

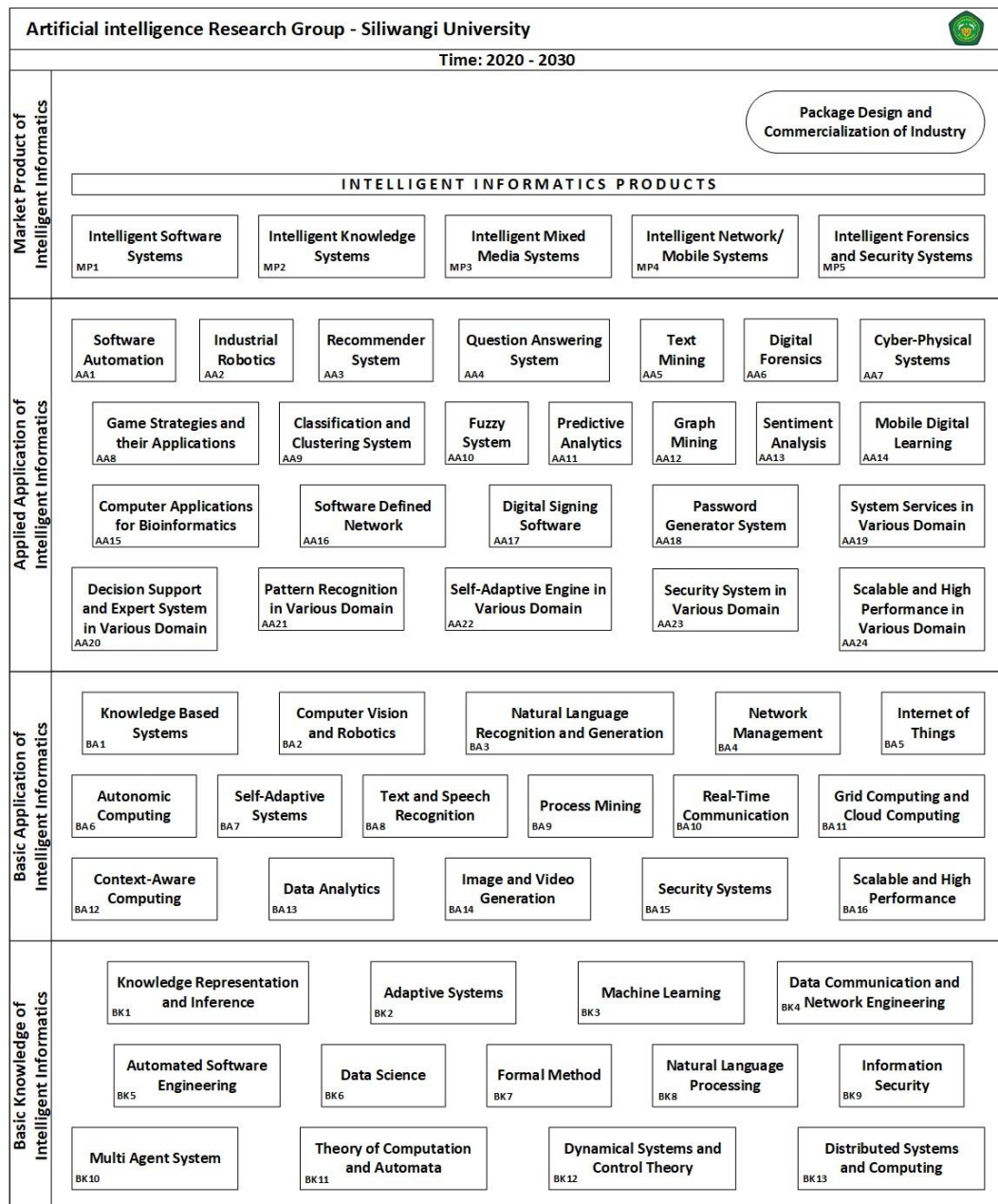
BAB III METODE PENELITIAN

Bab ini menjelaskan metodologi yang digunakan dalam penelitian, mulai dari persiapan dan anotasi dataset hingga pelatihan dan evaluasi model RF-DETR-Seg untuk segmentasi penyakit daun teh. Proses penelitian menggunakan pendekatan *Human-in-the-Loop* berbantuan *Segment Anything Model* (SAM) dalam pembentukan anotasi, serta membandingkan skenario pelatihan *baseline* dan augmentasi konservatif. Seluruh tahapan penelitian disusun secara sistematis untuk mengevaluasi performa model berdasarkan metrik yang telah ditentukan.

3.1 Roadmap Penelitian

Roadmap atau peta jalan penelitian yang digunakan pada penelitian ini menggunakan *roadmap* penelitian *Artificial Intelligence Research Group* Universitas Siliwangi (AIS) untuk rentang waktu 2020-2030 yang terdapat pada gambar 3.1.

Berdasarkan *Roadmap* AIS yang terdapat pada gambar 3.1, penelitian ini berfokus pada penerapan *Segment Anything Model* (SAM) dalam proses anotasi semi-otomatis untuk meningkatkan performa dan konsistensi anotasi agar model RF-DETR-Seg dapat menangkap konteks objek dalam segmentasi penyakit daun teh. Maka dari itu, penelitian ini berkaitan dengan *Computer Vision* (BA2) dan juga *Pattern Recognition in Various Domain* (AA21). Dalam pengembangan lebih lanjut, penelitian ini dapat berpotensi ke dalam ranah *Intelligent Software System* (MP1) melalui implementasi model ke dalam perangkat lunak cerdas untuk deteksi penyakit daun.



Gambar 3. 1 Roadmap Artificial Intelligence Research Group Universitas Siliwangi Tahun 2020-2030 (AIS, 2020)

3.2 Tahapan Penelitian



Gambar 3. 2 Tahapan Penelitian Sumber: Ilustrasi disusun oleh penulis berdasarkan tahapan penelitian dengan bantuan generative artificial intelligence

Penelitian ini menggunakan pendekatan *mixed methods* (metode campuran) dengan desain eksperimental untuk mengevaluasi performa RF-DETR-Seg dalam melakukan *instance segmentation* penyakit daun teh. Pendekatan *mixed methods* merupakan pendekatan penelitian yang menggabungkan metode kualitatif dan kuantitatif dalam satu penelitian sehingga dapat memberikan gambaran yang lebih menyeluruh terhadap permasalahan yang diteliti (Akinjide dkk., 2023). Pendekatan kualitatif diterapkan pada proses anotasi menggunakan *Human-in-the-Loop* (HITL), yaitu melalui proses validasi dan perbaikan hasil segmentasi yang dihasilkan oleh *Segment Anything Model* (SAM) serta interpretasi hasil inferensi model melalui analisis visual terhadap hasil segmentasi dan klasifikasi citra daun teh. Sementara itu, pendekatan kuantitatif diterapkan pada proses pelatihan, pengujian, dan evaluasi model menggunakan metrik performa berupa data numerik. Desain eksperimental dilakukan melalui dua skenario pelatihan, yaitu *baseline* tanpa augmentasi dan augmentasi konservatif. Kedua skenario tersebut kemudian diuji dan dibandingkan menggunakan prosedur evaluasi yang sama untuk mengetahui pengaruh augmentasi terhadap performa model. Evaluasi model *deep learning* dilakukan melalui eksperimen yang terstruktur dengan menggunakan metrik kuantitatif untuk mengukur dan membandingkan performa model secara objektif (Elshawy dkk., 2021). Tahapan penelitian disusun secara sistematis dengan mengadaptasi penelitian terdahulu dan disajikan pada Gambar 3.2 (Muhajirin dkk., 2024; Waruwu dkk., 2025).

3.2.1 Studi Literatur (*Literature Study*)

Pada tahapan ini, studi literatur dilakukan untuk membangun landasan teoritis dan memperjelas arah penelitian (Muhajirin dkk., 2024). Pada penelitian ini, literatur yang dikaji meliputi konsep penyakit pada daun teh, segmentasi gambar, *Segment Anything Model* (SAM), arsitektur *Transformer* seperti *Vision Transformer* dan RF-DETR, serta metrik evaluasi COCO dan metrik evaluasi segmentasi. Luaran yang dihasilkan disampaikan pada BAB 1 latar belakang penelitian dan BAB 2 landasan teori.

3.2.2 Perumusan Masalah (*Problem Formulation*)

Perumusan masalah yang dilakukan bertujuan untuk menentukan arah penelitian, ruang lingkup, serta desain penelitian yang akan digunakan (Waruwu dkk., 2025). Berdasarkan studi literatur, ditemukan beberapa tantangan utama, yaitu kompleksitas anotasi *pixel-level* yang memerlukan ketelitian tinggi, keterbatasan anotasi manual yang rentan terhadap subjektivitas, serta kebutuhan model segmentasi yang mampu merepresentasikan bentuk penyakit daun teh secara presisi dan efisien.

Untuk mengatasi permasalahan tersebut, penelitian ini tidak hanya berfokus pada evaluasi performa model RF-DETR-Seg, tetapi juga pada proses pembentukan *ground truth* melalui penggunaan model *Segment Anything Model* (SAM) pada *tools* X-AnyLabeling sebagai instrumen anotasi. Penggunaan SAM dan *tools* X-AnyLabeling digunakan untuk meningkatkan efisiensi serta konsistensi pelabelan *mask* segmentasi dibandingkan metode manual atau otomatis. Dengan demikian, perumusan masalah diarahkan pada bagaimana kualitas anotasi yang dihasilkan

menggunakan pendekatan tersebut dan mendukung proses pelatihan model RF-DETR-Seg yang diukur menggunakan metrik evaluasi COCO dan metrik evaluasi segmentasi.

3.2.3 Akuisisi dan Praproses Data (*Data Acquisition and Preprocessing*)

Tahap akuisisi data dilakukan untuk memperoleh dan menyiapkan data yang digunakan dalam penelitian. Pada tahap ini, penelitian menggunakan dataset *teaLeafBD* sebagai sumber data utama, kemudian dilakukan persiapan data, pembagian dataset ke dalam subset pelatihan, validasi, dan pengujian, serta anotasi gambar untuk menghasilkan ground truth segmentasi yang digunakan pada proses pemodelan.

3.2.3.1 Dataset

Dataset yang digunakan dalam penelitian ini adalah *teaLeafBD*, yang merupakan dataset *open-source* yang dipublikasikan oleh Alam dkk. (2025). Dataset ini terdiri dari 5278 gambar daun teh yang dikumpulkan dari 8 kebun teh di wilayah Bangladesh pada tahun 2024. Dataset ini telah memiliki label kelas untuk kebutuhan klasifikasi, namun belum menyediakan anotasi pada tingkat piksel yang dibutuhkan untuk kebutuhan segmentasi. Berikut merupakan ringkasan spesifikasi dataset *teaLeafBD* yang disajikan pada tabel 3.1

Tabel 3. 1 Ringkasan Spesifikasi Dataset (Alam dkk., 2025)

Parameter	Nilai
Kebun	8
Kelas	7
Total gambar	5278
Format gambar	JPG
Resolusi gambar	1200 x 1600 px

Berdasarkan tabel 3.1, dataset *teaLeafBD* memiliki resolusi gambar yang relatif tinggi sehingga dapat mendukung proses segmentasi, karena detail area penyakit dapat direpresentasikan secara lebih jelas pada tingkat piksel dan memudahkan dalam proses anotasi. Selain itu, dataset *teaLeafBD* telah terbagi menjadi 7 kelas, yaitu enam jenis penyakit daun teh dan satu kelas daun sehat. Berikut merupakan pembagian distribusi kelas pada dataset *teaLeafBD* yang disajikan pada tabel 3.2

Tabel 3. 2 Distribusi Kelas (Alam dkk., 2025)

No	Kelas	Jumlah Gambar
1	Tea algal leaf spot	418
2	Brown blight	508
3	Gray blight	1013
4	Helopeltis	607
5	Red spider	515
6	Green mirid bug	1282
7	Healthy	935
Total		5278

Berdasarkan karakteristik dan distribusi kelas yang dimiliki, *teaLeafBD* digunakan sebagai dataset sumber pada penelitian ini. Selanjutnya, dilakukan tahap persiapan data untuk menyusun dataset penelitian akhir dari *teaLeafBD*, termasuk pemilihan gambar yang digunakan pada masing-masing kelas sebelum proses pembagian data dan anotasi dilakukan.

3.2.3.2 Persiapan Data

Tahap persiapan data dilakukan untuk menyusun dataset akhir penelitian sebelum digunakan pada tahap anotasi dan pemodelan. Meskipun dataset *TeaLeafBD* menyediakan 5278 gambar daun teh, penelitian ini tidak menggunakan seluruh gambar tersebut secara langsung. Sebaliknya, data tersebut diseleksi kembali untuk menghasilkan dataset penelitian yang lebih terkendali, seimbang antar kelas dan realistis untuk anotasi tingkat piksel. Hal ini penting karena anotasi

segmentasi memerlukan penelusuran batas lesi dan verifikasi secara manual, sehingga beban waktu dan tenaga anotasi jauh lebih besar dibandingkan pelabelan klasifikasi tradisional. Dalam konteks *human-in-the-loop*, biaya anotasi tetap menjadi faktor penting dalam proses pembentukan dataset karena setiap *mask* yang dihasilkan masih perlu diverifikasi dan dikoreksi oleh anotator (CVHub520, 2023; Farook dkk., 2023; Mosqueira dkk., 2023). Ringkasan parameter persiapan data yang digunakan pada penelitian ini ditunjukkan pada Tabel 3.3.

Tabel 3. 3 Parameter Persiapan Data

Aspek	Nilai
Ratio	80:10:10
Gambar/Kelas	150
Total Dataset	1050
Random Seed	42

Pada penelitian ini, dataset akhir disusun dengan memilih 150 gambar untuk setiap kelas, sehingga total dataset penelitian berjumlah 1050 gambar. Pemilihan jumlah yang sama pada tiap kelas dilakukan untuk membentuk distribusi data yang seimbang, mengingat distribusi kelas pada *teaLeafBD* sumber bersifat tidak merata. Pendekatan ini digunakan untuk mengurangi potensi dominasi kelas mayoritas selama pelatihan serta memberikan representasi yang setara bagi setiap kelas pada proses evaluasi (W. Chen dkk., 2024).

Setelah dataset penelitian akhir terbentuk, data dibagi ke dalam subset *train*, *validation*, dan *test* dengan rasio 80:10:10. Rasio ini dipilih agar sebagian besar data tetap dialokasikan untuk pelatihan model, sementara subset validasi dan pengujian tetap tersedia untuk memantau proses pelatihan dan mengevaluasi performa model secara terpisah (Xu & Goodacre, 2018). Proses pembagian data dilakukan menggunakan *random seed* 42 untuk menjaga reproduktibilitas, sehingga

penyusunan subset data dapat diulang kembali secara konsisten pada kondisi eksperimen yang sama (Ahmed & Lofstead, 2022).

3.2.3.3 Proses Anotasi

Anotasi dilakukan menggunakan pendekatan *Human-in-the-Loop* (HITL) dengan memanfaatkan *Segment Anything Model* (SAM) melalui *software* X-AnyLabeling. Pada pendekatan ini, SAM digunakan sebagai alat bantu untuk menghasilkan *mask* segmentasi awal berdasarkan *prompt* yang diberikan oleh annotator, sedangkan keputusan akhir terhadap hasil anotasi tetap ditentukan oleh annotator melalui proses verifikasi dan koreksi manual. Pendekatan semacam ini dipilih karena anotasi segmentasi lesi penyakit daun memiliki karakteristik yang cukup menantang, seperti bentuk lesi yang tidak beraturan, ukuran lesi yang bervariasi, serta kemunculan bercak yang dapat terpisah atau saling berdekatan dalam satu daun. Dengan memanfaatkan *mask* awal dari model, proses anotasi dapat dilakukan secara lebih efisien dibandingkan anotasi manual penuh, namun tetap mempertahankan kontrol manusia terhadap kualitas anotasi akhir (Alam dkk., 2025; CVHub520, 2023; Kirillov dkk., 2023; Mosqueira dkk., 2023).

Dalam penelitian ini, peran annotator tidak hanya terbatas pada pemberian label kelas, tetapi juga mencakup penentuan area target, pemberian prompt, evaluasi hasil *mask*, serta koreksi anotasi apabila segmentasi awal yang dihasilkan SAM belum sesuai dengan area gejala yang ingin dilabeli. Agar hasil anotasi tetap konsisten antar gambar, proses anotasi pada penelitian ini menerapkan aturan anotasi yang mengacu pada prinsip pembuatan dataset segmentasi penyakit tanaman, khususnya

dalam penentuan objek anotasi, granularitas lesi, dan validasi hasil anotasi (Wei, 2024).

Berdasarkan pendekatan HITL yang digunakan, proses anotasi pada penelitian ini dilakukan secara bertahap mulai dari pemeriksaan gambar, penentuan pendekatan anotasi, pembuatan anotasi awal, evaluasi hasil segmentasi, hingga finalisasi anotasi. Rangkaian tahapan tersebut dirangkum dalam Algoritma 3.1.

Algoritma 3. 1 Human-in-the-Loop Annotation	
1	Input:
2	I : Citra daun teh
3	SAM : Segment Anything Model
4	MODE : Pendekatan Anotasi
5	An : Annotator
6	
7	Output:
8	GT : Ground Truth
9	
10	HITL-ANNOTATION(I)
11	// Image Inspection
12	CHECK-IMAGE(I)
13	
14	// Determine Annotation Strategy
15	MODE ← SELECT-ANNOTATION-MODE(I)
16	
17	// Initial Annotation
18	if MODE = SAM-GENERATED then
19	GT ← SAM-GENERATE(I)
20	else if MODE = HUMAN-SAM then
21	GT ← HUMAN-ANNOTATE(I, An)
22	GT ← SAM-REFINE(I, GT, An)
23	else
24	GT ← MANUAL-ANNOTATE(I, An)
25	end if
26	
27	// Annotation Evaluation
28	VALID ← EVALUATE-ANNOTATION(GT)
29	
30	while VALID = FALSE do
31	// Annotation Refinement
32	GT ← REFINE-ANNOTATION(I, GT, SAM)
33	
34	// Re-evaluation

35	VALID $\leftarrow$ EVALUATE-ANNOTATION (GT)
36	end while
37	
38	// Save Annotation
39	SAVE-ANNOTATION (GT)
40	
41	return GT

Berdasarkan Algoritma 3.1, proses anotasi diawali dengan memasukkan gambar hasil persiapan data ke dalam perangkat lunak anotasi untuk dilakukan pemeriksaan awal terhadap objek daun dan gejala penyakit yang tampak. Pemeriksaan ini bertujuan untuk menilai keterlihatan area target yang akan dianotasi serta menentukan pendekatan anotasi yang paling sesuai.

Setelah pendekatan anotasi ditentukan, annotator melakukan pembuatan anotasi awal sesuai kondisi gambar. Hasil anotasi kemudian dievaluasi untuk memastikan bahwa area gejala yang dilabeli telah sesuai dengan target anotasi. Apabila hasil anotasi belum sesuai, annotator melakukan penyempurnaan melalui koreksi manual hingga diperoleh anotasi yang representatif. Setelah hasil anotasi dinilai sesuai, anotasi disimpan sebagai *ground truth* segmentasi yang digunakan pada tahap pelatihan model (Oehme dkk., 2026).

Pada penelitian ini, pendekatan anotasi dibedakan ke dalam tiga mode utama yang disesuaikan dengan karakteristik citra dan kualitas hasil segmentasi awal. Ringkasan ketiga mode anotasi tersebut disajikan pada Tabel 3.4.

Tabel 3. 4 Mode Anotasi

Mode Anotasi	Kondisi
<i>SAM Generated</i> + Validasi	Digunakan ketika area lesi terlihat cukup jelas dan <i>mask</i> awal yang dihasilkan SAM sudah mendekati area target (Kirillov dkk., 2023; Mosqueira dkk., 2023).

<i>Human Primary + SAM Reference</i>	SAM masih mampu membantu menunjukkan lokasi gejala, tetapi batas lesi belum cukup akurat untuk langsung dijadikan anotasi akhir (Sofiiuk dkk., 2021).
<i>Fully Manual</i>	Digunakan ketika hasil segmentasi SAM tidak representatif, misalnya pada lesi yang sangat kecil, samar, atau memiliki batas yang sulit dipisahkan secara otomatis.

Proses anotasi pada penelitian ini menerapkan tiga mode yang dipilih sesuai karakteristik lesi dan kualitas hasil segmentasi awal. Mode anotasi yang berbeda tersebut memungkinkan annotator menyesuaikan tingkat keterlibatan SAM dan koreksi manual pada setiap gambar, mulai dari validasi *mask* awal hingga anotasi manual penuh pada kasus yang lebih kompleks. Setelah pendekatan anotasi ditentukan, proses anotasi selanjutnya mengikuti aturan anotasi yang digunakan dalam penelitian ini, khususnya terkait penentuan objek anotasi, penanganan lesi yang terpisah atau menyatu, serta anotasi pada kelas *healthy*.

Setelah pendekatan anotasi ditentukan, proses anotasi pada penelitian ini selanjutnya mengikuti aturan anotasi untuk menjaga konsistensi pelabelan antar gambar. Aturan tersebut mencakup penentuan objek anotasi pada masing-masing kelas, penanganan lesi yang terpisah atau menyatu, serta penentuan anotasi pada lesi berukuran kecil atau samar. Ringkasan aturan anotasi yang diterapkan pada penelitian ini disajikan pada Tabel 3.5.

Tabel 3. 5 Aturan Anotasi (Wei, 2024)

Aspek	Aturan Anotasi
Kelas Penyakit	Anotasi dilakukan pada area lesi atau gejala penyakit yang tampak pada permukaan daun, bukan pada seluruh daun.
Kelas <i>Healthy</i>	Anotasi dilakukan pada seluruh area daun sebagai objek segmentasi karena tidak terdapat lesi yang dapat dijadikan target anotasi.

Aspek	Aturan Anotasi
Lesi terpisah	Lesi yang terpisah secara visual dan memiliki batas yang masih dapat dibedakan dianotasi sebagai polygon terpisah.
Lesi berdekatan atau menyatu	Lesi yang berdekatan atau memiliki batas yang tidak jelas dapat dianotasi sebagai satu area gabungan mengikuti kontur luar area terinfeksi.
Lesi kecil atau samar	Lesi tetap dianotasi apabila gejala masih dapat dikenali dan batas area target masih dapat ditentukan secara visual. Untuk memastikan keterlihatan objek setelah penyesuaian resolusi model, anotasi diperiksa menggunakan <i>preview</i> 432×432 piksel. Lesi yang tidak lagi dapat dibedakan secara visual pada resolusi tersebut tidak dijadikan target anotasi terpisah.
Area Sehat disekitar lesi	Polygon diusahakan mengikuti area gejala secara rapat dan tidak memasukkan area sehat yang tidak termasuk bagian dari lesi.

Aturan anotasi yang disajikan pada tabel 3.5 difokuskan pada konsistensi penentuan objek segmentasi sesuai karakteristik masing-masing kelas. Pada kelas penyakit, anotasi diarahkan pada area lesi sebagai target utama segmentasi, sedangkan pada kelas *healthy* anotasi dilakukan pada seluruh daun. Selain itu, aturan mengenai pemisahan atau penggabungan lesi serta penanganan lesi kecil diterapkan agar anotasi tetap representatif terhadap gejala yang tampak pada gambar, sekaligus selaras dengan kebutuhan pelatihan model segmentasi. Pendekatan ini sejalan dengan prinsip pembangunan dataset segmentasi penyakit tanaman yang menekankan ketepatan pelabelan area gejala serta konsistensi anotasi antar gambar (Wei, 2024).

3.2.4 Augmentasi (*Augmentation*)

Augmentasi data merupakan teknik untuk meningkatkan keragaman data latih melalui transformasi gambar tanpa mengubah label dasarnya, sehingga model dapat belajar dari variasi orientasi, posisi, maupun kondisi visual objek dan diharapkan memiliki kemampuan generalisasi yang lebih baik serta lebih tahan terhadap *overfitting*. Dalam konteks deteksi dan segmentasi penyakit tanaman, augmentasi juga umum digunakan untuk membantu model mengenali gejala dengan ukuran, bentuk, dan distribusi visual yang beragam pada gambar lapangan, terutama ketika data pelatihan terbatas atau memiliki variasi kondisi akuisisi yang tinggi (Diallo dkk., 2025). Pada penelitian ini, jumlah data yang digunakan dibatasi sebanyak 150 citra per kelas, sehingga augmentasi digunakan untuk meningkatkan keragaman data latih tanpa menambah data baru secara manual. Penerapannya juga digunakan sebagai skenario eksperimen untuk menguji apakah augmentasi konservatif dapat memberikan pengaruh terhadap kemampuan generalisasi RF-DETR-Seg dibandingkan model *baseline* tanpa augmentasi (Yang dkk., 2020). Augmentasi hanya diterapkan pada subset *training*, sedangkan subset *validation* dan *test* tetap menggunakan citra asli agar evaluasi dilakukan pada data yang tidak mengalami transformasi augmentasi.

Tabel 3. 6 Parameter Augmentasi

Kategori	Augmentasi	Parameter
<i>Geometric</i>	<i>HorizontalFlip</i>	P = 0.5
	<i>Rotate</i>	Limit = 15 ⁰ , p = 0.4
	<i>ShiftScaleRotate</i>	Shift_limit = 0.05, scale_limit = 0.1, rotate_limit = 0, p = 0.3

Kategori	Augmentasi	Parameter
<i>Photometric</i>	<i>RandomBrightnessContrast</i>	Brightness_limit = 0.2, contast_limit = 0.2, p = 0.5
	<i>HueSaturationValue</i>	Hue_shift_limit = 20, sat_shift_limit = 30, val_shift_limit = 20, p = 0.5
<i>Noise / Blur</i>	<i>GaussianBlur</i>	Blur_limit = 3-5, p = 0.2

Berdasarkan Tabel 3.6, penelitian ini menerapkan augmentasi konservatif yang mencakup transformasi geometrik, fotometrik, serta *blur* ringan pada subset *training*. Transformasi geometrik berupa *HorizontalFlip*, *Rotate*, dan *ShiftScaleRotate* digunakan untuk mensimulasikan variasi orientasi, posisi, dan skala daun yang masih realistis pada proses akuisisi gambar di lapangan. Pemilihan transformasi tersebut mengacu pada penelitian sebelumnya pada gambar penyakit daun yang menggunakan *horizontal flip*, rotasi ringan, dan transformasi affine untuk meningkatkan keragaman data latih tanpa mengubah struktur utama objek secara berlebihan. Selanjutnya, transformasi fotometrik berupa *RandomBrightnessContrast* dan *HueSaturationValue* diterapkan untuk mengakomodasi variasi pencahayaan, kontras, dan warna daun, sejalan dengan penggunaan *color jitter* serta augmentasi berbasis hue–saturation pada penelitian deteksi penyakit tanaman sebelumnya. Sementara itu, *GaussianBlur* dengan intensitas ringan digunakan untuk mensimulasikan sedikit ketidakjelasan gambar yang masih mungkin dijumpai pada pengambilan gambar lapangan, mengingat gambar penyakit tanaman di lingkungan nyata dapat mengalami degradasi kualitas berupa *blur*, noise, dan pencahayaan yang tidak seragam. Dengan demikian, pemilihan parameter augmentasi pada penelitian ini ditujukan untuk menambah

variasi data latih secara terkendali agar model lebih robust terhadap variasi visual gambar penyakit daun teh, namun tetap mempertahankan karakteristik lesi sebagai target segmentasi (Duhan dkk., 2026; Mia dkk., 2026).

3.2.5 Pemodelan (*Modelling*)

Pemodelan pada penelitian ini menggunakan RF-DETR-Seg Medium sebagai model utama untuk melakukan deteksi dan segmentasi penyakit daun teh. Tahapan pemodelan meliputi implementasi Vision Transformer sebagai backbone ekstraksi fitur, implementasi RF-DETR-Seg untuk proses prediksi objek dan segmentasi, serta konfigurasi parameter pelatihan yang digunakan selama proses pembelajaran model.

3.2.5.1 DINOv2 *Vision Transformer*

Vision Transformer pada penelitian ini digunakan sebagai *backbone* RF-DETR-Seg Medium untuk mengekstraksi representasi fitur dari citra daun teh. Implementasi *Vision Transformer* mengikuti arsitektur DINOv2 yang telah dijelaskan pada Bab II. Tahapan transformasi citra menjadi representasi fitur yang digunakan sebagai masukan bagi RF-DETR-Seg dirangkum pada Algoritma 3.2.

Algoritma 3. 2 DINOv2 <i>Vision Transformer</i>	
1	Input:
2	I : Input gambar tensor (432 x 432 x 3)
3	D : Dimensi embedding (768)
4	P_size : Ukuran patch (12 x 12)
5	L_enc : Jumlah lapisan encoder (12)
6	d_k : Dimensi dari setiap attention
7	head(64)
8	H : Jumlah Attention Head (12)
9	
10	Output:
11	Z_L : Urutan token keluaran Vision
12	Transformer
13	

14	Notasi:
15	X : Urutan patch embedding
16	PE : Positional Embedding
17	CLS : Token klasifikasi
18	REG : Token register
19	Z : Urutan representasi fitur
20	Q : Matriks Query
21	K : Matriks Key
22	V : Matriks Value
23	S : Matriks skor attention
24	Head : Output dari setiap attention head
25	A : Output Multi-Head Attention
26	F : Output Feed-Forward Network (FFN)
27	P : Patch hasil pembagian citra
28	Z_norm : Representasi fitur yang telah
29	dinormalisasi
30	W_Q : Matriks bobot proyeksi Query
31	W_K : Matriks bobot proyeksi Key
32	W_V : Matriks bobot proyeksi Value
33	W_O : Matriks bobot output Projection
34	
35	VISION-TRANSFORMER(I)
36	// Patch Partition
37	$P \leftarrow \text{SPLIT-INTO-PATCHES}(I, P_size)$
38	$// P \in \mathbb{R}^{(1296 \times 12 \times 12 \times 3)}$
39	
40	// Flatten
41	$P \leftarrow \text{FLATTEN}(P)$
42	$// P \in \mathbb{R}^{(1296 \times 432)}$
43	
44	// Patch Embedding
45	$X \leftarrow \text{EMBEDDING-PATCH}(P)$
46	$X \leftarrow PW_E + b_E$
47	$// X \in \mathbb{R}^{(1296 \times 768)}$
48	
49	// Add CLS Token
50	$CLS \leftarrow \text{INITIALIZE-CLS-TOKEN}()$
51	$X \leftarrow \text{JOIN}(CLS, X)$
52	$// X \in \mathbb{R}^{(1297 \times 768)}$
53	
54	// Add Register Tokens
55	$REG \leftarrow \text{INITIALIZE-REGISTER-TOKENS}(4)$
56	$X \leftarrow \text{JOIN}(X, REG)$
57	$// X \in \mathbb{R}^{(1301 \times 768)}$
58	
59	// Add Positional Embedding
50	

```

61 PE ← POSITIONAL-EMBEDDING(X)
62 Z0 ← X + PE
63 // Z0 ∈ ℝ(1301 × 768)
64
65 // Transformer Encoder
66 for l ← 1 to Lenc do
67     Znorm ← LAYER-NORM(Z(l-1))
68
69     // Linear Projection
70     Q ← Znorm WQ
71     K ← Znorm WK
72     V ← Znorm WV
73     // Q, K, V ∈ ℝ(1301 × 768)
74
75     // Split into Multiple Heads
76     (Q1, ..., QH) ← SPLIT(Q)
77     (K1, ..., KH) ← SPLIT(K)
78     (V1, ..., VH) ← SPLIT(V)
79     // Each head ∈ ℝ(1301 × 64)
80
81     // Multi-Head Self-Attention
82     for h ← 1 to H do
83         if WINDOW-BLOCK(l) then
84             Headh ← WINDOW-ATTENTION(Qh, Kh, Vh)
85         else
86             Sh ← (Qh KhT) / √dk
87             Headh ← SOFTMAX(Sh) Vh
88         end if
89     end for
90
91     // Concatenate Heads
92     A ← CONCAT(Head1, ..., HeadH)
93     // A ∈ ℝ(1301 × 768)
94
95     // Output Projection
96     A ← A WO
97     // A ∈ ℝ(1301 × 768)
98
99     // Residual Connection
100    Z' ← Z(l-1) + A
101    // Z' ∈ ℝ(1301 × 768)
102
103    // Feed-Forward Network
104    F ← FFN(LAYER-NORM(Z'))
105    // F ∈ ℝ(1301 × 768)
106
107

```

108	// Residual Connection
109	$Z_L \leftarrow Z' + F$
110	// $Z_L \in \mathbb{R}^{(1301 \times 768)}$
111	end for
112	
113	return Z_L

Berdasarkan Algoritma 3.2, proses Vision Transformer diawali dengan membagi citra masukan menjadi patch berukuran tetap yang kemudian diubah menjadi representasi vektor melalui proses *patch embedding*. Selanjutnya, *CLS Token*, *Register Token*, dan *positional embedding* ditambahkan untuk memberikan informasi posisi serta token tambahan yang digunakan selama proses pembelajaran. Representasi tersebut kemudian diproses secara bertahap oleh setiap lapisan *Transformer Encoder* melalui mekanisme *Multi-Head Self-Attention*, *Feed-Forward Network*, dan *Residual Connection* sehingga menghasilkan *output token sequence* yang merepresentasikan informasi visual citra. Output token sequence tersebut selanjutnya digunakan sebagai masukan bagi RF-DETR-Seg pada proses deteksi dan segmentasi (Oquab dkk., 2024; Zhai dkk., 2021).

3.2.5.2 RF-DETR-Seg

RF-DETR-Seg pada penelitian ini digunakan untuk melakukan deteksi dan segmentasi penyakit daun teh berdasarkan representasi fitur yang dihasilkan Vision Transformer. Implementasi RF-DETR-Seg mengikuti arsitektur yang telah dijelaskan pada Bab II, sedangkan tahapan implementasinya dirangkum pada Algoritma 3.3.

Algoritma 3.3 RF-DETR-Seg	
1	Input:
2	Z_L : Output token sequence dari Vision
3	Transformer
4	

5	L_dec	: Jumlah lapisan Transformer Decoder
6		(6)
7	H	: Jumlah Attention Head (8)
8	d_k	: Dimensi setiap attention head (32)
9		
10	Output:	
11	BOX	: Prediksi bounding box (cx, cy, w, h)
12	CLASS	: Prediksi kelas objek
13	MASK	: Mask logits prediksi segmentasi
14		
15	Notasi:	
16	F_seq	: Token setelah penghapusan CLS dan
17		Register
18	F_maps	: Peta fitur spasial
19	F_proj	: Multi-scale feature maps
20		{P3, P4, P5, P6}
21	F_up	: Feature map hasil Bilinear Upsampling
22	Q_emb	: Learnable Query Embeddings
23	Q	: Matriks Query
24	K	: Matriks Key
25	V	: Matriks Value
26	S	: Matriks skor attention
27	Head	: Output attention head
28	A	: Output Multi-Head Attention
29	F	: Output Feed-Forward Network
30	F_sp	: Feature spasial hasil penyempurnaan
31	Q_mask	: Feature Query untuk segmentasi
32	Q_norm	: Feature Query hasil Layer
33		Normalization
34	ΔP	: Sampling Offset
35	W_Q	: Bobot proyeksi Query
36	W_K	: Bobot proyeksi Key
37	W_V	: Bobot proyeksi Value
38	W_O	: Bobot output projection
39		
40	RF-DETR(Z_L)	
41	// Remove Special Tokens	
42	$F_{seq} \leftarrow \text{REMOVE-CLS-AND-REGISTER}(Z_L)$	
43	$// F_{seq} \in \mathbb{R}^{(1296 \times 768)}$	
44		
45	// Generate Spatial Feature Map	
46	$F_{maps} \leftarrow \text{RESHAPE}(F_{seq})$	
47	$// F_{maps} \in \mathbb{R}^{(36 \times 36 \times 768)}$	
48		
49	// Multi-Scale Feature Projection	
50	$F_{proj} \leftarrow \text{MULTISCALE-PROJECTOR}(F_{maps})$	
51	$// F_{proj} = \{P3, P4, P5, P6\}$	

```

52 //  $P_i \in \mathbb{R}^{(H_i \times W_i \times 256)}$ 
53
54 // Initialize Learnable Queries
55  $Q_{\text{emb}} \leftarrow \text{INITIALIZE-QUERY}()$ 
56 //  $Q_{\text{emb}} \in \mathbb{R}^{(300 \times 256)}$ 
57
58 // Transformer Decoder
59 for  $l \leftarrow 1$  to  $L_{\text{dec}}$  do
60      $Q_{\text{norm}} \leftarrow \text{LAYER-NORM}(Q_{\text{emb}})$ 
61
62     // Linear Projection
63      $Q \leftarrow Q_{\text{norm}} W_Q$ 
64      $K \leftarrow Q_{\text{norm}} W_K$ 
65      $V \leftarrow Q_{\text{norm}} W_V$ 
66     //  $Q, K, V \in \mathbb{R}^{(300 \times 256)}$ 
67
68     // Split into Multiple Heads
69      $(Q_1, \dots, Q_H) \leftarrow \text{SPLIT}(Q)$ 
70      $(K_1, \dots, K_H) \leftarrow \text{SPLIT}(K)$ 
71      $(V_1, \dots, V_H) \leftarrow \text{SPLIT}(V)$ 
72     // Each Head  $\in \mathbb{R}^{(300 \times 32)}$ 
73
74     // Multi-Head Self Attention
75     for  $h \leftarrow 1$  to  $H$  do
76          $S_h \leftarrow (Q_h K_h^T) / \sqrt{d_k}$ 
77          $\text{Head}_h \leftarrow \text{SOFTMAX}(S_h) V_h$ 
78     end for
79
80     // Concatenate Heads
81      $A \leftarrow \text{CONCAT}(\text{Head}_1, \dots, \text{Head}_H)$ 
82     //  $A \in \mathbb{R}^{(300 \times 256)}$ 
83
84     // Output Projection
85      $A \leftarrow A W_O$ 
86     //  $A \in \mathbb{R}^{(300 \times 256)}$ 
87
88     // Residual Connection
89      $Q' \leftarrow Q_{\text{emb}} + A$ 
90     //  $Q' \in \mathbb{R}^{(300 \times 256)}$ 
91
92     // Multi-Scale Deformable Cross Attention
93      $\Delta P \leftarrow \text{LINEAR}(\text{LAYER-NORM}(Q'))$ 
94
95      $A \leftarrow \text{MS-DEFORMABLE-ATTENTION}(\text{LAYER-NORM}(Q'),$ 
96      $F_{\text{proj}}, \Delta P)$ 
97     //  $A \in \mathbb{R}^{(300 \times 256)}$ 
98

```

```

99
100     // Residual Connection
101      $Q'' \leftarrow Q' + A$ 
102      $// Q'' \in \mathbb{R}^{(300 \times 256)}$ 
103
104     // Feed-Forward Network
105      $F \leftarrow \text{FFN}(\text{LAYER-NORM}(Q''))$ 
106      $// F \in \mathbb{R}^{(300 \times 256)}$ 
107
108     // Residual Connection
109      $Q_{\text{emb}} \leftarrow Q'' + F$ 
110      $// Q_{\text{emb}} \in \mathbb{R}^{(300 \times 256)}$ 
111 end for
112
113 // Detection Head
114  $\text{BOX} \leftarrow \text{BOX-HEAD}(Q_{\text{emb}})$ 
115  $// \text{BOX} = (cx, cy, w, h)$ 
116  $// \text{BOX} \in \mathbb{R}^{(300 \times 4)}$ 
117
118 // Classification Head
119  $\text{CLASS} \leftarrow \text{CLASS-HEAD}(Q_{\text{emb}})$ 
120  $// \text{CLASS} \in \mathbb{R}^{(300 \times C)}$ 
121
122 // Bilinear Upsampling
123  $F_{\text{up}} \leftarrow \text{BILINEAR-UPSAMPLE}(F_{\text{proj}})$ 
124  $// F_{\text{up}} \in \mathbb{R}^{(H_i \times W_i \times 256)}$ 
125
126 // Spatial Feature Refinement
127  $F_{\text{sp}} \leftarrow \text{DEPTHWISE-CONV-BLOCK}(F_{\text{up}})$ 
128  $// F_{\text{sp}} \in \mathbb{R}^{(H_i \times W_i \times 256)}$ 
129
130 // Spatial Projection
131  $F_{\text{sp}} \leftarrow \text{CONV-1}\times\text{1}(F_{\text{sp}})$ 
132  $// F_{\text{sp}} \in \mathbb{R}^{(H_i \times W_i \times 256)}$ 
133
134 // Query Feature Refinement
135  $Q_{\text{mask}} \leftarrow \text{MLP}(Q_{\text{emb}})$ 
136  $// Q_{\text{mask}} \in \mathbb{R}^{(300 \times 256)}$ 
137
138 // Query Projection
139  $Q_{\text{mask}} \leftarrow \text{QUERY-PROJECTION}(Q_{\text{mask}})$ 
140  $// Q_{\text{mask}} \in \mathbb{R}^{(300 \times 256)}$ 
141
142 // Mask Prediction
143  $\text{MASK} \leftarrow \text{MASK-PREDICTION}(F_{\text{sp}}, Q_{\text{mask}})$ 
144  $// \text{MASK} = \text{Mask Logits}$ 
145

```

146	// MASK $\in \mathbb{R}^{(300 \times H \times W)}$
147	
148	return BOX, CLASS, MASK

Berdasarkan Algoritma 3.3, *output token sequence* yang dihasilkan oleh Vision Transformer terlebih dahulu diproses dengan menghapus *CLS Token* dan *Register Token*, kemudian diubah menjadi *feature map* spasial melalui proses *reshaping* dan *multi-scale projection*. Selanjutnya, *feature map* diproses oleh *Transformer Decoder* menggunakan mekanisme *Multi-Head Self-Attention*, *Multi-Scale Deformable Cross Attention*, dan *Feed-Forward Network* untuk menghasilkan representasi query yang telah diperkaya dengan informasi spasial. Representasi tersebut kemudian digunakan oleh *Detection Head* untuk memprediksi *bounding box* dan kelas objek, serta oleh *Segmentation Head* untuk menghasilkan *mask logits* sebagai prediksi segmentasi penyakit daun teh (Robinson dkk., 2025).

3.2.5.3 Pelatihan RF-DETR-Seg

Pelatihan model pada penelitian ini dilakukan menggunakan dua skenario, yaitu model *baseline* tanpa augmentasi dan model dengan augmentasi konservatif. Kedua skenario menggunakan konfigurasi parameter pelatihan yang sama sehingga perbedaan performa yang diperoleh hanya dipengaruhi oleh penggunaan augmentasi pada data pelatihan. Ringkasan parameter pelatihan yang digunakan dalam penelitian ini disajikan pada Tabel 3.7.

Tabel 3. 7 Parameter Pelatihan (Robinson dkk., 2025)

Parameter	Nilai
Model	RF-DETR-Seg Medium
<i>Input Resolution</i>	432
Max. epoch	100
<i>Batch size</i>	2
<i>Gradient accumulation</i>	8
<i>Learning rate</i>	1e-4

Parameter	Nilai
<i>Encoder learning rate</i>	1.5e-4
<i>Weight decay</i>	1e-4
<i>Scheduler</i>	Cosine
<i>Early stopping patience</i>	15
<i>Random seed</i>	42

Berdasarkan Tabel 3.7, pelatihan model pada penelitian ini menggunakan konfigurasi yang dirancang untuk menjaga stabilitas proses optimisasi. Pengaturan *learning rate* dan *encoder learning rate* digunakan untuk mengontrol pembaruan parameter selama proses *fine-tuning*, sedangkan *weight decay* berfungsi sebagai regularisasi untuk membantu mengurangi risiko *overfitting* (Loshchilov & Hutter, 2019; Robinson dkk., 2025). Selain itu, *cosine scheduler* digunakan untuk menyesuaikan laju pembelajaran secara bertahap dan *early stopping* dengan *patience* 15 digunakan untuk menghentikan pelatihan ketika performa validasi tidak lagi menunjukkan peningkatan. *Random seed* ditetapkan sebesar 42 untuk mengontrol proses yang melibatkan keacakan sehingga eksperimen dapat dilakukan secara lebih konsisten dan mendukung reproduksibilitas. Pengendalian random seed penting karena perbedaan proses acak selama pelatihan dapat menghasilkan variasi pada model dan performa yang diperoleh (Ahmed & Lofstead, 2022). Berikut merupakan pipeline dari alur pelatihan RF-DETR-Seg yang disajikan pada algoritma 3.4.

Algoritma 3. 4 Pipeline Pelatihan RF-DETR-Seg	
1	Input:
2	D : Dataset COCO Segmentation
3	Mpre : Model RF-DETR-Seg
4	CFG : Konfigurasi pelatihan
5	
6	Output:
7	RESULT : Hasil evaluasi model
8	Mbest : Model Terbaik

```

9
10 RF-DETR-SEG TRAINING (D,Mpre,CFG)
11
12 // Dataset Validation
13 CHECK-DATASET-STRUCTURE (D)
14 VALIDATE-COCO-ANNOTATION (D)
15 GENERATE-DATASET-SUMMARY (D)
16 CALCULATE-IMAGE-DISTRIBUTION (D)
17 CALCULATE-ANNOTATION-DISTRIBUTION (D)
18 VISUALIZE-DISTRIBUTION (D)
19 PREVIEW-ANNOTATION (D)
20
21 // Environment Setup
22 IMPORT-LIBRARIES ()
23 INITIALIZE-ENVIRONMENT ()
24 SELECT-COMPUTING-DEVICE ()
25
26 // Training Configuration
27 SET-RANDOM-SEED (CFG)
28 INITIALIZE-OUTPUT-DIRECTORY ()
29 SAVE-TRAINING-CONFIGURATION (CFG)
30
31 // Model Initialization
32 MODEL ← LOAD-PRETRAINED-MODEL (Mpre)
33 DISPLAY-MODEL-SUMMARY (MODEL)
34
35 // Model Training
36 MODEL.TRAIN (
37     TRAIN = D.train,
38     VALID = D.valid,
39     CONFIG = CFG
40 )
41
42 // Training Verification
43 DISPLAY-MODEL-COMPLEXITY (MODEL)
44 VERIFY-TRAINING-OUTPUT ()
45
46 // Best Model Selection
47 Mbest ← LOAD-BEST-CHECKPOINT ()
48
49 // Model Inference
50 PREDICTION ← RUN-INFERENCE (Mbest,D.test)
51
52 // Model Evaluation
53 COCO_RESULT ← COMPUTE-COCO-METRICS (PREDICTION)
54 CUSTOM_RESULT ← COMPUTE-CUSTOMMETRICS (PREDICTION)
55 GENERATE-CONFUSION-MATRIX (PREDICTION)

```

56	
57	// Visualization
58	VISUALIZE-PREDICTION (PREDICTION)
59	VISUALIZE-BINARY-MASK (PREDICTION)
60	
61	RESULT $\leftarrow$ COMBINE (COCO_RESULT, CUSTOM_RESULT)
62	
63	return RESULT

3.2.6 Evaluasi (*Evaluation*)

Tahap evaluasi pada penelitian ini bertujuan untuk menilai kemampuan model RF-DETR-Seg dalam mendeteksi sekaligus mensegmentasi penyakit daun teh pada data *testing*. Evaluasi dilakukan menggunakan dua kelompok metrik. Metrik COCO digunakan untuk menggambarkan performa segmentasi model secara umum melalui AP@50–95, AP@50, AP@75, AP Small, AP Medium, AP Large, dan AR@100 (Wood, 2022). Selanjutnya, *custom metrics* berupa *precision*, *recall*, F1-score, *mean IoU*, dan *mean Dice* digunakan untuk memberikan penilaian yang lebih rinci terhadap hasil segmentasi (Egi dkk., 2026). Dalam evaluasi *custom metrics*, prediksi model diseleksi menggunakan *confidence threshold* sebesar 0,25 dan pencocokan antara *mask* prediksi dengan *ground truth* dilakukan menggunakan ambang $\text{IoU} \geq 0,5$. *Confidence threshold* digunakan untuk menyaring prediksi berdasarkan skor kepercayaan model sebelum proses evaluasi. Menurut Wenkel dkk. (2021), pemilihan *confidence threshold* merupakan *operating point* yang memengaruhi keseimbangan antara *false positive* dan *false negative*, sehingga tidak terdapat satu nilai baku yang berlaku untuk seluruh model maupun dataset (Wenkel dkk., 2021). Sementara itu, penggunaan AP/mAP pada berbagai tingkat IoU umum diterapkan untuk mengevaluasi performa model, sedangkan IoU dan *Dice*

digunakan untuk mengukur tingkat kesesuaian hasil segmentasi (Schlosser dkk., 2024; Terven dkk., 2025).

3.2.7 Dokumentasi (*Documentation*)

Tahap dokumentasi mencakup penyusunan laporan dan penyajian hasil secara terstruktur sebagai bentuk pertanggungjawaban ilmiah (Muhajirin dkk., 2024). Dokumentasi dalam penelitian ini meliputi laporan tugas akhir dan publikasi ilmiah dengan target jurnal terindeks nasional atau internasional.