktualnya positif, namun hasil pada kelas prediksi salah (*false*).

c. TN (*True Negative*)

Pada kondisi ini menunjukkan jumlah data pada kelas aktualnya positif dengan hasil prediksinya salah (*false*).

d. FN (*False Negative*)

Pada kondisi ini menunjukkan jumlah data pada kelas aktual negatif dengan hasil prediksinya salah (*false*).

## 2.2.12 PARAMETER YANG DIUJI

Beberapa parameter yang digunakan untuk mengukur performa model CNN diantaranya akurasi, presisi, *recall*, dan *F1-Score* [50].

### 1. Akurasi

Nilai *accuracy* atau akurasi dapat dikatakan sebagai perbandingan antara data yang diprediksi secara benar dengan total prediksi yang dilakukan. Akurasi menggambarkan seberapa akurat model yang telah dibuat untuk mengklasifikasikan data dengan benar [51].

$$\text{Akurasi} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \times 100\% \quad (2.3)$$

### 2. Presisi

Presisi menggambarkan rasio seberapa banyak nilai yang benar-benar positif dari semua nilai yang diprediksi positif. Nilai presisi dapat digunakan untuk menentukan apakah model yang dibuat dapat diandalkan atau tidak.

$$\text{Presisi} = \frac{\text{TP}}{\text{TP} + \text{FP}} \times 100\% \quad (2.4)$$

### 3. *Recall*

*Recall* atau disebut juga dengan *sensitivity* (*true positive rate*) yang menggambarkan berapa banyak data kelas aktual positif yang dideteksi benar.

Nilai *recall* menggambarkan seberapa baik model dalam menentukan kembali sebuah informasi yang relevan.

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \times 100\% \quad (2.5)$$

#### 4. *F1-Score*

Nilai *F1-Score* yang umumnya disebut *f-measure* adalah *harmonic mean* atau perbandingan rata-rata dari presisi dan *recall*, sehingga akan memberikan gambaran nilai presisi dan *recall* secara bersamaan.

$$\text{F1 - Score} = 2 \times \frac{\text{Presisi} \times \text{Recall}}{\text{Presisi} + \text{Recall}} \quad (2.6)$$

### 2.2.13 *PHYTON*

*Python* merupakan bahasa pemrograman tingkat tinggi yang dapat diinterpretasikan, interaktif, dan beroperasi pada hampir semua platform seperti *Mac*, *Linux* dan *Windows*. *Python* adalah bahasa pemrograman yang mudah dipelajari karena sintaksnya yang jelas, dapat digabungkan dengan penggunaan modul-modul siap pakai, dan memiliki struktur data tingkat tinggi yang kuat. *Python* memiliki dukungan yang banyak dari *programmer*, sehingga memiliki segudang fitur yang cocok untuk digunakan. *Python* juga memiliki tata penulisan *script* yang mudah untuk dipahami, dapat mengolah data dan memori secara otomatis. *Python* juga dapat diaplikasikan pada berbagai sistem operasi diantaranya *Linux*, *Windows*, *MacOS*, *Android*, *Amiga*, *Palm*, *Symbian OS*, dan lain-lain.