

BAB II

KAJIAN PUSTAKA

A. Hasil Penelitian Terdahulu

Banyak penelitian yang telah dilakukan dalam mengembangkan model klasifikasi penyakit pertanian dengan berbagai macam pendekatan. Salah satu metode yang sering digunakan adalah CNN, yang memiliki potensi untuk ditingkatkan lagi melalui penerapan metode *ensemble*. Beberapa studi terkait penggunaan metode CNN dapat dilihat pada Tabel 2.1.

Tabel 2.1 Penelitian terdahulu terkait CNN

No	Peneliti	Objek Penelitian	Metode	Hasil
1	Yang et al. (2023)	Penyakit daun padi dengan 1.122 data gambar penyakit	<i>Ensemble stacking</i> dengan <i>base learner</i> (<i>AlexNet</i> , <i>G</i> , <i>RE-GoogLeNet</i> , <i>ResNet50</i> , dan <i>MobileNetV3</i>) dengan SVM sebagai <i>meta-learner</i>	Akurasi model mencapai 99,69%
2	Dhanasekaran (2023)	Daun kentang, jagung, apel, dan anggur	<i>Ensemble stacking</i> dengan <i>base learner</i> (<i>Xception</i> , <i>InceptionV3</i> , dan <i>DenseNet201</i>) dengan CNN sebagai <i>meta-learner</i>	Akurasi model mencapai 99,93%
3	Aishwarya & Reddy (2023)	Penyakit pada daun kacang tanah	<i>Ensemble CNN</i> dengan menggabungkan (<i>DenseNet-169</i> , <i>InceptionV3</i> , dan <i>Xception</i>)	Akurasi model mencapai 98,46%
4	Alidrus et al. (2021)	Penyakit daun padi dengan 950 dataset gambar	Model <i>Convolutional Neural Network</i> (<i>CNN</i>)	Akurasi rata-rata sebesar 92% pada data latih dan 77% pada validasi

No	Peneliti	Objek Penelitian	Metode	Hasil
5	Jinan et al. (2022)	Penyakit pada daun padi dengan 120 data yang terdiri dari tiga kategori	Model <i>Convolutional Neural Network</i> (CNN)	Dengan akurasi 91,7%. Dan tingkat kesalahan 8,8%
6	Yuliany & Rachman (2022)	Hama padi dengan data berjumlah 1.065 gambar yang terdiri dari lima kategori	Model <i>Convolutional Neural Network</i> (CNN)	Akurasi validasi 83,02% dan akurasi pengujian tertinggi 77,33%
7	Widianita (2023)	Hama padi dengan 250 dataset gambar yang terdiri dari lima kategori	Model <i>Convolutional Neural Network</i> (CNN)	Akurasi awal 61% yang berhasil ditingkatkan menjadi 90%
8	Setyadi et al. (2024)	Penyakit pada daun cabai merah dengan data berjumlah 2.128 gambar yang terdiri dari lima kategori	Model <i>Convolutional Neural Network</i> (CNN) dengan arsitektur <i>ResNet101</i>	Akurasi validasi 89,72% - 97,18%, serta arsitektur <i>ResNet101</i> mencapai akurasi 98,12%
9	Rozi & Tamam (2024)	Penyakit daun padi dengan data berjumlah 200 gambar	Model <i>Convolutional Neural Network</i> (CNN) dan <i>Random Forest</i>	Hasil akhir akurasi 100% dan validasi 94,73%
10	Kusuma et al. (2025)	Penyakit daun padi dengan 53.300 dataset gambar yang terdiri dari 4 kategori	Model <i>Convolutional Neural Network</i> (CNN) dengan <i>MobileNetV3 Large</i>	Dengan akurasi 75% data uji dan 76% untuk data latih
11	Rekomendasi kebaruan	Penyakit pada daun padi	<i>Ensemble stacking</i> dengan model CNN sebagai <i>base learner</i> dan <i>Logistic Regression</i> sebagai <i>meta-learner</i>	

Pada penelitian yang dilakukan oleh Yang et al. (2023) menggunakan metode *ensemble learning* berbasis *stacking* untuk identifikasi penyakit daun padi. Model *stacking* menggabungkan empat model CNN sebagai *base learner* (*AlexNet_G*, *RE-GoogLeNet*, *ResNet50*, dan *MobileNetV3*) dengan SVM sebagai *meta-learner*. Dataset yang digunakan terdiri dari 1.122 gambar penyakit daun padi dengan delapan jenis penyakit berbeda, data berasal dari berbagai sumber dengan gabungan data dari lapangan secara langsung. Data diproses menggunakan augmentasi (*flipping*, rotasi, *cropping*, transformasi warna, dan *blurring*) serta juga dilakukan *resize* ke 224×224 piksel pada gambar, tujuannya untuk mengurangi *overfitting* pada data. Hasil pada penelitian ini menunjukkan model mencapai akurasi 99,69%, lebih unggul dibanding model individu dan pendekatan sebelumnya dalam identifikasi penyakit daun padi yang hanya menggunakan satu model CNN saja.

Sistem klasifikasi tanaman kentang, jagung, apel, dan anggur telah dikembangkan oleh Dhanasekaran (2023) dengan menggunakan metode *ensemble stacking*. Dalam penelitian yang telah dibuat, model *Xception*, *InceptionV3*, dan *DenseNet201* digunakan sebagai *base learner*, sedangkan model CNN digunakan sebagai *meta-learner*. Model yang telah dibuat diuji menggunakan dataset PlantVillage yang terdiri dari data gambar daun kentang, jagung, apel, dan anggur. Hasil eksperimen menunjukkan bahwa model mencapai tingkat akurasi 99.93%, mengungguli metode *ensemble* lainnya.

Pada penelitian yang dilakukan oleh Aishwarya & Reddy (2023) yang menggunakan *ensemble CNN* untuk klasifikasi penyakit daun tanaman kacang tanah. Penelitian menggunakan model *DenseNet-169*, *InceptionV3*, dan *Xception* sebagai *base learner*, yang nantinya akan digabungkan menggunakan model CNN sebagai *meta-learner*-nya. Data yang digunakan berasal dari data gambar daun kacang tanah terdampak hama yang diambil langsung dari lingkungan nyata dan telah diproses. Model menggunakan dua persamaan non-linear untuk menggabungkan hasil dari *base learner* guna meningkatkan akurasi prediksi. Hasil menunjukkan bahwa metode ini mencapai akurasi

98.46%, lebih unggul dibandingkan metode individual dan *ensemble* tradisional seperti *voting*.

Alidrus et al. (2021) mengusulkan penggunaan *Convolutional Neural Network* (CNN) untuk mendeteksi dan mengklasifikasikan penyakit pada daun tanaman padi dengan tiga kelas utama: blast, brown spot, dan hispa. Dataset yang digunakan terdiri dari 900 citra dari RiceLeafs dan 50 citra hasil observasi langsung, yang diproses melalui *resizing*, *filtering*, dan pelatihan model menggunakan arsitektur CNN dengan 4 *convolutional* dan 4 max pooling layers. Model dilatih selama 50 epoch dengan hyperparameter seperti *ReLU*, *RMSprop*, dan *dropout* 0.4. Hasil pengujian menunjukkan akurasi rata-rata sebesar 92% pada data latih dan 77% pada data validasi, mengindikasikan bahwa CNN efektif dalam tugas klasifikasi citra daun padi, meskipun performa validasi masih dapat ditingkatkan melalui pengoptimalan arsitektur dan augmentasi data.

Pada penelitian yang dilakukan oleh Jinan et al. (2022) dalam pembuatan sistem klasifikasi hama pada tanaman padi, menggunakan model *Convolutional Neural Network* (CNN) dalam mengklasifikasikan penyakit pada daun padi berdasarkan citra daun. Dataset yang digunakan terdiri dari 120 gambar daun padi yang terdiri dari tiga kategori penyakit: Bacterial Leaf Blight, Brown Spot, dan Leaf Spot. Model CNN diuji dengan ukuran 100×100 piksel, kernel 3×3 , *learning rate* 0,01, *Adam optimizer*, epoch 150, dan *batch size* 30. Hasil eksperimen menunjukkan bahwa model mencapai akurasi 91,7%, membuktikan efektivitas CNN dalam mengklasifikasikan penyakit padi dengan tingkat kesalahan hanya 8,8%.

Yuliany & Rachman (2022) mengimplementasikan *Convolutional Neural Network* (CNN) sebagai metode dalam pembuatan sistem klasifikasi hama tanaman padi. Data yang digunakan berjumlah 1.065 gambar tanaman padi yang terserang hama. Ada lima jenis hama yaitu belalang, hispa padi, penggerek batang, ulat grayak, dan wereng hijau. Data yang sudah diambil kemudian dibagi dalam skenario 70%:30%, 80%:20%, dan 90%:10%, dengan

augmentasi citra untuk mengatasi *overfitting*. Hasil terbaik diperoleh dengan pembagian data 90%:10%, epoch 200, dan *learning rate* 0,001, mencapai akurasi validasi 83,02% dan akurasi pengujian tertinggi 77,33%.

Pada penelitian yang dilakukan Widianita (2023) tentang penggunaan *Convolutional Neural Network* (CNN) untuk klasifikasi hama tanaman padi berdasarkan citra daun dengan *framework* TensorFlow berbasis Android. Dataset terdiri dari 250 gambar, mencakup lima jenis hama: hawar bakteri, walang sangit, wereng padi, penggerek batang padi, dan keong mas. Model diuji dengan berbagai konfigurasi dan mencapai akurasi awal 61% yang berhasil ditingkatkan hingga 99%, dengan rata-rata akurasi 90%. Hasil menunjukkan model CNN dapat mengklasifikasikan hama dengan lebih akurat

Setyadi et al. (2024) telah melakukan penelitian dengan tujuan mengklasifikasikan penyakit pada daun cabai merah menggunakan metode *convolutional neural network* (CNN) dengan pendekatan *transfer learning*, terutama menggunakan arsitektur *ResNet101*. Dalam rangka meningkatkan deteksi dini penyakit pada tanaman cabai yang penting secara ekonomi di Indonesia, peneliti menggunakan dataset citra daun cabai yang telah diaugmentasi, terdiri dari lima kelas: embun tepung, daun sehat, kompleks murda (tungau dan thrips), bercak daun (*Cercospora*), dan kurang nutrisi. Total dataset yang digunakan sebanyak 2.128 gambar (1.702 untuk training dan 426 untuk validasi). Hasil penelitian menunjukkan bahwa augmentasi data berhasil meningkatkan akurasi validasi dari 89,72% menjadi 97,18%. Dibandingkan berbagai arsitektur CNN seperti *AlexNet*, *GoogleNet*, *VGGNet16*, dan *ResNet50*, arsitektur *ResNet101* mencapai akurasi validasi tertinggi yaitu 98,12%, mengungguli *ResNet152* meskipun *ResNet152* memiliki akurasi pelatihan tertinggi. Penelitian ini menyimpulkan bahwa *transfer learning* dengan arsitektur *ResNet101* sangat efektif dalam mengklasifikasikan penyakit daun cabai, terutama dalam kondisi keterbatasan data, dan memberikan kontribusi nyata terhadap pengembangan sistem pertanian cerdas yang mampu mendeteksi penyakit tanaman secara cepat dan tepat.

Penelitian yang dilakukan oleh Rozi & Tamam (2024) dalam mengembangkan sistem klasifikasi penyakit pada daun padi dengan memanfaatkan algoritma *Convolutional Neural Network (CNN)* dan *Random Forest*. Dataset berisi 200 citra daun yang terbagi dalam dua kelas: sehat dan berpenyakit. Proses pelatihan menggunakan CNN menghasilkan akurasi pelatihan hingga 100% dan validasi 94,73%, meskipun terdapat indikasi *overfitting*. Sementara itu, *Random Forest* juga menunjukkan akurasi sangat tinggi mendekati 99,97%. Evaluasi menggunakan *Confusion Matrix* dan metrik seperti *precision*, *recall*, dan *F1-score* menunjukkan performa yang sangat baik pada kedua model. CNN unggul dalam ekstraksi fitur visual, sedangkan *Random Forest* memberikan kestabilan prediksi. Penelitian ini menunjukkan bahwa kombinasi CNN dan *Random Forest* efektif untuk klasifikasi penyakit daun padi, serta berpotensi membantu petani dalam mendeteksi penyakit secara cepat dan akurat. Namun, pengembangan lebih lanjut seperti penggunaan teknik augmentasi data dan arsitektur model yang lebih kompleks disarankan untuk meningkatkan kinerja model secara keseluruhan.

Dalam penelitian yang telah dilakukan oleh Kusuma et al. (2025) yang bertujuan untuk mengembangkan model deteksi dan klasifikasi penyakit pada tanaman padi menggunakan algoritma *Convolutional Neural Network (CNN)* dengan arsitektur *MobileNetV3 Large*, serta mengimplementasikannya ke dalam aplikasi mobile berbasis Android. Dataset yang digunakan berasal dari Kaggle dengan jumlah 53.300 citra daun padi yang terdiri dari empat kelas: *Brownspot*, *Leafblast*, *Hispa*, dan *Healthy*. Proses data *preparation* mencakup *resize*, augmentasi (*flip*, *rotate*, *zoom*), dan *rescaling pixel*. Model dilatih menggunakan 30 epoch, batch size 45, dan dioptimasi menggunakan *optimizer* Lion, menghasilkan akurasi 75% pada data uji dan 76% pada data latih, dengan loss masing-masing 59% dan 61%. Evaluasi model dilakukan menggunakan *confusion matrix* dan metrik evaluasi lainnya seperti *Precision*, *Recall*, dan *F1-Score*, yang menunjukkan performa sangat baik pada kelas *Brownspot* dan *Healthy*, namun kurang optimal pada kelas *Hispa* karena kekhasan ciri visualnya yang minim. Keunggulan utama penelitian ini terletak pada

implementasi model ke aplikasi Android, memberikan solusi praktis dan efisien bagi petani untuk mendeteksi penyakit padi secara cepat. Meskipun terdapat keterbatasan seperti sensitivitas model terhadap kondisi lingkungan baru, penelitian ini menunjukkan potensi besar dalam mendukung pertanian cerdas dan dapat dikembangkan lebih lanjut untuk memperluas cakupan penyakit dan jenis tanaman.

B. Landasan Teori

1. Tanaman Padi

Tanaman padi merupakan salah satu jenis tanaman pangan yang paling dibutuhkan di Indonesia, bersal dari Benua Asia dan Afrika yang merupakan tanaman musiman yang dapat tumbuh dengan baik pada daeran tropis dan subtropis (Sodiq & Mujoko, 2017). Namun, keberadaan penyakit pada tanaman padi menjadi ancaman yang serius karena dapat menjadi akibat menurunnya kualitas padi bahkan sampai gagal panen. adapun klasifikasi penyakit tanaman padi berdasarkan pada bentuk daunnya antara lain.

a. Blast

Penyakit blast pada tanaman padi disebabkan oleh jamur *Pyricularia oryzae* dan merupakan salah satu penyakit utama yang menyerang pada fase vegetatif maupun generatif. Serangannya dapat menghambat pertumbuhan serta mengurangi jumlah anakan produktif, yang berdampak pada penurunan hasil gabah. Dalam kasus yang parah, tanaman bahkan bisa mati sebelum memasuki fase berbunga. Gejala penyakit ini ditandai dengan munculnya bercak pada bagian daun, buku, malai, dan gabah. Bercak pada daun biasanya berbentuk oval atau elips dengan ujung meruncing dan tepian berwarna coklat hingga merah kecoklatan (Sodiq & Mujoko, 2017).



Gambar 2.1 Penyakit daun blast

b. Hawar Daun

Menurut Sodiq dan Mujoko (2017), penyakit hawar daun disebabkan oleh bakteri *Xanthomonas oryzae*. Jika serangan terjadi sejak awal pertumbuhan, penyakit ini dapat menyebabkan gagal panen. Sementara itu, apabila infeksi terjadi saat fase pembungaan, proses pengisian bulir padi menjadi tidak optimal karena daun bendera mengalami kekeringan. Gejala yang muncul antara lain tepi daun tampak basah dengan warna kuning, abu-abu, atau kadang putih, serta daun terlihat seperti tersiram air panas, menggulung, dan akhirnya layu.



Gambar 2.2 Penyakit daun hawar daun

c. Tungro

Penyakit tungro disebabkan oleh virus tungro padi. Dampaknya masa pertumbuhan padi tertunda, mulai menjadi lebih kecil, dan pada

bulir terdapat bintik-bintik coklat kehitaman. Gejala serangan akan terlihat pada saat umur tanaman padi berumur 2-3 minggu sesudah masa tanam. Ukuran daun dan pelepah memendek serta jumlah anakan akan berkurang. warna daun menjadi kuning hingga oranye. Daun yang masih muda tampak adanya bercak atau warna hijau pucat hingga bergaris putih sejajar dengan tulang daun yang berukuran berbeda. Dan daun tua akan tampak memiliki bercak berwarna coklat tua akibat gigitan hama wereng (Sodiq & Mujoko, 2017).



Gambar 2.3 Penyakit daun tungro

d. Bercak Coklat

Menurut Sodiq dan Mujoko (2017), penyakit bercak coklat pada tanaman padi disebabkan oleh cendawan *Cercospora oryzae*. Serangan penyakit ini mengganggu proses fotosintesis, sehingga pertumbuhan tanaman menjadi terhambat. Gejala umumnya berupa munculnya garis-garis berwarna coklat sejajar dengan urat daun, dengan panjang sekitar 0,5 cm dan lebar sekitar 1 mm. Jika infeksi meluas, daun akan tampak menguning dimulai dari bagian ujungnya.



Gambar 2.4 Penyakit daun bercak coklat

2. *Machine Learning* (ML)

Machine learning merupakan suatu pendekatan dalam kecerdasan buatan yang membuat sistem dapat belajar dari data dan membuat prediksi atau keputusan tanpa perlu diperintah secara eksplisit untuk melakukan tugas tersebut. cara kerja *machine learning* adalah dengan cara melatih model menggunakan data agar dapat mengenali pola dan menghasilkan output sesuai pengalaman belajar dari data yang diberikan. Dengan kata lain, sistem tidak diberi instruksi, melainkan memperoleh hasil dari proses pembelajaran dari data yang tersedia (Deuschle, 2021). Salah satu tugas utama dalam *machine learning* adalah klasifikasi, yaitu proses memprediksi suatu label atau kategori dari data yang telah dipelajari sebelumnya. Tujuan dari klasifikasi adalah untuk membedakan data ke dalam kelas-kelas tertentu, misalnya mengidentifikasi apakah suatu objek termasuk ke dalam kategori A atau kategori B. Beberapa algoritma yang umum digunakan dalam membangun model klasifikasi antara lain *k-Nearest Neighbors* (*k*-NN), *decision tree*, dan *logistic regression* (Ernst & Schweikard, 2021).

3. *Computer Vision*

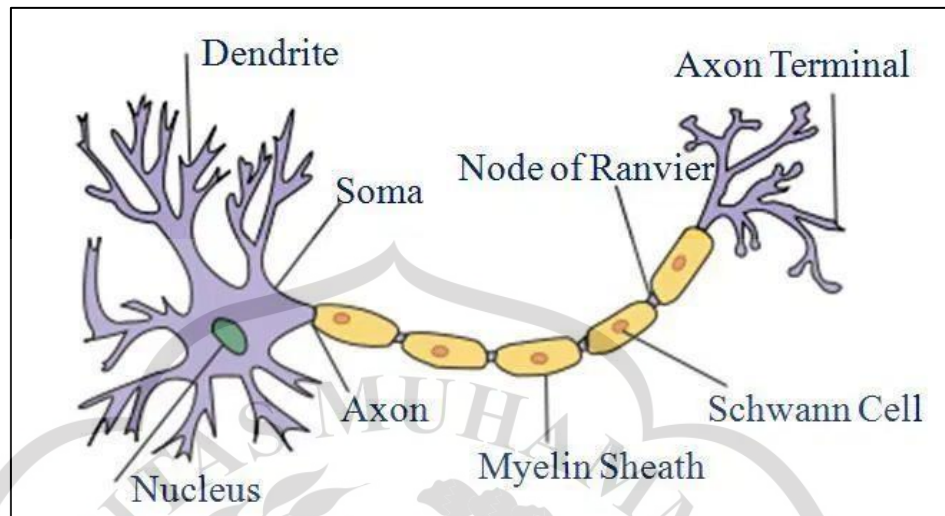
Computer vision merupakan bidang yang mendapatkan manfaat dari berkembangnya teknologi *machine learning*, khususnya *deep learning*. *Computer vision* berfungsi untuk memungkinkan sistem komputer untuk mengekstraksi, menginterpretasi, dan memahami informasi dari data visual seperti gambar maupun video. *Convolutional neural network* (CNN) menjadi salah satu arsitektur paling umum yang digunakan dalam *computer vision* karena kemampuannya dalam mengekstraksi fitur dari citra secara efektif (The Royal Society, 2017).

4. Citra Digital

Citra digital merupakan representasi visual dari suatu objek yang disimpan dalam bentuk digital, sehingga nantinya dapat diolah oleh komputer itu sendiri. Citra digital terdiri dari banyak piksel kecil yang tersusun dalam grid, yang mana setiap piksel memiliki nilai numerik sendiri. Citra digital disini berperan sebagai data utama yang akan dianalisis dan diolah menggunakan algoritma *convolutional neural network* (CNN). Citra digital dimanfaatkan dalam penelitian ini karena mampu menyimpan informasi visual dalam bentuk piksel yang dapat diinterpretasikan oleh komputer (Khairullah et al., 2020).

5. *Artificial Neural Network* (ANN)

Kecerdasan buatan atau *Artificial Intelligence* (AI) adalah bidang studi yang berfokus pada cara membuat komputer mampu melaksanakan tugas seperti yang dilakukan manusia. Salah satu bentuk AI adalah Jaringan Syaraf Tiruan (JST) atau *Artificial Neural Network* (ANN), yang meniru mekanisme kerja otak manusia dalam mereplikasikan kecerdasannya. Otak manusia menerima dan mengolah informasi dari indera melalui neuron yang saling terhubung dalam jaringan syaraf manusia. Proses ini terjadi karena sinyal listrik yang melewati neuron. Jaringan syaraf bekerja dengan logika membuka dan menutup gerbang, memungkinkan sinyal listrik mengalir sesuai dengan karakteristik objek yang diterima dari indera (Alim, 2020).



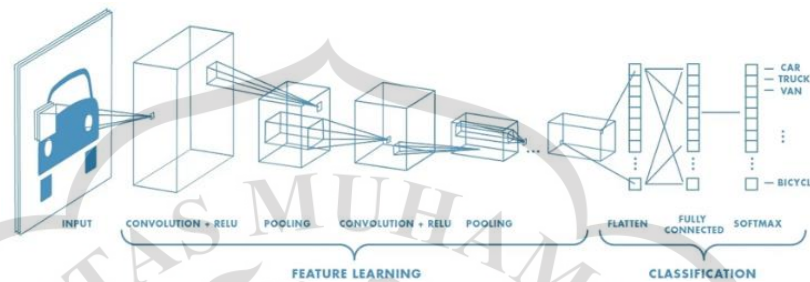
Gambar 2.5 Jaringan syaraf pada manusia

Artificial Neural Network (ANN) beroperasi dengan prinsip yang menyerupai jaringan syaraf biologis, sehingga dapat dianggap sebagai representasi matematis dari mekanisme sebuah jaringan syaraf manusia, sebagaimana yang ditunjukkan dalam Gambar 2.5. *Artificial Neural Network* (ANN) menggunakan model berbasis neuron buatan untuk memproses informasi dan meniru cara biologis yang saling terhubung dan bekerja dalam sistem saraf manusia (Alim, 2020).

6. *Convolutional Neural Network* (CNN)

Convolutional neural network (CNN) merupakan arsitektur dari *deep learning* yang dirancang untuk memproses data dalam bentuk citra, dengan kemampuan unggulnya yang dapat mengenali pola dan fitur visual (Maulana & Rochmawati, 2020). *Convolutional neural network* (CNN) digunakan dalam mengolah data dengan bentuk grid, seperti gambar dua dimensi, dan juga mampu menangani data dengan dimensi lebih tinggi seperti video. CNN memiliki cara kerja yang serupa dengan jaringan syaraf buatan standar, tetapi menggunakan kernel dua dimensi atau lebih tinggi pada setiap unit dalam lapisan untuk melakukan proses konvolusi. Kernel ini berfungsi untuk menggabungkan fitur dari elemen spasial dengan pola

spasial serupa dalam data masukan. Selain itu, CNN menggunakan beberapa parameter untuk mengurangi jumlah variabel sehingga proses pembelajaran menjadi lebih efisien (Amiril, 2020). Berikut ilustrasi secara umum dari CNN pada Gambar 2.6.



Gambar 2.6 *Convolutional neural network*

Berikut ini adalah penjelasan dari dua bagian lapisan arsitektur pada Gambar 2.6 menurut Amiril (2020).

a. *Feature Learning (Feature Extraction Layer)*

Dalam proses pembelajaran fitur (*Feature Learning*), terdapat lapisan yang berfungsi dalam menerima masukan (*input*) berupa gambar secara langsung pada tahap awal, lalu mengolahnya hingga menghasilkan keluaran (*output*) berupa array multidimensi. Lapisan yang terlibat dalam proses ini terdiri dari lapisan konvolusi dan lapisan *pooling*, dimana masing-masing lapisan tersebut berperan dalam membentuk *feature maps* yang terdiri dari angka-angka yang mempresentasikan gambar. Hasil dari lapisan ini kemudian diteruskan ke lapisan klasifikasi untuk tahap pengenalan pola atau kategori.

b. *Classification Layer (Lapisan Klasifikasi)*

Lapisan ini terdiri dari beberapa lapisan dengan neuron yang sepenuhnya terhubung (*fully connected*) dengan lapisan lainnya. Masukan (*input*) yang diterima berasal dari lapisan *output* bagian *feature learning*, kemudian diproses melalui *flattening*, dengan penambahan beberapa *hidden layer* pada jaringan *fully connected*.

Proses ini menghasilkan keluaran berupa nilai akurasi untuk klasifikasi setiap kelas.

Berikut adalah penjelasan dari setiap komponen penting dalam arsitektur CCN pada Gambar 2.6 menurut Mukhtarom (2020).

a. Konvolusi

Konvolusi adalah salah satu tahap dalam arsitektur CNN. Secara sistematis, konvolusi mengacu pada penerapan suatu fungsi terhadap *output* dari fungsi lain secara berulang. Dalam pengolahan cita, proses ini dilakukan dengan menerapkan *kernel* pada citra.

Rumus konvolusi dapat ditemukan pada persamaan (1), di mana *I* merepresentasikan *input* citra, sedangkan *K* berfungsi sebagai *kernel* atau filter konvolusi.

$$S(i, j) = (I * K)(i, j) = \sum_m \sum_n I(m, n)K(i - m, j - n) \quad (1)$$

Keterangan:

S(*i,j*) = fungsi hasil operasi konvolusi

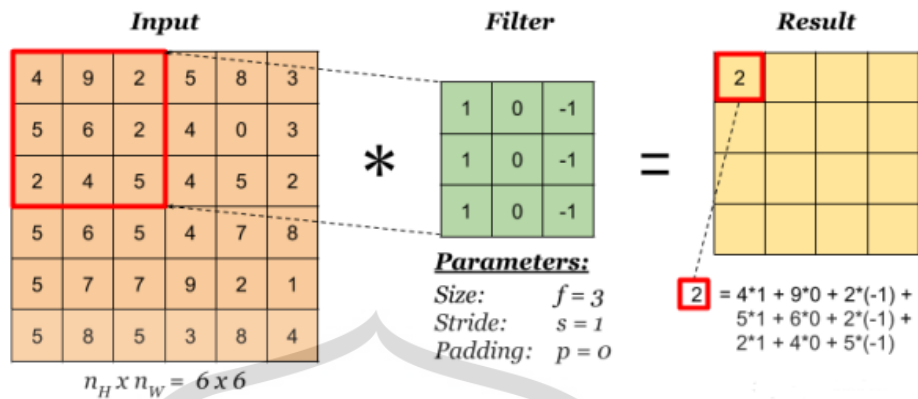
I = *Input*

K = *kernel* atau filter

i,j = *pixel input*

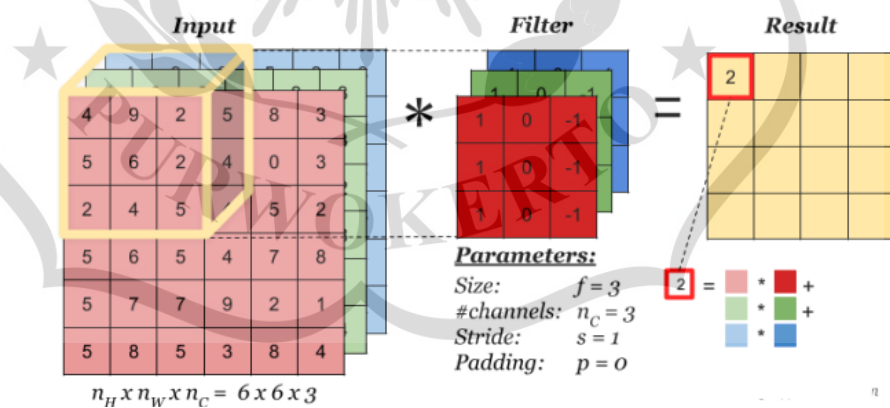
m.n = *pixel kernel*

Konvolusi pada citra bertujuan untuk mengekstraksi fitur dari *input* gambar, menghasilkan transformasi linier dari data masukan (*input*). Pada Gambar 2.7, operasi konvolusi diilustrasikan menggunakan matriks yang mempresentasikan citra dalam bentuk numerik. Proses ini dilakukan antara matriks *input* berukuran 6x6 dan matriks filter berukuran 3x3, dengan *stride* sebesar satu serta penerapan *zero padding*. Hasil akhir berupa matriks hasil konvolusi yang merepresentasikan fitur yang telah dieksekusi.



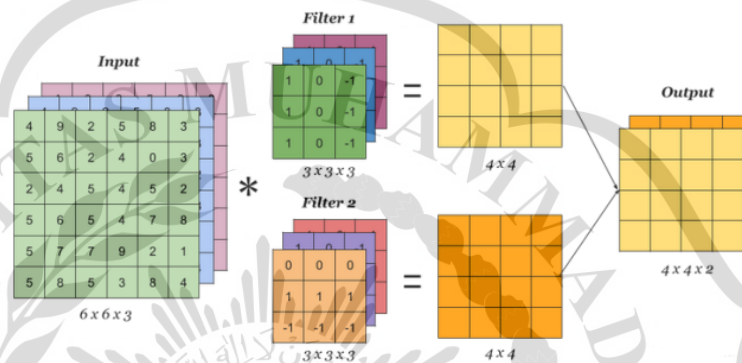
Gambar 2.7 Operasi konvolusi pada citra *grayscale*

Operasi konvolusi pada Gambar 2.7 hanya berlaku untuk untuk citra *grayscale*, yang memiliki satu lapisan dalam representasi numeriknya. Sedangkan untuk citra RGB, representasinya terdiri dari tiga lapisan matriks. Dalam proses konvolusi, setiap lapisan matriks akan diproses secara terpisah menggunakan filter, lalu hasil dari masing-masing operasi konvolusi akan dijumlahkan. Akhirnya, proses ini menghasilkan satu lapisan matriks hasil konvolusi, sebagaimana ditunjukkan pada Gambar 2.8.



Gambar 2.8 Operasi konvolusi pada citra RGB

Dalam operasi konvolusi pada CNN, umumnya tidak hanya digunakan satu lapisan filter. Pada Gambar 2.9, diilustrasikan bagaimana konvolusi diterapkan pada citra RGB dengan dua filter. Proses konvolusi untuk masing-masing filter dilakukan dengan cara yang sama seperti pada Gambar 2.8. Jumlah lapisan output dari hasil konvolusi dengan banyak filter akan sesuai dengan jumlah filter yang digunakan dalam proses tersebut.



Gambar 2.9 Operasi konvolusi citra RGB lapisan lebih dari satu

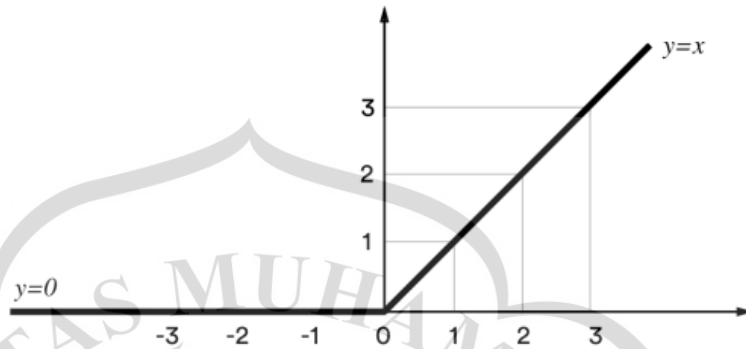
b. Fungsi Aktivasi

Fungsi aktivasi berperan sebagai lapisan pemetaan *non-linear* yang meningkatkan kemampuan klasifikasi jaringan saraf tiruan. Salah satu fungsinya adalah memungkinkan jaringan untuk menyelesaikan tugas-tugas yang bersifat *non-linear*. Tanpa fungsi aktivasi, jaringan saraf hanya akan melakukan operasi *linear*, sehingga kemampuannya terbatas pada tugas-tugas *linear* saja. Sementara itu, sebagian besar permasalahan di dunia nyata bersifat *non-linear*, sehingga fungsi aktivasi menjadi elemen penting dalam jaringan saraf tiruan.

Rectified Linear Unit (ReLU) adalah salah satu jenis fungsi aktivasi yang umum digunakan dalam *Convolutional Neural Network*. ReLU memiliki bentuk fungsi sebagai berikut:

$$f(x) = \begin{cases} x, & x \geq 0 \\ 0, & x < 0 \end{cases} \quad (2)$$

Pada fungsi aktivasi ReLU, semua nilai x negatif akan dipetakan ke 0, seperti pada persamaan (2).

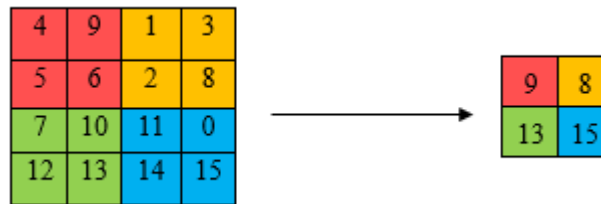


Gambar 2.10 Grafik aktivasi ReLU

Kelebihan fungsi ReLU yaitu:

- 1) Fungsi ReLU cenderung lebih cepat saat digunakan bersama *stochastic gradient descent* dibandingkan dengan fungsi aktivasi sigmoid atau tanh.
- 2) Proses komputasi pada neuron dengan fungsi ReLU lebih sederhana dibandingkan sigmoid atau tanh, karena fungsi ReLU hanya melakukan operasi *thresholding* terhadap matriks aktivasi pada nilai 0 saja, tanpa melibatkan perhitungan eksponensial.
- 3) *Pooling*

Pooling bertujuan untuk mengurangi kompleksitas pada lapisan berikutnya dalam jaringan saraf. Dalam konteks pengolahan citra, teknik ini bertujuan menurunkan resolusi citra, sehingga jumlah parameter yang harus diperbarui berkurang. Hal ini membantu mempercepat proses komputasi sekaligus mengurangi risiko *overfitting*. Meskipun *pooling* tidak memengaruhi jumlah lapisan dalam jaringan, teknik ini tetap berperan penting dalam mengoptimalkan ekstraksi fitur. Salah satu metode *pooling* yang sering digunakan adalah *max pooling*.

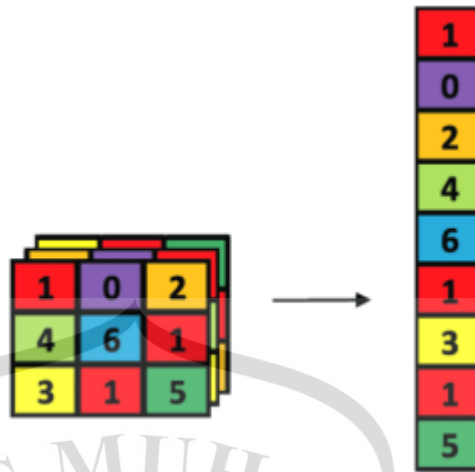


Gambar 2.11 Operasi *pooling*

Dalam operasi *pooling*, terdapat dua parameter penting yang perlu diperhatikan, yaitu filter dan *stride*. Filter adalah ukuran matriks yang digunakan untuk menjalankan proses *pooling*, sedangkan *stride* menunjukkan seberapa jauh pergeseran baris dan/atau kolom setelah satu kali proses *pooling* dilakukan. Sebagai contoh pada Gambar 2.11, digunakan filter berukuran 2x2 dengan *stride* sebesar 2. Jenis *pooling* yang dipakai adalah *max pooling*. Proses dilakukan dari kiri atas menuju kanan bawah. Setiap hasil dari *max pooling* merupakan nilai terbesar dari area yang tercakup oleh filter. Pada proses pertama, area filter mencakup nilai 4, 9, 5, dan 6. Nilai hasil dari proses tersebut adalah 9, karena merupakan yang paling tinggi di antara keempat nilai tersebut.

c. *Flatten*

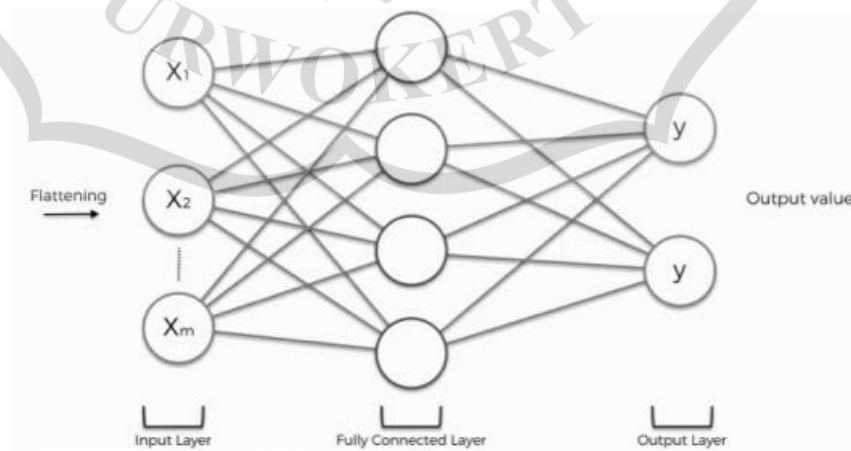
Flatten adalah metode yang digunakan untuk mengubah data dari bentuk matriks berdimensi-n menjadi bentuk satu dimensi. Proses ini biasanya dilakukan setelah tahap akhir ekstraksi fitur. *Output* yang awalnya berbentuk matriks multidimensi akan diubah menjadi vektor satu dimensi agar dapat diproses lebih lanjut dalam lapisan *fully connected* untuk keperluan klasifikasi.



Gambar 2. 12 Ilustrasi operasi *flatten*

d. *Fully Connected Layer*

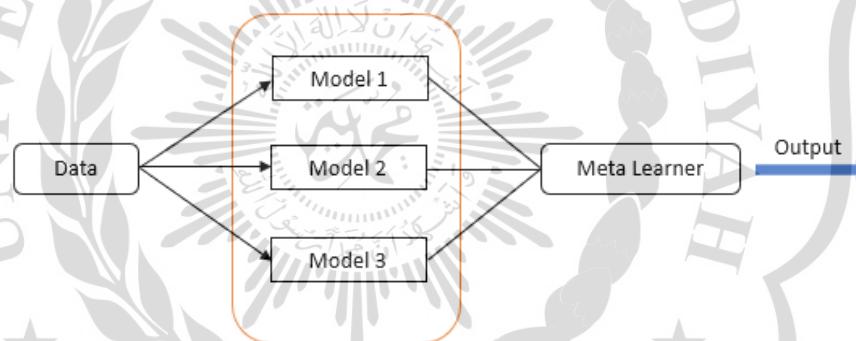
Fully connected layer adalah bagian dari CNN yang memiliki kemiripan dengan jaringan saraf tiruan pada umumnya. Pada lapisan ini, setiap node terhubung secara langsung dengan seluruh node di lapisan sebelumnya maupun di lapisan setelahnya. Lapisan ini biasanya mengandung jumlah parameter yang paling banyak dan memerlukan waktu komputasi yang cukup tinggi selama proses pelatihan. Seperti yang ditunjukkan pada Gambar 10, input untuk *fully connected layer* berasal dari hasil proses *flattening* sebagaimana diperlihatkan pada Gambar 2.13.



Gambar 2.13 *Fully connected layer*

7. Ensemble Stacking

Ensemble stacking merupakan teknik dalam *ensemble learning* yang menggabungkan beberapa prediksi dari model pembelajaran dasar (*base learners*) dengan melatih model pembelajaran lain yang disebut *meta-learner*. Model-model dasar yang dipakai sebagai *base learner* bisa berasal dari algoritma yang berbeda yang dilatih secara independen dengan menggunakan data yang sama. Dan hasil dari prediksi model-model ini yang nantinya akan dijadikan input untuk *meta-learner*. Dan tugas dari *meta-learner* itu sendiri adalah mempelajari cara terbaik menggabungkan prediksi tersebut agar memperoleh hasil akhir yang lebih baik (Kumar & Jain, 2020). Metode *stacking* memiliki 2 level berbeda yaitu *level-0* dengan *base learner* dan *level-1* dengan *meta-learner*.



Gambar 2.14 Tahapan metode *ensemble stacking*

8. Confusion Matrix

Confusion matrix merupakan metode pengukuran kinerja yang digunakan dalam klasifikasi. *Confusion matrix* digunakan untuk menampilkan visualisasi hasil prediksi model terhadap data uji. Kinerja model dievaluasi menggunakan beberapa metrik, yaitu *accuracy*, *precision*, *recall*, dan *F1-score*. Berikut penjelasan beberapa metrik dalam *Confusion matrix* menurut Akram et al. (2023).

a. *Accuracy*

Accuracy adalah matrik dalam *confusion matrix* yang digunakan dalam mengukur perbandingan seluruh prediksi yang benar terhadap total nilai kelas yang ada. Berikut rumus *Accuracy* dapat dilihat pada persamaan (3).

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} \quad (3)$$

b. *Precision*

Precision digunakan dalam menghitung akurasi hasil prediksi nilai positif dari seluruh hasil yang diprediksi sebagai positif. Rumus perhitungan *Precision* dapat dilihat pada persamaan (4).

$$Precision = \frac{TP}{TP + FP} \quad (4)$$

c. *Recall*

Recall adalah metrik yang digunakan untuk mengukur seberapa banyak prediksi yang relevan dan sesuai dengan kelas yang seharusnya. Berikut rumus *recall* dapat dilihat pada persamaan (5).

$$Recall = \frac{TP}{TP + FN} \quad (5)$$

d. *F1-score*

F1-score adalah perbandingan antara rata-rata *precision* dan *recall*. Rumus perhitungan *F1-score* dapat dilihat pada persamaan (6).

$$F1 - score = \frac{2 * precision * recall}{precision + recall} \quad (6)$$

Keterangan:

- TP = *True Positive*
- TN = *True Negative*
- FP = *False Positive*
- FN = *False Negative*

9. Logistic Regression

Logistic regression merupakan sebuah metode regresi yang digunakan untuk memodelkan hubungan antara satu atau lebih variabel bebas dengan variabel yang bersifat dependen. Tujuan utama dari metode ini adalah untuk memperoleh model terbaik yang secara statistik dan klinis dapat menjelaskan hubungan antara variabel bebas dengan kemungkinan (probabilitas) terjadinya suatu peristiwa tertentu.

Menurut Scott et al. (2013) *logistic regression* digunakan ketika variabel *output* (respon) merupakan variabel kategori yang bernilai 0 atau 1. Model ini memodelkan probabilitas terjadinya suatu kejadian (misalnya, $Y = 1$) sebagai fungsi dari variabel-variabel penjelas. dengan bentuk fungsi logistik yang menghasilkan nilai antara 0 hingga 1. Model dasar *logistic regression* dituliskan dengan persamaan (7)

$$\pi(x) = \frac{e^{\beta_0 + \beta_1 x}}{1 + e^{\beta_0 + \beta_1 x}} \quad (7)$$

Keterangan:

- $\pi(x)$ = Probabilitas kejadian (misalnya kelas 1) yang diprediksi oleh model logistik
- β_0 = Bias
- β_1 = Koefisien regresi
- x = Nilai Input