BAB II

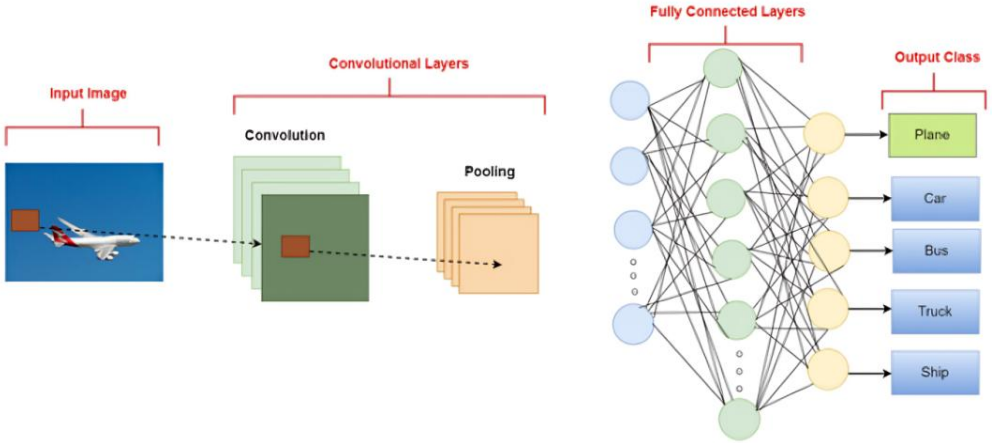
LANDASAN TEORI

2.1 *Convolutional Neural Network*

Convolutional Neural Network (CNN) merupakan bagian dari *deep neural network* yang sudah terbukti dengan keefektivitasan yang tinggi pada pengenalan gambar dan vidio. CNN biasanya memiliki beberapa tipe lapisan, seperti *convolutional layers*, *activation layers*, *pooling layers*, dan *fully connected layers*, yang dimana semua lapisan berkontribusi dalam kemampuan jaringan untuk mengenali pola dan fitur (Zafar dkk, 2024). *Convolution layers* menerapkan filter yang dapat dipelajari pada input data, memungkinkan jaringan untuk mengekstraksi fitur yang berguna dari gambar. Lapisan-lapisan ini mempunyai bobot yang dipelajari selama proses pelatihan, meningkatkan kemampuan model untuk mengenali pola dan objek (Al-Malah, 2023). *Pooling layer* umumnya mengikuti *convolutional layer*, berperan penting untuk mengurangi kerumitan kompleksitas komputasi dengan melakukan *down-sampling* dan menjaga fitur-fitur penting. Peran pada *pooling layers* yaitu untuk menjaga keseimbangan simetri informasi di seluruh jaringan yang sangat krusial untuk mendapatkan performa yang optimal (Zafar dkk, 2024).

Seperti yang dapat dilihat pada gambar 2.1, merupakan bentuk arsitektur umum untuk tugas identifikasi gambar sederhana. Penginputan gambar secara langsung kedalam jaringan melibatkan beberapa lapisan konvolusi dan *pooling*, yang kemudian lapisan-lapisan tersebut dimasukkan ke dalam satu atau lebih lapisan *fully connected*, kemudian klasifikator memberikan hasil evaluasi

berdasarkan lapisan *full connected*. Meskipun ini adalah contoh desain dasar arsitektur CNN yang paling umum dalam literatur, telah dilakukan banyak pengembangan pada arsitektur yang telah diusulkan baru-baru ini untuk meningkatkan akurasi identifikasi gambar ataupun untuk mengurangi beban komputasi (Zafar dkk, 2024).



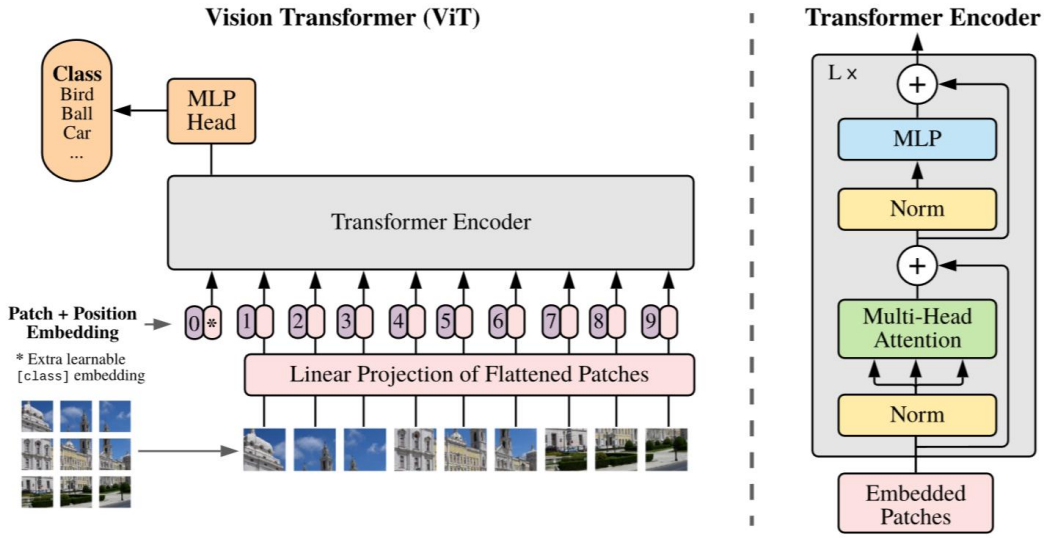
Gambar 2.1 Arsitektur CNN Sederhana (Zafar dkk, 2024)

2.2 *Vision Transformer*

Vision Transformer merupakan arsitektur jaringan saraf yang menggunakan model *transformer*, awalnya model *transformer* didesain untuk pemrosesan bahasa alami tetapi, setelah dikembangkan model *transformer* bisa digunakan untuk pemrosesan gambar. Model *transformer* memproses gambar dengan membagi gambar menjadi beberapa *patch* dan menggunakan mekanisme *attention* untuk menangkap hubungan spasial antar bagian gambar (Park dkk, 2024).

Pada ilustrasi yang dapat dilihat pada gambar 2.2, model *transformer* standar menerima input yang berbentuk urutan 1D token *embedding*. Untuk menangani gambar 2D, maka dilakukannya perubahan gambar dengan formula

$x \in \mathbb{R}^{H \times W \times C}$ sehingga menjadi urutan *patch* 2D yang telah dipipihkan dengan formula $x_p \in \mathbb{R}^{N \times (P^2.C)}$, dimana (H, W) adalah resolusi dari gambar asli, C adalah jumlah saluran, (P, P) adalah resolusi setiap *patch* gambar, dan $N = HW|P^2$ adalah jumlah *patch* yang dihasilkan dan berfungsi sebagai panjang urutan input yang efektif. *Transformer* menggunakan ukuran vektor laten konstan D pada semua lapisannya, sehingga pada ViT diperlukan untuk memipihkan *patch* dan memetakannya ke D dimensi menggunakan proyeksi linier seperti pada persamaan 2.1 dan keluaran dari proyeksi ini adalah *patch embeddings* (Dosovitskiy dkk, 2021).



Gambar 2.2 Arsitektur Model *Vision Transformer* (Dosovitskiy dkk, 2021)

Mirip dengan pemodelan pada *Bidirectional Encoder Representations from Transformers* (BERT) untuk token [*class*], tetapi pada ViT ditambahkan *learnable embedding* pada awal urutan *patch* yang telah dienkapsulasi ($z_0^0 = x_{class}$), kemudian status dari *embedding* ini pada *output transformer encoder* (z_L^0) berfungsi sebagai representasi gambar y dengan formula $y = \mathrm{LN}(z_L^0)$. Selama masa *pre-*

training ataupun *fine-tuning*, sebuah *classification head* dipasangkan pada z_L^0. *Classification head* diimplementasikan menggunakan *Multi-Layer Perceptron* (MLP) dengan satu lapisan tersembunyi pada masa *pre-training* dan satu lapisan linear pada masa *fine-tuning*. Kemudian, *position embedding* ditambahkan ke *patch embedding* untuk mempertahankan posisi informasi. ViT menggunakan *learnable embedding* posisi 1D standar, karena tidak adanya peningkatan performa yang signifikan jika menggunakan *embedding* posisi 2D. Urutan vektor *embedding* yang dihasilkan berfungsi sebagai *input* ke *encoder* (Dosovitskiy dkk, 2021).

Menurut (Vaswani dkk, 2017), *encoder* pada *transformer* terdiri dari beberapa lapisan yang bergantian pada *Multi-Headed Self-Attention* (MSA) dan blok *Multi-Layer Perceptron* (MLP), adapun formula yang dapat dilihat pada persamaan 2.2 dan 2.3.

Layer Normalization (LN) yang diterapkan sebelum setiap blok dan *residual connections* digunakan setelah setiap blok (Wang dkk, 2019). MLP terdiri dari dua lapisan dengan fungsi aktivasi *Gaussian Error Linear Unit* (GELU) *non-linearitas* (Dosovitskiy dkk, 2021).

$$z_0 = \left[x_{class}; x_p^1\mathbf{E}; x_p^2\mathbf{E}; \dots; x_p^N\mathbf{E}\right] + \mathbf{E}_{pos}, \mathbf{E} \in \mathbb{R}^{(\boldsymbol{P^2.C})\times \boldsymbol{D}}, \mathbf{E}_{\boldsymbol{pos}} \in \mathbb{R}^{(\boldsymbol{N+1})\times \boldsymbol{D}} \quad (2.1)$$

$$z'_\ell = \mathrm{MSA}\big(\mathrm{LN}(z_{\ell-1})\big) + z_{\ell-1}, \qquad \ell = 1 \dots L \quad (2.2)$$

$$z_\ell = \mathrm{MLP}\big(\mathrm{LN}(z'_\ell)\big) + z'_\ell, \qquad \ell = 1 \dots L \quad (2.3)$$

Dimana z_0 merupakan *input* awal ke *transformer encoder* atau hasil *embedding* dari semua *patch*. x_{class} merupakan token [CLS] yang akan menjadi representasi keseluruhan gambar. $x_p^1\mathbf{E}$ merupakan hasil *linear projection* dari patch

ke-i ke dalam dimensi *embedding* melalui matriks **E**. $\mathbf{E}_{pos}$ merupakan *position embeddings* agar model dapat mengetahui urutan *patches.*

z'_{ℓ} merupakan output dari proses *attention* sebelum masuk ke lapisan MLP. MSA yaitu *Multi-Head Self-Attention* yang berguna untuk mengekstraksi hubungan antar bagian gambar. LN yaitu *layer normalization* yang berguna untuk menormalkan *input* sebelum dimasukkan pada MSA. $z_{\ell-1}$ merupakan lapisan *input* ke-l yang merupakan *output* dari lapisan sebelumnya. $+z_{\ell-1}$ merupakan bagian dari *residual connection* untuk menambahkan input kembali.

z_{ℓ} merupakan *output* akhir dari lapisan ke-l dalam *transformer encoder*. MLP merupakan *feed-forward neural network* untuk memproses informasi dari setiap token secara independen dan meningkatkan kapasitas representasi dari model. LN merupakan fungsi untuk menormalkan data *input* sehingga memiliki rata-rata dan variasi tertentu. z'_{ℓ} merupakan *input* untuk lapisan ke-l setelah melewati MSA dan *residual connection* sebelumnya, atau biasa disebut *hidden state* sementara. $+z'_{\ell}$ merupakan teknik untuk menghindari hilangnya informasi penting dengan menggabungkan input awal dengan hasil dari MLP.

Terdapat tiga varian ViT yaitu ViT-*Base*, ViT-*Large*, dan ViT-*Huge*. Digunakannya notasi singkat untuk mengindikasikan ukuran model dan ukuran *input patch*, seperti pada ViT-L/16 yang berarti varian "*Large*" dengan 16x16 ukuran *input patch*, panjang *sequence* dari *transformer* berbanding terbalik pada kuadrat dari ukuran *patch*, sehingga model dengan ukuran *patch* yang lebih kecil memerlukan komputasi yang lebih tinggi (Dosovitskiy dkk, 2021).

Pada tabel 2.1, terdapat tiga varian dari model ViT yang memiliki konfigurasi yang berbeda-beda, seperti pada ViT-*Base* hanya memiliki 12 lapisan pada modelnya, sementara ViT-*Large* memiliki 24 lapisan, dan ViT-*Huge* memiliki 32 lapisan. Terdapat perbedaan pada *hidden size D* atau jumlah fitur dalam setiap token yang masuk kedalam model. Terdapat perbedaan pada ukuran MLP yang berguna untuk mengolah representasi fitur setiap token. Terdapat perbedaan juga pada *heads* yang dimana merupakan jumlah *attention heads* pada lapisan MSA dari setiap *Transformer encoder block*. Perbedaan terakhir yaitu pada parameter yang dimana merupakan nilai-nilai yang dipelajari oleh model selama masa pelatihan.

Tabel 2.1 Detail varian model ViT (Dosovitskiy dkk, 2021)

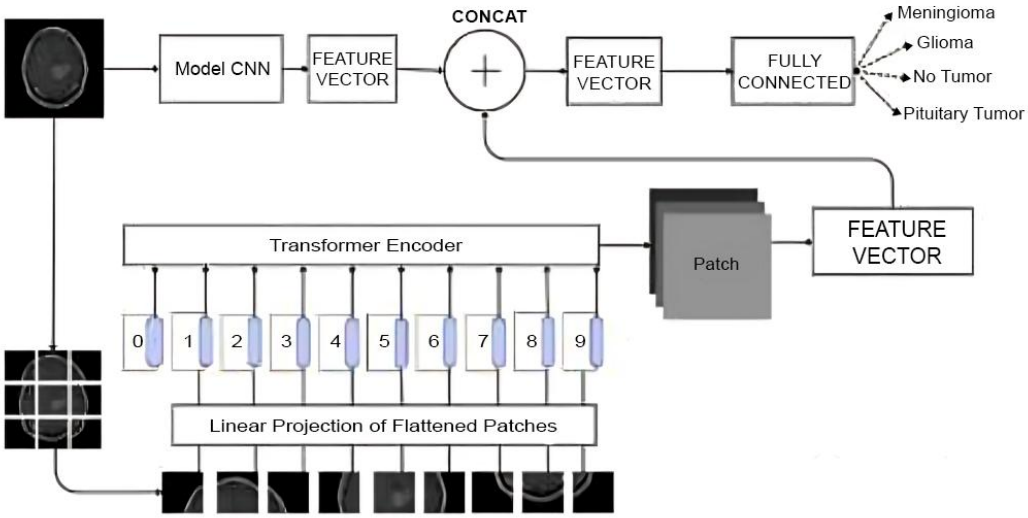
Model	Lapisan	*Hidden Size D*	Ukuran MLP	*Heads*	Parameter
ViT-*Base*	12	768	3072	12	86 Jt
ViT-*Large*	24	1024	4096	16	307 Jt
ViT-*Huge*	32	1280	5120	16	632 Jt

2.3 *Hybrid CNN-ViT*

Metode *hybrid* CNN-ViT merupakan sebuah metode yang menggabungkan CNN untuk fitur ekstraksi lokal dan ViT untuk menangkap dependensi gambar global, metode *hybrid* CNN-ViT dapat secara efektif untuk mengenali identifikasi gambar dengan akurasi yang tinggi, bahkan pada *dataset* yang kecil (Xu dkk, 2024).

Pada gambar 2.3, merupakan sebuah arsitektur *hybrid* CNN-ViT paralel, model ini terdiri dari dua model utama yaitu CNN dan ViT, penggunaan model CNN dan ViT untuk mengambil *feature vector* untuk digabungkan sebelum

dilakukan klasifikasi. Model ini nanti akan menerima masukan berupa satu gambar yang dimana terdapat gambar yang akan dimasukkan pada model CNN dan gambar yang akan dimasukkan kedalam ViT secara paralel, lalu kedua model ini akan menghasilkan *feature vector* yang kemudian dilakukan *concat* untuk menggabungkan kedua *feature vector* dari hasil model CNN dan ViT untuk dihubungkan pada lapisan *fully connected* yang kemudian dilakukan klasifikasi (Sukandar dkk, 2024).



Gambar 2.3 Model *Hybrid* CNN-ViT (Sukandar dkk, 2024)

2.4 *Metode Evaluasi Model*

Untuk memastikan keandalan model dalam memprediksi, diperlukannya proses evaluasi model untuk menilai keakuratan dan kefektifitasan model dalam mendeteksi penyakit (Khandagale dan Patil, 2023). Terdapat beberapa metode evaluasi yang akan digunakan untuk memastikan keandalan model yang diusulkan penulis seperti yang diuraikan dibawah.

2.4.1 *Confusion Matrix*

Confusion matrix merupakan sebuah tabel evaluasi yang menunjukkan jumlah prediksi benar dan salah untuk masing-masing kategori atau kelas dalam model klasifikasi. *Confusion matrix* berfungsi untuk memetakan hasil prediksi terhadap kondisi aktual melalui empat kemungkinan hasil yang dapat dilihat pada tabel 2.2 (Google, 2025).

Tabel 2.2 *Confusion Matrix*

	Actual positive	*Actual negative*
Predicted positive	*True positive* (TP)	*False Positive* (FP)
Predicted negative	*False Negative* (FN)	*True Negative* (TN)

2.4.2 *Accuracy*

Accuracy merupakan proporsi dari semua klasifikasi yang benar, baik positif ataupun negatif, model yang bagus dapat tidak mempunyai FP dan FN sehingga mendapatkan akurasi sebesar 1.0 atau 100% (Google, 2025). Terdapat penulisan secara metematikanya didefinisikan sebagai:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{2.4}$$

2.4.3 *Recall*

Recall merupakan proposisi dari semua positif aktual yang diklasifikasikan dengan benar sebagai positif, model hipotesis yang sempurna dapat tidak memiliki FN sehingga memiliki recall sebesar 1.0 atau 100% tingkat deteksi (Google, 2025). Terdapat penulisan secara matematikanya didefinisikan sebagai:

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

2.4.4 *Precision*

Precision merupakan proporsi dari semua model klasifikasi positif yang benar-benar positif, model hipotesis yang sempurna dapat tidak memiliki FP sehingga mendapatkan presisi 1.0 atau 100% (Google, 2025). Terdapat penulisan secara matematikanya didefinisikan sebagai:

$$Precision = \frac{TP}{TP + FP} \quad (2.6)$$

2.4.5 F1 *Score*

F1 *score* merupakan rata-rata harmonik dari *precision* dan *recall*, matriks ini menyeimbangkan pentingnya *precision* dan *recall*, matriks ini lebih cocok untuk kumpulan data yang kelasnya tidak seimbang, ketika *precision* dan *recall* memiliki perbedaan yang jauh, hasil F1 *score* akan mirip dengan metrik yang paling buruk (Google, 2025). Terdapat penulisan secara matematikanya didefinisikan sebagai:

$$F1 = \frac{2TP}{2TP + FP + FN} \quad (2.7)$$

2.4.6 ROC AUC

Area Under the Receiver Operating Characteristic Curve (ROC AUC) merupakan suatu pengukuran performa untuk pengklasifikasi, metode ini mengindikasikan probabilitas bahwa suatu contoh positif yang dipilih secara acak diberi nilai yang lebih tinggi dibandingkan pada suatu contoh negatif yang dipilih secara acak, dengan nilai dari nol sampai satu, pada kasus model yang memiliki empat kelas, maka digunakan pendekatan *One-vs-Rest* (OvR), dimana ROC AUC

dihitung untuk setiap kelas dengan menganggap kelas yang dipilih sebagai positif dan kelas lainnya sebagai negatif yang kemudian nilai dari tiap kelas akan diambil rata-ratanya atau disebut *macro average* (Jaskowiak dkk, 2022).

2.4.7 *Class Weight*

Class weight merupakan suatu metode untuk menentukan bobot yang berbeda-beda pada *dataset* yang *imbalanced*, metode ini membantu agar algoritma dapat fokus pada setiap kelas walaupun terdapat kelas yang dominan sehingga dapat meningkatkan akurasi klasifikasi pada model (Bakirarar dan Elhan, 2023). Berdasarkan situs *Scikit-learn* untuk mendapatkan bobot dari kelas x yaitu dengan membagi keseluruhan data dengan total kelas dikali dengan jumlah kelas x atau pada persamaan 2.8 agar model dapat lebih mengenali fitur citra pada kelas minoritas sehingga ketidakseimbangan jumlah data pada kelas tidak membuat bias untuk model.

$$bobot\ x = \frac{keseluruhan\ data}{total\ kelas \times jumlah\ kelas\ x} \tag{2.8}$$

2.5 *Penyakit pada Daun Pisang*

Penyakit pada daun pisang merupakan masalah serius yang dialami oleh petani pisang di Indonesia, penyakit pada daun pisang secara signifikan mengancam produktifitas pisang yang dihasilkan dan memberikan dampak yang signifikan pada pendapatan petani yang pada akhirnya memengaruhi katahanan pangan global, sehingga pentingnya untuk melakukan deteksi dini pada penyakit daun pisang untuk mengurangi penyebaran penyakit, meningkatkan hasil panen, dan menjaga stabilitas hasil panen, adapun penyakit pada daun pisang yaitu *Cordana*, *Pestalotiopsis*, dan *Sigatoka* (Helmawati dan Ema, 2024). Dalam bidang agrikultur,

penyakit pada tanaman seperti yang disebabkan oleh mikroba, jamur, *ringworm*, dan kekurangan nutrisi merupakan masalah utama yang dialami saat ini karena dapat mengurangi produksi hasil dan kualitas panen, bahkan sampai membunuh tanaman, pisang merupakan buah yang paling penting pada wilayah Pasifik dan Asia, tanaman pisang rentan terhadap berbagai penyakit yang bisa diketahui jenis penyakitnya dengan melihat penyakitnya pada daun pisang, karenanya deteksi dini pada daun pisang menjadi suatu hal yang sangat penting untuk mencegah penyebaran penyakit (Raja dan Selvi, 2022).

1. *Cordana*

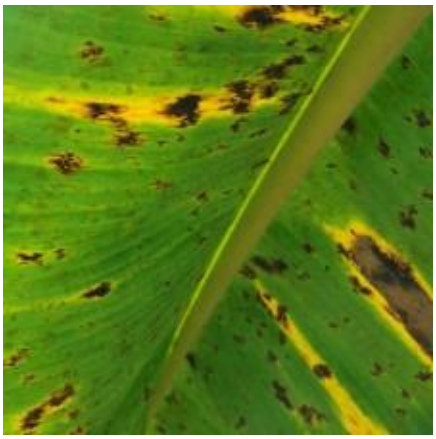
Penyakit *Cordana* pada daun pisang merupakan penyakit yang mempengaruhi tanaman pisang, penyakit *Cordana* disebabkan oleh patogen *Neocordana musae*, penyakit *Cordana* biasa terjadi di daerah tropis dan sub-tropis (Yu dkk, 2024).

Gambar 2.4 Daun Pisang Terinfeksi *Cordana*

2. *Pestalotiopsis*

Penyakit *Pestalotiopsis* menyebabkan penyakit *leaf blight* dengan gejala berupa lesi sempit berwarna cokelat tua yang berkembang menjadi bercak cokelat tidak

beraturan, penyakit *Pestalotiopsis* pada pohon pisang pertama kali dilaporkan pada Juni 2020 dengan kejadian berkisar antara 5-20% (Bhuiyan dkk, 2022).



Gambar 2.5 Daun Pisang Terinfeksi *Pestalotiopsis*

3. Sigatoka

Penyakit *Sigatoka* disebabkan oleh jamur *Mycosphaerella musicola* yang dapat secara signifikan mempengaruhi panen dan kualitas pada pisang, dengan kerugian yang dilaporkan mencapai 65% pada kondisi yang mendukung (Nagesh dkk, 2023).

Gambar 2.6 Daun Pisang Terinfeksi *Sigatoka*

2.6 Penelitian Terkait

Terdapat beberapa penelitian yang menggunakan berbagai pendekatan untuk memodelkan prediksi penyakit pada daun pisang. Temuan dari penelitian terkait memiliki potensi untuk menjadi dasar penting dalam pengembangan model yang lebih akurat dan optimal dalam memprediksi penyakit pada daun pisang. Temuan dari penelitian yang telah didapatkan dibuat kedalam tabel yang dapat dilihat pada tabel 2.3.

Tabel 2.3 Penelitian Terkait

No	Penulis	Algoritma	Fokus Pembahasan
1	(Syihad dkk, 2023)	CNN dengan ResNet50 dan VGG-19.	Model prediksi penyakit pada daun pisang dengan menggunakan metode CNN dengan menggunakan ResNet50 dan VGG-19, yang dimana model CNN dengan ResNet50 mendapatkan akurasi prediksi sebesar 94% akurasi, 88% *precision*, 91% *recall*, dan 89% *F1-score*, sementara pada model CNN dengan VGG-19 mendapatkan akurasi sebesar 91% akurasi. Sehingga model CNN dengan ResNet50 merupakan model pilihan untuk identifikasi penyakit pada daun pisang yang menawarkan peningkatan yang signifikan.

No	Penulis	Algoritma	Fokus Pembahasan
2	(Helmawati dan Ema, 2024)	CNN dengan augmentasi data.	Model prediksi penyakit pada daun pisang dengan menggunakan model CNN dan augmentasi data berhasil mendapatkan akurasi untuk identifikasi penyakit pada daun pisang sebesar 92.85% akurasi, 95.73% *recall*, 93.52% *precision*, dan 94.60% *F1-score*.
3	(Saini, 2024)	*Vision Transformer* (ViT)	Model identifikasi penyakit pada daun pisang menggunakan metode *Vision Transformer* (ViT) berhasil mendapatkan akurasi sebesar 96.88% yang didapatkan pada *epoch* 100, sehingga model ViT efektif dalam mengidentifikasikan penyakit pada daun pisang.
4	(Tiwari, 2024)	*Vision Transformer Batch* 16 (ViT-B16)	Model identifikasi penyakit pada daun pisang dengan menggunakan model ViT-B16 berhasil mengalahkan beberapa arsitektur *deep learning* yang terbaik seperti *Graph Convolutional Networks* (GCN), EfficientNet-B3, MobileNet-V2, dan ResNet-cut, dengan akurasi mencapai 96.88%.

No	Penulis	Algoritma	Fokus Pembahasan
5	(Arifin, 2024)	*Convolutional Neural Network* (CNN)	Model identifikasi penyakit pada daun pisang menggunakan CNN dan arsitektur GoogleNet dengan 1289 data citra daun pisang dengan data latih 1031 dan data uji 258 yang menghasilkan akurasi sebesar 89.58%.
6	(Pratama dkk, 2024)	*Convolutional Neural Network* (CNN) dengan t*ransfer learning*	Model identifikasi penyakit pada daun pisang menggunakan *Convolutional Neural Network* (CNN) dengan metode *transfer learning* pada VGG-19 dan didapatkannya akurasi sebesar 92%, presisi 92%, sensitifitas 91%, dan skor F1 92%.
7	(Srivastav dkk, 2024)	*Convolutional Neural Network* (CNN)	Model identifikasi penyakit pada daun pisang menggunakan *Convolutional Neural Netowrk* (CNN) dengan akurasi tertinggi sebesar 96.52% pada *epoch* 80, yang dimana didapatkan akurasi sebesar 94.23% pada *epoch* 20.
8	(Criollo dkk, 2020)	*Convolutional Neural Network (CNN)*	Model identifikasi penyakit pada daun pisang menggunakan *Convolutional Neural Netowrk* (CNN) tanpa teknik regularisasi mendapatkan hasil yang terbaik dengan

No	Penulis	Algoritma	Fokus Pembahasan
			akurasi sebesar 87.5% dan skor F1 sebesar 87.39% dalam 300 *epochs*.
9	(Prashanthi dkk, 2024)	*Enhanced Vision Transformer (ViT)*	Model identifikasi penyakit pada daun pisang menggunakan *enhanced Vision Transformer* (ViT) menggunakan ViT-B32 yang mendapatkan akurasi sebesar 95.22%, akurasi validasi 96.19%, dan akurasi test 92.33%.
10	(Lubis dan Alifia, 2023)	*Faster R-CNN*	Model Identifikasi penyakit pada daun pisang menggunakan metode *Faster Region-Convolutional Neural Network* (*Faster R-CNN*) yang mendapatkan akurasi sebesar 91.66%.
11	(Tanwar dkk, 2023)	CNN-SVM	Model Identifikasi penyakit pada daun pisang menggunakan metode *hybrid* CNN-SVM dengan pembagian 80% *training* dan 20% *validation* yang mendapatkan rata-rata akurasi sebesar 90%.
12	(Banerjee dkk, 2023)	CNN-SVM	Model Identifikasi penyakit pada daun pisang menggunakan metode *hybrid* CNN-SVM dengan pembagian 80% *training* dan

No	Penulis	Algoritma	Fokus Pembahasan
			20% *validation* yang mendapatkan rata-rata akurasi tertinggi sebesar 94%.
13	(Kalim dkk, 2022)	CNN-RF	Model Identifikasi penyakit pada daun jeruk menggunakan metode *hybrid* CNN-RF, Model CNN sebagai ekstraktor fitur dan *Random Forest* sebagai klasifikasi yang mendapatkan akurasi sebesar 87% pada VGG16-RF.
14	(David dkk, 2021)	CNN-RNN	Model identifikasi penyakit pada daun tomat menggunakan metode *hybrid* CNN-RNN dengan pembagian 80% *training* dan 20% *testing*, mendapatkan akurasi sebesar 81.75% dengan 200 *epochs*.
15	(Thakur dkk, 2023)	CNN-ViT	Model identifikasi penyakit pada daun tanaman menggunakan *dataset* PlantVillage dan metode *hybrid* CNN-ViT mendapatkan akurasi sebesar 98.86% dan presisi sebesar 98.9%.
16	(Rehman dkk, 2024)	CNN-ViT	Model identifikasi tumor otak menggunakan *dataset* SARTAJ dan dengan metode *hybrid* CNN-ViT yang mendapatkan akurasi sebesar 98.1%.

Matriks pengujian yang berisi penulis, metode, arsitektur tambahan, teknik tambahan, dataset, dan akurasi berdasarkan penelitian terdahulu yang dapat dilihat pada tabel 2.4.

Tabel 2.4 Matriks Pengujian

No	**Penulis**	**Metode**	**Arsitektur Tambahan**	**Teknik Tambahan**	***Dataset***	**Akurasi**
1	(Syihad dkk, 2023)	CNN	ResNet50	-	-	94%
2	(Helmawati dan Ema, 2024)	CNN	-	Augmentasi Data	BananaLSD	92.85%
3	(Saini, 2024)	ViT	-	-	-	96.88%
4	(Tiwari, 2024)	ViT	ViT-B16	-	-	96.88%
5	(Arifin, 2024)	CNN	GoogleNet	-	-	89.58%
6	(Pratama dkk, 2024)	CNN	VGG-19	-	BSMRAU	92%

No	Penulis	Metode	Arsitektur Tambahan	Teknik Tambahan	*Dataset*	Akurasi
7	(Srivastav dkk, 2024)	CNN	-	-	-	96.52%
8	(Criollo dkk, 2020)	CNN	-	-	-	87.5%
9	(Prashanthi dkk, 2024)	ViT	ViT-B32	-	*New Plant Disease Dataset*	95.22%
10	(Lubis dan Alifia, 2023)	CNN	*Faster* R-CNN	-	-	91.66%
11	(Tanwar dkk, 2023)	CNN-SVM	-	-	-	90%
12	(Banerjee dkk, 2023)	CNN-SVM	-	-	-	94%
13	(Kalim dkk, 2022)	CNN-RF	VGG-16	-	-	87%

No	Penulis	Metode	Arsitektur Tambahan	Teknik Tambahan	*Dataset*	Akurasi
14	(David dkk, 2021)	CNN-RNN	*Xception*	-	-	81.75%
15	(Thakur dkk, 2023)	CNN-ViT	VGG16, Inception v7	-	PlantVillage	98.86%
16	(Rehman dkk, 2024)	CNN-ViT	-	-	SARTAJ	98.1%
17	Usulan Penelitian	CNN-ViT	ViT-B16	-	BananaLSD	-

Berdasarkan pada tabel 2.3 dan 2.4, terdapat berbagai penelitian identifikasi penyakit pada daun pisang dan juga penelitian identifikasi penyakit pada daun tumbuhan lain seperti yang ada dalam *dataset* PlantVillage dan penelitian identifikasi pada tumor otak, penelitian-penelitian terdahulu ini menggunakan berbagai model, seperti pada penelitian oleh (Syihad dkk, 2023) menggunakan model berbasis CNN dengan ResNet50 dan CNN dengan VGG-19, dari penelitian ini menunjukkan bahwa model CNN dengan ResNet50 mendapatkan akurasi yang lebih tinggi daripada CNN dengan VGG-19. Penelitian yang dilakukan oleh (Helmawati dan Ema, 2024) menggunakan model CNN dengan augmentasi data pada *dataset* BananaLSD dengan akurasi mencapai 92.85%. Penlitian oleh (Arifin,

2024) menggunakan model CNN dengan arsitektur GoogleNet dan mendapatkan akurasi sebesar 89.58%. Penelitian oleh (Pratama dkk, 2024) menggunakan CNN dengan VGG-19 dengan dataset bernama *Bangadbundu Sheikh Mu-jubur Rahman Agricultural University* yang mendapatkan akurasi sebesar 92%. Penelitian oleh (Srivastav dkk, 2024) menggunakan CNN dengan akurasi tertinggi sebesar 96.52% pada *epoch* ke 80. Penelitian oleh (Criollo dkk, 2020) menggunakan CNN tanpa teknik regularisasi dengan hasil akurasi terbaik sebesar 87.5%. Penelitian oleh (Lubis dan Alifia, 2023) menggunakan model *Faster Region-Convolutional Neural Network* (Faster R-CNN) yang mendapatkan akurasi sebesar 91.66%.

Terdapat penelitian yang menggunakan model ViT untuk mengidentifikasi penyakit pada daun pisang seperti pada penelitian oleh (Saini, 2024) dengan mendapatkan akurasi yang baik sebesar 96.88% walau didapatkan pada *epoch* 100. Penelitian oleh (Tiwari, 2024) menggunakan model variasi dari ViT yaitu ViT-B16 dengan hasil akurasi mencapai 96.88%. Penelitian oleh (Prashanthi dkk, 2024) menggunakan model ViT dengan variasi ViT-B32 yang mendapatkan akurasi sebesar 95.22%.

Terdapat penelitian yang menggabungkan beberapa metode untuk mengidentifikasi penyakit pada berbagai tanaman seperti pada penelitian (Tanwar dkk, 2023) yang menggunakan metode *hybrid* CNN-SVM (*Convolutional Neural Network-Support Vector Machine*) yang mendapatkan rata-rata akurasi sebesar 90%. Penelitian oleh (Banerjee dkk, 2023) juga menggunakan metode hybrid CNN-SVM yang mendapatkan rata-rata akurasi tertinggi sebesar 94%. Penelitian oleh (Kalim dkk, 2022) menggunakan metode *hybrid* CNN-RF (*Convolutional Neural*

Network-Random Forest) dan menggunakan arsitektur VGG16 pada CNN yang mendapatkan akurasi sebesar 87%. Penelitian oleh (David dkk, 2021) menggunakan metode *hybrid* CNN-RNN (*Conventional Neural Network-Recurrent Neural Network*) yang dimana mendapatkan akurasi sebesar 81.75% dengan 200 *epochs*. Penelitian oleh (Thakur dkk, 2023) menggunakan metode *hybrid* CNN-ViT (Convolutional Neural Network-Vision Transformer) pada *dataset* PlantVillage dan mendapatkan akurasi sebesar 98.86%. Penelitian oleh (Rehman dkk, 2024) menggunakan metode *hybrid* CNN-ViT pada *dataset* SARTAJ dan mendapatkan akurasi sebesar 98.1%.

Berdasarkan penelitian terdahulu, pendekatan CNN dalam mengidentifikasi penyakit pada daun pisang telah terbukti efektif dengan akurasi yang bervariasi mulai dari 89% sampai 96.52%. Selain itu, model ViT untuk mengidentifikasi penyakit pada daun pisang juga telah menunjukkan hasil akurasi yang cukup tinggi mulai dari 95.22% sampai 96.88%. Lebih lanjut, pendekatan metode *hybrid* untuk mengidentifikasi penyakit pada daun tanaman seperti menggunakan metode hybrid CNN-SVM, CNN-RF, CNN-RNN, dan CNN-ViT mendapatkan akurasi yang bervariasi mulai dari 81.75% sampai 98.86%, yang dimana metode *hybrid* yang menghasilkan akurasi tertinggi yaitu menggunakan metode *hybrid* CNN-ViT. Pemilihan ViT-B16 sebagai varian model ViT yang digunakan karena ViT-B16 merupakan varian model ViT *Base* yang tidak terlalu berat karena hanya mempunyai 86 juta parameter dibandingkan ViT *Large* yang mempunyai 307 juta parameter, ViT-B16 membagi *input* gambar menjadi 16x16 *patch* sehingga

mendapatkan pola fitur citra secara detail dan diharapkan mendapatkan akurasi yang tinggi dan presisi (Dosovitskiy dkk, 2021).

Berdasarkan dari performa masing-masing pendekatan, penggunaan model *hybrid* CNN-ViT menjadi menjanjikan untuk diimplementasikan dalam identifikasi penyakit pada daun pisang. Penggunaan metode *hybrid* CNN-ViT dalam identifikasi penyakit pada daun pisang, diharapkan dapat meningkatkan akurasi dibandingkan dengan hanya menggunakan metode tunggal saja.