

CEK TURNITIN

SKRIPSI Dewi Nawangsari SI

 SATSET DIGITAL 1

Document Details

Submission ID

trn:oid:::3618:150983691

Submission Date

Sep 8, 2026, 3:27 PM GMT+7

Download Date

Sep 8, 2026, 3:32 PM GMT+7

File Name

SKRIPSI Dewi Nawangsari SI.pdf

File Size

2.7 MB

90 Pages

14,798 Words

97,984 Characters

40% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

Filtered from the Report

- Bibliography

Top Sources

- 26%  Internet sources
- 14%  Publications
- 34%  Submitted works (Student Papers)

Integrity Flags

0 Integrity Flags for Review

Our system's algorithms look deeply at a document for any inconsistencies that would set it apart from a normal submission. If we notice something strange, we flag it for you to review.

A Flag is not necessarily an indicator of a problem. However, we'd recommend you focus your attention there for further review.

Top Sources

- 26% Internet sources
- 14% Publications
- 34% Submitted works (Student Papers)

Top Sources

The sources with the highest number of matches within the submission. Overlapping sources will not be displayed.

1	Internet	repository.unisi.ac.id	8%
2	Internet	jutif.if.unsoed.ac.id	1%
3	Internet	eprints3.upgris.ac.id	<1%
4	Student papers	Universitas Pendidikan Indonesia on 2026-07-05	<1%
5	Student papers	UPN Veteran Yogyakarta on 2026-06-23	<1%
6	Student papers	Universitas Islam Riau on 2026-06-19	<1%
7	Student papers	UPN Veteran Yogyakarta on 2026-07-02	<1%
8	Internet	doaj.org	<1%
9	Student papers	UM Surabaya on 2026-07-21	<1%
10	Internet	jurnal.ugp.ac.id	<1%
11	Internet	ojs.ninetyjournal.com	<1%

12	Internet	repositori.uma.ac.id	<1%
13	Student papers	Politeknik Manufaktur Negeri Bangka Belitung on 2026-07-27	<1%
14	Student papers	UPN Veteran Yogyakarta on 2026-06-26	<1%
15	Internet	ejurnal.stmik-budidarma.ac.id	<1%
16	Internet	proceeding.unpkediri.ac.id	<1%
17	Student papers	Institut Pertanian Bogor on 2026-08-27	<1%
18	Student papers	Universitas Islam Indonesia on 2026-06-15	<1%
19	Internet	publikasi.hawari.id	<1%
20	Student papers	UPN Veteran Yogyakarta on 2026-05-25	<1%
21	Internet	ejournal.itn.ac.id	<1%
22	Publication	Haiping Li, Xingzhen Tao. "Enhanced YOLOv8-Based Real-Time Navel Orange Det..."	<1%
23	Publication	Stevanus Januar Latuluma, Magdalena A. Ineke Pakereng. "DETEKSI JUMLAH KEN..."	<1%
24	Student papers	Universitas Bina Sarana Informatika on 2026-06-11	<1%
25	Student papers	Universitas Jember on 2026-01-12	<1%

26	Internet	ojs.unida.info	<1%
27	Student papers	UPN Veteran Yogyakarta on 2026-06-02	<1%
28	Internet	repository.nusaputra.ac.id	<1%
29	Student papers	Universitas Islam Riau on 2026-07-29	<1%
30	Student papers	Universitas Pancasila on 2025-08-07	<1%
31	Internet	www.mdpi.com	<1%
32	Student papers	Sekolah Teknik Elektro & Informatika on 2026-08-03	<1%
33	Student papers	Politeknik Manufaktur Negeri Bangka Belitung on 2025-11-06	<1%
34	Student papers	UPN Veteran Yogyakarta on 2026-06-18	<1%
35	Student papers	Universitas Khairun on 2026-05-11	<1%
36	Internet	library.poltekkes-surabaya.ac.id	<1%
37	Student papers	UPN Veteran Yogyakarta on 2026-04-22	<1%
38	Student papers	UPN Veteran Yogyakarta on 2026-08-27	<1%
39	Publication	Virganda Rimba Asmara, Abdi Pandu Kusuma, Wahyu Dwi Puspitasari. "Penerapa...	<1%

40	Internet	digilibadmin.unismuh.ac.id	<1%
41	Internet	www.jurnalteknik.unisla.ac.id	<1%
42	Student papers	UPN Veteran Yogyakarta on 2026-01-06	<1%
43	Internet	digilib.uinsa.ac.id	<1%
44	Internet	farida_a.staff.gunadarma.ac.id	<1%
45	Student papers	Universitas Sangga Buana YPKP on 2026-07-28	<1%
46	Student papers	Universitas Pertahanan RI on 2026-06-19	<1%
47	Internet	repository.pnj.ac.id	<1%
48	Internet	repository.unhas.ac.id	<1%
49	Student papers	East Union High School on 2026-06-30	<1%
50	Student papers	UPN Veteran Yogyakarta on 2026-04-23	<1%
51	Student papers	Universitas Islam Riau on 2026-08-21	<1%
52	Internet	123dok.com	<1%
53	Student papers	Universitas Muhammadiyah Makassar on 2026-05-11	<1%

54	Student papers	Syiah Kuala University on 2025-07-17	<1%
55	Student papers	UPN Veteran Yogyakarta on 2026-07-29	<1%
56	Internet	eprints.polsri.ac.id	<1%
57	Student papers	UPN Veteran Yogyakarta on 2026-05-05	<1%
58	Student papers	Universitas Jember on 2026-06-21	<1%
59	Internet	repository.its.ac.id	<1%
60	Internet	repository.um-surabaya.ac.id	<1%
61	Internet	www.frontiersin.org	<1%
62	Student papers	Universitas Islam Riau on 2025-12-15	<1%
63	Student papers	Universitas Pamulang on 2026-06-06	<1%
64	Internet	repository.uin-suska.ac.id	<1%
65	Student papers	Sekolah Teknik Elektro & Informatika on 2026-07-17	<1%
66	Internet	digilib.unila.ac.id	<1%
67	Student papers	Universitas Islam Malang on 2025-07-08	<1%

68	Student papers	Universitas Pamulang on 2026-07-28	<1%
69	Student papers	Universitas Pertamina on 2026-08-10	<1%
70	Student papers	Universitas Sangga Buana YPKP on 2026-07-21	<1%
71	Internet	ejournal2.unud.ac.id	<1%
72	Internet	repository.um-palembang.ac.id	<1%
73	Internet	scholar.unand.ac.id	<1%
74	Student papers	Politeknik Negeri Jember on 2026-08-05	<1%
75	Publication	Reimon Aldin. "Implementasi Convolutional Neural Network Pada Deteksi Peny...	<1%
76	Student papers	Universitas Islam Riau on 2025-09-02	<1%
77	Student papers	Universitas Islam Riau on 2026-06-10	<1%
78	Student papers	Universitas Sultan Ageng Tirtayasa on 2025-04-08	<1%
79	Internet	ejournal.uksw.edu	<1%
80	Internet	journal.cib.institute	<1%
81	Internet	repository.polibatam.ac.id	<1%

82	Internet	repository.unpas.ac.id	<1%
83	Internet	repository.unsri.ac.id	<1%
84	Student papers	College of Banking and Financial Studies on 2026-03-04	<1%
85	Publication	Derza Andreas, Shafiq Najwan, Muhammad Fajar Raihan, Maria Erviana Asinta Le...	<1%
86	Student papers	Fakultas Teknik on 2025-07-23	<1%
87	Student papers	Universitas Khairun on 2026-07-13	<1%
88	Student papers	Universitas Sangga Buana YPKP on 2026-07-23	<1%
89	Student papers	Universitas Sangga Buana YPKP on 2026-07-24	<1%
90	Student papers	Universitas Sangga Buana YPKP on 2026-07-31	<1%
91	Student papers	Universitas Sangga Buana YPKP on 2026-08-03	<1%
92	Internet	kc.umh.ac.id	<1%
93	Internet	repository.unissula.ac.id	<1%
94	Internet	repository.upnvj.ac.id	<1%
95	Publication	Dina Ayuni. "PEMANFAATAN METODE CANNY EDGE DETECTION UNTUK PENGENA...	<1%

96	Student papers	Universitas Islam Negeri Sumatera Utara on 2026-06-18	<1%
97	Student papers	Universitas Maritim Raja Ali Haji on 2025-01-12	<1%
98	Student papers	Universitas Muhammadiyah Palembang on 2025-07-18	<1%
99	Student papers	Universitas Pamulang on 2026-06-22	<1%
100	Internet	repository.ulb.ac.id	<1%
101	Student papers	FAKULTAS ILMU KOMPUTER on 2026-07-09	<1%
102	Student papers	Politeknik Negeri Jember on 2026-08-05	<1%
103	Student papers	UPN Veteran Yogyakarta on 2026-05-04	<1%
104	Student papers	Universitas Muhammadiyah Purwokerto on 2025-07-14	<1%
105	Student papers	Universitas Pamulang on 2026-06-17	<1%
106	Internet	repo.bunghatta.ac.id	<1%
107	Internet	repository.ub.ac.id	<1%
108	Student papers	Politeknik Penerbangan Surabaya on 2026-06-23	<1%
109	Student papers	Universitas Putera Batam on 2026-07-25	<1%

110	Student papers	Universitas Tarumanagara on 2025-12-12	<1%
111	Internet	repository.widyatama.ac.id	<1%
112	Student papers	UNIVERSITAS BUDI LUHUR on 2026-02-06	<1%
113	Student papers	Universitas Jember on 2026-07-13	<1%
114	Student papers	Universitas Sumatera Utara on 2026-03-09	<1%
115	Internet	repository.pertanian.go.id	<1%
116	Internet	www.ejournal.itn.ac.id	<1%
117	Publication	Ririn Anugerah Ikasatya, Cahya Apriliani, Fathir Jannatul Firdaus, Gede Yogi Prata...	<1%
118	Student papers	Telkom University on 2026-06-20	<1%
119	Student papers	Telkom University on 2026-06-30	<1%
120	Student papers	UPN Veteran Yogyakarta on 2026-06-26	<1%
121	Student papers	Universitas Islam Riau on 2025-07-17	<1%
122	Student papers	Universitas Muhammadiyah Purwokerto on 2025-08-15	<1%
123	Student papers	Fakultas Teknik on 2025-08-21	<1%

124	Publication	M. Aufaa Rahman, Benny Sukma Negara, Muhammad Irsyad, Lestari Handayani, ...	<1%
125	Student papers	Universitas Bengkulu on 2024-09-27	<1%
126	Student papers	Universitas Islam Riau on 2025-12-04	<1%
127	Student papers	Universitas Merdeka Malang on 2026-05-19	<1%
128	Student papers	Universitas Muhammadiyah Purwokerto on 2025-07-10	<1%
129	Student papers	Universitas Sangga Buana YPKP on 2026-07-29	<1%
130	Internet	geometa.co.uk	<1%
131	Internet	journal.artei.or.id	<1%
132	Internet	jurnal.wicida.ac.id	<1%
133	Publication	Charladinna Arvianne Reyvalda. "Analisis Kinerja YOLOv11 untuk Deteksi Makana...	<1%
134	Student papers	Institut Pertanian Bogor on 2026-01-19	<1%
135	Student papers	Institut Teknologi Sumatera on 2025-09-22	<1%
136	Student papers	Konsorsium 2 Perguruan Tinggi Swasta Indonesia 2022 on 2023-05-31	<1%
137	Publication	Muhammad Sidiq Pramono, Aditya Permana Wibowo. "PENERAPAN CONVOLUTIO...	<1%

138	Student papers	Perpustakaan on 2025-08-06	<1%
139	Student papers	Sekolah Teknik Elektro & Informatika on 2025-08-13	<1%
140	Student papers	Sriwijaya University on 2019-11-18	<1%
141	Student papers	Telkom University on 2026-07-21	<1%
142	Student papers	Universitas Brawijaya on 2017-01-27	<1%
143	Student papers	Universitas Brawijaya on 2017-07-24	<1%
144	Student papers	Universitas Islam Riau on 2026-04-30	<1%
145	Student papers	Universitas Islam Riau on 2026-06-18	<1%
146	Student papers	Universitas Islam Riau on 2026-08-27	<1%
147	Student papers	Universitas Pamulang on 2026-05-04	<1%
148	Student papers	Universitas Pancasila on 2026-09-02	<1%
149	Student papers	Universitas Pertahanan RI on 2026-07-02	<1%
150	Student papers	Universitas Putera Batam on 2018-11-19	<1%
151	Publication	Zafira A'idah Gunawan, Mauladi Fadhillah, Anargya Rabbani Aslam, Mayanda M...	<1%

152	Internet	adoc.pub	<1%
153	Internet	journal.trunojoyo.ac.id	<1%
154	Internet	rama.kemdiktisaintek.go.id	<1%
155	Publication	Arga Satria Ramadhan, Yunianita Rahmawati, Ika ratna Indra Astutik, Sumarno. ...	<1%
156	Publication	Daniel Daniel, Dedy Hermanto. "Implementasi Arsitektur YOLOv11 untuk Deteksi...	<1%
157	Student papers	Gyeongsang National University on 2020-07-13	<1%
158	Publication	Indah Clara Sari. "INTEGRASI MODEL DEEP LEARNING EFFICIENTNET-B0 UNTUK D...	<1%
159	Publication	Rahandya Cahyo, Alda Cendekia Siregar, Barry Ceasar Octariadi. "Implementasi A...	<1%
160	Student papers	Telkom University on 2026-01-19	<1%
161	Student papers	UNIVERSITAS BUDI LUHUR on 2025-07-24	<1%
162	Student papers	Universitas Bengkulu on 2023-11-06	<1%
163	Student papers	Universitas Islam Riau on 2026-05-13	<1%
164	Student papers	Universitas Islam Riau on 2026-07-21	<1%
165	Student papers	Universitas Mulawarman on 2021-01-04	<1%

166	Student papers	Universitas Pertahanan RI on 2026-06-18	<1%
167	Student papers	Universitas Tarumanagara on 2025-12-13	<1%
168	Student papers	University of Wollongong on 2023-02-16	<1%
169	Publication	Wildan Amru Hidayat, Vinna Rahmayanti Setyaning Nastiti. "PERBANDINGAN KIN...	<1%
170	Internet	journal.aira.or.id	<1%
171	Internet	journal.ugm.ac.id	<1%
172	Internet	journal.untar.ac.id	<1%
173	Internet	jurnal.risetilmiah.ac.id	<1%
174	Internet	repository.uinsaizu.ac.id	<1%
175	Internet	www.scilit.net	<1%
176	Publication	Benni Agung Nugroho, Afta Ramadhan Zayn, Ellya Nurfarida, Riswan Eko Wahyu ...	<1%
177	Publication	Doms Upuy. "Perbandingan Kinerja Model YOLOv8n dan YOLOv8s pada Deteksi O...	<1%
178	Publication	Jenny Rahma Hidayat, Jemakmum. "Implementasi Klasifikasi Citra Berbasis Tenso...	<1%
179	Publication	Rizqy Supriyadi, Muhamad Irfan, Rizki Dwi Hapijar, Fadil Abubakar et al. "PENGE...	<1%

180	Publication	Sri Damayanti, Husna Sari, Bobby Ardiansyah, Dicky Apdillah. "Analisis Sentimen ..."	<1%
181	Student papers	Telkom University on 2026-01-23	<1%
182	Student papers	UIN Raden Intan Lampung on 2025-11-25	<1%
183	Student papers	UIN Raden Intan Lampung on 2025-11-30	<1%
184	Student papers	UIN Sunan Ampel Surabaya on 2026-05-27	<1%
185	Student papers	UM Surabaya on 2025-07-16	<1%
186	Student papers	UNIVERSITAS BUDI LUHUR on 2025-08-19	<1%
187	Student papers	Universidad Tecnológica Centroamericana UNITEC on 2026-08-06	<1%
188	Student papers	Universitas Brawijaya on 2018-07-19	<1%
189	Student papers	Universitas Brawijaya on 2020-04-28	<1%
190	Student papers	Universitas Dian Nuswantoro on 2026-01-09	<1%
191	Student papers	Universitas Diponegoro on 2019-02-13	<1%
192	Student papers	Universitas Islam Riau on 2026-04-30	<1%
193	Student papers	Universitas Muhammadiyah Purwokerto on 2025-05-20	<1%

194	Student papers	Universitas Muhammadiyah Surakarta on 2015-01-17	<1%
195	Student papers	Universitas Pancasila on 2025-08-10	<1%
196	Student papers	Universitas Sangga Buana YPKP on 2026-07-28	<1%
197	Student papers	Universitas Sangga Buana YPKP on 2026-08-14	<1%
198	Internet	conference.ut.ac.id	<1%
199	Internet	ejournal.unama.ac.id	<1%
200	Internet	eprints.umm.ac.id	<1%
201	Internet	id.123dok.com	<1%
202	Internet	repository.teknokrat.ac.id	<1%
203	Publication	Andreas Rezeki Zai, Bambang Suhardi, Surya Tri Nowo, Rika Rosnelly, Adil Setiaw...	<1%
204	Publication	Hong Deng, Yiyi Chen, Yilu Xu. "ALD-YOLO: a lightweight attention detection mod...	<1%
205	Student papers	Perpustakaan on 2025-08-06	<1%
206	Publication	Radit Widiyanto, Hanindya Aisyah, Ardiansyah Ardiansyah. "Implementasi Hand G...	<1%
207	Student papers	Universitas Islam Riau on 2026-07-08	<1%

208	Student papers	Universitas Pamulang on 2026-07-25	<1%
209	Student papers	Universitas Putera Batam on 2026-07-28	<1%
210	Student papers	University of Keele on 2025-04-17	<1%
211	Student papers	Admin Turnitin UIIS on 2025-07-31	<1%
212	Publication	Andreas Saputra, Dedy Hermanto. "Deteksi Penyakit Daun Teh Berdasarkan Citra..."	<1%
213	Publication	Firmansyah Nur Hidayah, Yunianita Rahmawati, Yulian Findawati, Nuril Lutvi Aziz...	<1%
214	Publication	Muhammad Reza Pahlevi, Imam Tahyudin, Ades Tikaningsih. "AUTOMATED EDIBI..."	<1%
215	Publication	Rizal Rudiansyah, Asril Adi Sunarto, Didik Indrayana. "IMPLEMENTASI ALGORITM..."	<1%
216	Publication	Sukhpreet Kaur, Amanpreet Kaur, Manish Kumar. "Recent Advances in Computati..."	<1%
217	Student papers	UIN Sjech M.Djamil Djambek Bukittinggi on 2026-06-15	<1%
218	Student papers	UM Surabaya on 2026-07-21	<1%
219	Student papers	UPN Veteran Yogyakarta on 2026-05-04	<1%
220	Student papers	Universitas Budi Luhur on 2025-07-10	<1%
221	Student papers	Universitas Pamulang on 2026-07-11	<1%

222	Student papers	Universitas Putera Batam on 2026-07-25	<1%
223	Student papers	Universitas Respati Indonesia on 2021-10-10	<1%
224	Student papers	Universitas Sangga Buana YPKP on 2026-07-20	<1%
225	Student papers	Universitas Sangga Buana YPKP on 2026-07-22	<1%
226	Student papers	Universitas Sumatera Utara on 2025-05-23	<1%
227	Internet	eprints.umpo.ac.id	<1%

33

**SISTEM DETEKSI TANAMAN PENYAKIT DAUN MELON
MENGUNAKAN YOU ONLY LOOK ONCE (YOLOV8)
DAN CONVOLUTIONAL BLOCK ATTENTION
MODULE (CBAM)
TUGAS AKHIR**



DEWI NAWANGSARI

403221010061

1

**PROGRAM STUDI SISTEM INFORMASI
FAKULTAS TEKNIK DAN ILMU KOMPUTER
UNIVERSITAS ISLAM INDRAGIRI
TEMBILAHAN**

2026

**SISTEM DETEKSI TANAMAN PENYAKIT DAUN MELON
MENGUNAKAN YOU ONLY LOOK ONCE (YOLOV8)
DAN CONVOLUTIONAL BLOCK ATTENTION
MODULE (CBAM)
TUGAS AKHIR**

*Diajukan Sebagai Salah Satu Syarat Untuk Mendapatkan Gelar Sarjana Sistem
Informasi Stratal (SI) Pada Fakultas Teknik dan Ilmu Komputer Universitas Islam
Indragiri*



Disusun oleh:

DEWI NAWANGSARI

403221010061

**PROGRAM STUDI SISTEM INFORMASI
FAKULTAS TEKNIK DAN ILMU KOMPUTER
UNIVERSITAS ISLAM INDRAGIRI
TEMBILAHAN
2026**

PERNYATAAN

Saya yang bertanda tangan di bawah ini menyatakan bahwa, skripsi ini merupakan karya saya sendiri (ASLI), dan isi dalam skripsi ini tidak terdapat karya yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu institusi Pendidikan tinggi manapun, dan sepanjang pengetahuan saya juga tidak terdapat karya atau pendapat yang pernah ditulis dan/atau diterbitkan oleh orang lain, kecuali yang secara tertulis diacu dalam naskah ini dan disebutkan dalam daftar pustaka.

Segala sesuatu terkait dengan naskah dan karya yang telah dibuat menjadi tanggung jawab saya pribadi.

Tembilahan, Juli 2026

Dewi Nawangsari

Nim. 403221010061

KATA PENGANTAR

Puji syukur penulis panjatkan ke hadirat Allah SWT atas limpahan rahmat dan karunia-Nya, sehingga penulis dapat menyelesaikan penyusunan Proposal Tugas Akhir ini dengan baik. Karya tulis ini disusun sebagai salah satu hasil penelitian dan merupakan syarat untuk memperoleh gelar Sarjana Strata Satu (S1) pada Universitas Islam Indragiri. Dalam proses penyusunan proposal ini, penulis banyak menerima bimbingan, arahan, serta dukungan dari berbagai pihak. Oleh karena itu, dengan segala kerendahan hati penulis menyampaikan rasa terima kasih yang sebesar-besarnya kepada:

1. Terimakasih yang tak terhingga kepada Ibu dan Alm. Bapak selaku orang tua penulis, atas doa, kasih sayang, dan menjadi motivasi penulis untuk menyelesaikan skripsi dengan tepat waktu.
2. Bapak Dr. H. Najamuddin, Lc., M.A, selaku Rektor Universitas Islam Indragiri.
3. Bapak Bayu Riyanto, S.SI., M.Kom, selaku Dekan Fakultas Teknik dan Ilmu Komputer.
4. Ibu Fitri Yunita, S.SI., M.Kom, selaku Ketua Program Studi Sistem Informasi.
5. Bapak Muhammad Jibril, S.Kom., M.MSI selaku dosen pembimbing utama yang telah membimbing dalam penyusunan proposal ini.
6. Bapak Usman, ST., M.Kom selaku dosen pembimbing pendamping yang memberikan bimbingan dalam penyusunan ini.
7. Seluruh dosen Program Studi Sistem Informasi Universitas Islam Indragiri yang telah membekali penulis dengan ilmu dan wawasan selama masa perkuliahan.

56 8. Terima kasih kepada Nenek dan Kakek saya, yang selalu mendoakan, memberikan semangat dan nasehat untuk penulis.

60 9. Kepada adik penulis, yang senantiasa menjadi pelengkap dinamika dalam hidup penulis. Kehadiranmu adalah motivasi tersendiri dalam setiap langkah perjuangan ini, termasuk dalam menyelesaikan skripsi ini.

9. Terima kasih juga untuk sahabat saya Elvi suciati dan Siti nursehati yang selalu menemani penulis dalam keadaan susah, senang dan selalu membantu penulis saat kesusahan dari awal perantauan sampai sekarang.

10. Terima kasih untuk teman-teman saya, Jamiya ranika, Diana Ruvita, Febrina sari, Nur Ainah, yang selalu bikin suasana selalu rame dan bahagia.

67 11. Kepada seseorang yang tak kalah penting kehadirannya, yaitu dengan Nim 403221010076 terimakasih telah menjadi bagian dari perjalanan hidup penulis, dan menjadi salah satu penyemangat karena selalu ada dalam keadaan suka maupun duka.

1 Penulis menyadari bahwa Tugas Akhir ini masih jauh dari sempurna, baik dari segi isi maupun penyajiannya. Oleh karena itu, penulis sangat mengharapkan kritik dan saran yang membangun untuk perbaikan di masa yang akan datang. Akhir kata, semoga Tugas Akhir ini dapat memberikan manfaat bagi pembaca dan menjadi referensi bagi penelitian selanjutnya, khususnya dalam bidang sistem cerdas dan deteksi penyakit pada tanaman.

1 13 Desember 2025

Penulis

DAFTAR ISI

HALAMAN JUDUL

PERSETUJUAN

PENGESAHAN

PERNYATAAN

KATA PENGANTAR i

DAFTAR ISI iii

DAFTAR GAMBAR vi

INTISARI ix

ABSTRACT x

BAB 1 PENDAHULUAN 1

1.1 Latar Belakang 1

1.2 Rumusan Masalah 4

1.3 Batasan Masalah 4

1.4 Tujuan Penelitian 5

1.5 Manfaat Penelitian 6

1.6 Sistematika Penelitian 6

BAB II TINJAUAN LITERATUR 8

2.1 Tanaman Melon 8

2.2 Tanaman Hidroponik 8

2.3 *You Only Look Once (YOLO)* 9

2.4 *Convolutional Block Attention Module (CBAM)* 9

2.5 Penelitian Terdahulu 10

2.5 Rangkuman 17

BAB III METODE PENELITIAN 19

1	3.1 Kerangka Penelitian.....	19
	3.1.1 Tinjauan Literatur	20
	3.1.2 Analisis Pra-penelitian	20
	3.1.3 Pengumpulan Data	21
	3.1.4 Pra-processing data	21
	3.1.5 Pelatihan Model	22
	3.1.6 Analisis Sistem	23
	3.1.7 Perancangan Sistem	24
	3.1.8 Implementasi Sistem	29
	3.1.9 Pengujian Sistem.....	30
	3.1.10 Kesimpulan Penelitian.....	31
	BAB VI HASIL DAN PEMBAHASAN	32
1	4.1 Hasil.....	32
	4.1.1 Pengumpulan Data	32
	4.1.2 Pra-processing Data	33
	4.1.3 Pengolahan Data	38
	4.1.4 Pengembangan Model Arsitektur YOLOv8n dan YOLOv8n+CBAM.....	42
42	4.1.5 Pelatihan dan Evaluasi Model	48
	4.1.6 Implementasi Sistem	60
	4.1.7 Pengujian Sistem.....	65
	4.2 Pembahasan.....	69
27	4.2.1 Pembahasan Hasil Pelatihan YOLOv8+CBAM	69
	4.2.2 Pembahasan Hasil Evaluasi Model	69
	4.2.3 Pembahasan Confusion Matrik	70
	4.2.4 Pembahasan Implementasi Model Pada Android	71
	5.1 Kesimpulan	73

106

5.2 Saran	74
DAFTAR PUSTAKA	75
LAMPIRAN	77

DAFTAR GAMBAR

Gambar 2.1 Proses You Only Look Once (YOLO)	9
Gambar 2.2 Proses Convolutional Block Attention Module (CBAM).....	10
Gambar 3.1 Kerangka Penelitian.....	20
Gambar 3.2 Data training penyakit tanaman melon bercak daun bersudut, mosaik virus, layu fusarium, dan embun bulu.....	22
Gambar 3.3 Tampilan Awal Aplikasi	25
Gambar 3.4 Halaman Petunjuk Aplikasi.....	25
Gambar 3.5 Tampilan Halaman Deteksi.....	26
Gambar 3.6 Tampilan Detail Penyakit.....	27
Gambar 3.7 Diagram Use Case Sistem Deteksi Penyakit Daun Melon	28
Gambar 3.8 Diagram Aktivitas.....	29
Gambar 4.1 Proses Labeling Dataset	35
Gambar 4.2 Hasil Pelabelan	40
Gambar 4.3 Arsitektur YOLOv8	44
Gambar 4.4 Arsitektur Yolo+CBAM	47
Gambar 4.5 Grafik Training, Validation Box Loss, Precision, Recall dan F1 Score.	50
Gambar 4.6 Confusion Matrik.....	52
Gambar 4.7 Perhitungan Confusio Matrik	52
Gambar 4.8 F1-Confidence Curve.....	53
Gambar 4.9 Precision-Confidence Curve	54
Gambar 4.10 Recall-Confidence Curve	55
Gambar 4.11 Precision-Recall Curve	56
Gambar 4.12 Hasil Deteksi Bercak Bersudut.....	59

108

Gambar 4.13 Hasil Deteksi Embun Bulu 59

Gambar 4.14 Hasil Deteksi Layu Fusarium 60

Gambar 4.15 Hasil Deteksi Mosaik Virus 60

1

Gambar 4.16 Alur Integrasi Model YOLO ke Aplikasi Android 61

Gambar 4.17 Tampilan Awal Aplikasi Android 62

143

Gambar 4.18 Petunjuk Penggunaan Aplikasi Android 62

Gambar 4.19 Halaman Utama Aplikasi Android 63

Gambar 4.20 Tampilan Halaman Informasi Penyakit 65

DAFTAR TABEL

Tabel 2.1 Penelitian Terdahulu.....	10
Tabel 3.1 Spesifikasi Perangkat Keras (Hardware)	23
Tabel 3.2 Perangkat Lunak (Software) yang digunakan.....	23
Tabel 4.1 Pemberian class_id	39
Tabel 4.2 Distribusi Dataset Gambar Penyakit Daun Melon	41
Tabel 4.3 Parameter Pelatihan Model YOLOv8 + CBAM	49
Tabel 4.4 Perbandingan Kinerja Model	57
Tabel 4.5 Perbandingan Nilai mAP@0.5 per Kelas Penyakit.....	58
Tabel 4. 6 Black Box Testing Aplikasi Deteksi Penyakit Daun Melon.....	66
Tabel 4. 7 Hasil Prediksi Model pada Data Uji.....	67

INTISARI

Penyakit pada daun melon dapat menurunkan kualitas dan produktivitas hasil panen apa bila tidak dideteksi secara dini. Keterbatasan pengetahuan petani dalam mengenali gejala visual penyakit sering menyebabkan keterlambatan penanganan, sehingga diperlukan sistem deteksi otomatis berbasis citra digital. Penelitian ini bertujuan mengetahui kinerja model YOLOv8 yang dikombinasikan dengan Convolutional Block Attention Module (CBAM) dalam mendeteksi empat jenis penyakit daun melon, yaitu bercak bersudut, embun bulu, layu fusarium, dan mosaik virus, serta mengimplementasikannya ke dalam aplikasi Android dan membandingkan kinerjanya dengan model You Only Look Once (YOLOv7). Penelitian menggunakan pendekatan kuantitatif dengan dataset 4.000 citra daun melon yang melalui tahap pra pemrosesan, pelabelan, dan augmentasi data, kemudian dilatih menggunakan arsitektur YOLOv8n yang dimodifikasi dengan penyisipan modul CBAM pada bagian backbone. Model terbaik dikonversi ke format TensorFlow Lite dan diintegrasikan ke aplikasi Android. Hasil pengujian menunjukkan model YOLOv8-CBAM memperoleh precision 82,92%, recall 77,32%, mAP@0.5 82,00%, Matrik 83%, sementara YOLOv7 memperoleh Precision sebesar 57,30%, Recall 49,20%, F1-score 52,93%, mAP@0.5 sebesar 52,20%, dan mAP@0.5:0.95 sebesar 21,50%. Hasil perbandingan menunjukkan bahwa model YOLOv8+CBAM memiliki performa deteksi yang lebih baik dibandingkan YOLOv7 pada seluruh metrik evaluasi. Ole karena itu, model YOLOv8+CBAM dinilai lebih sesuai diimplementasikan pada aplikasi android.

Kata kunci: You Only Look Once (YOLOv8), Convolutional Block Attention Module (CBAM), Deteksi Penyakit Daun Melon, Android, YOLOv7

ABSTRACT

Diseases affecting melon leaves can reduce crop quality and productivity if they are not detected at an early stage. Farmers' limited ability to recognize the visual symptoms of leaf diseases often leads to delayed treatment, highlighting the need for an automatic detection system based on digital image processing. This study aims to evaluate the performance of the YOLOv8 model integrated with the Convolutional Block Attention Module (CBAM) in detecting four types of melon leaf diseases, namely angular leaf spot, downy mildew, Fusarium wilt, and mosaic virus, as well as to implement the model into an Android application and compare its performance with the You Only Look Once version 7 (YOLOv7) model. This study employed a quantitative approach using a dataset of 4,000 melon leaf images, which underwent preprocessing, annotation, and data augmentation. The dataset was then trained using a modified YOLOv8n architecture with the CBAM module inserted into the backbone network. The best-performing model was converted into the TensorFlow Lite format and integrated into an Android application. The experimental results showed that the YOLOv8+CBAM model achieved a precision of 82.92%, recall of 77.32%, mAP@0.5 of 82.00%, and an overall metric of 83%, while the YOLOv7 model achieved a precision of 57.30%, recall of 49.20%, F1-score of 52.93%, mAP@0.5 of 52.20%, and mAP@0.5:0.95 of 21.50%. The comparison results indicate that the YOLOv8+CBAM model outperformed YOLOv7 across all evaluation metrics.

Keywords: You Only Look Once (YOLOv8), Convolutional Block Attention Module (CBAM), Melon Leaf Disease Detection, Android, YOLOv7

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Tanaman melon merupakan salah satu komoditas hortikultura yang memiliki nilai ekonomi tinggi dan berperan penting dalam sektor pertanian di Indonesia. Melon memiliki keunggulan berupa masa tanam yang relatif singkat serta kemampuan berproduksi sepanjang tahun, sehingga permintaan pasar terhadap komoditas ini terus meningkat. Seiring dengan meningkatnya produksi melon, tantangan dalam proses budidaya juga semakin kompleks, terutama terkait serangan penyakit pada tanaman yang dapat menurunkan kualitas dan kuantitas hasil panen. Penyakit pada tanaman melon umumnya menyerang bagian daun dan menyebabkan gangguan fotosintesis, yang berdampak langsung terhadap pertumbuhan dan produktivitas tanaman. Oleh karena itu, upaya deteksi penyakit secara dini dan akurat menjadi kebutuhan penting dalam mendukung keberhasilan budidaya melon secara berkelanjutan[1].

Dalam proses budidaya, daun tanaman melon merupakan bagian yang paling rentan terhadap serangan penyakit karena berperan langsung dalam proses fotosintesis dan pertumbuhan tanaman. Gangguan pada daun, seperti perubahan warna, munculnya bercak, hingga gejala layu, dapat menurunkan kemampuan tanaman dalam menyerap cahaya dan menghasilkan fotosintat secara optimal. Penelitian yang dilakukan menunjukkan bahwa serangan penyakit pada tanaman melon masih sering ditemukan dan berdampak signifikan terhadap kondisi tanaman secara keseluruhan. Meskipun penelitian tersebut mengamati penyakit secara umum pada tanaman melon, gejala awal infeksi umumnya dapat diidentifikasi

30 melalui perubahan visual pada daun, seperti daun menguning, layu, dan mengering. Kondisi ini menunjukkan bahwa daun dapat dijadikan indikator utama dalam mendeteksi keberadaan penyakit pada tanaman melon secara dini. Oleh karena itu, deteksi penyakit berbasis citra daun menjadi aspek penting untuk mendukung upaya pengendalian penyakit secara lebih cepat dan tepat guna meminimalkan penurunan produktivitas tanaman melon[2].

188 Daun tanaman melon juga menjadi bagian yang paling mudah diamati dalam proses identifikasi penyakit karena gejala serangan patogen umumnya pertama kali muncul pada permukaan daun. Penelitian yang dilakukan menunjukkan bahwa berbagai jenis penyakit pada tanaman melon, seperti layu fusarium, embun bulu, bercak bersudut dan mosaik virus, dapat dikenali melalui perubahan visual pada daun, antara lain berupa daun mengering, munculnya lapisan putih seperti bubuk, serta perubahan bentuk daun menjadi mengeriting[3]. Gejala-gejala tersebut mencerminkan respons fisiologis tanaman terhadap serangan patogen yang mengganggu proses penyerapan air dan fotosintesis. Namun, di tingkat lapangan, keterbatasan pengetahuan petani dalam mengenali perbedaan gejala visual pada daun sering menyebabkan keterlambatan atau kesalahan dalam pengendalian penyakit. Kondisi ini menegaskan bahwa identifikasi penyakit berbasis pengamatan daun memerlukan pendekatan yang lebih sistematis dan objektif agar deteksi penyakit pada tanaman melon dapat dilakukan secara lebih akurat dan konsisten[4].

75 Perkembangan teknologi kecerdasan buatan, khususnya di bidang Deep Learning, telah mendorong penggunaan metode deteksi objek berbasis citra seperti You Only Look Once (YOLO) dalam berbagai bidang termasuk pertanian. Metode ini mampu mendeteksi objek secara cepat dan akurat dalam satu kali proses

146

14

84 pemindaian (*single shot detection*). Versi *YOLOv8*, memiliki peningkatan signifikan dalam akurasi, kecepatan, dan efisiensi dibanding versi sebelumnya. Kemampuannya dalam mengenali pola visual yang kompleks menjadikan *YOLOv8* sangat potensial untuk digunakan dalam sistem deteksi penyakit daun melon berbasis citra digital, sebagaimana terbukti pada penelitian deteksi penyakit daun tomat yang mencapai akurasi hingga 96,6.

36 Meskipun metode *YOLOv8* memiliki kemampuan yang tinggi dalam
24 mendeteksi objek secara cepat dan akurat, model ini masih memiliki keterbatasan
24 dalam mendeteksi gejala penyakit daun melon yang bersifat kompleks dan
198 berukuran kecil. Oleh karena itu, diperlukan pendekatan tambahan untuk
memperkuat kemampuan model dalam menyoroti fitur-fitur penting pada citra daun
36 yang sering kali memiliki variasi pola dan warna yang halus. Salah satu pendekatan
4 yang efektif untuk mengatasi masalah tersebut adalah penerapan *Convolutional Block Attention Module (CBAM)* pada arsitektur *YOLOv8*. *CBAM* merupakan modul perhatian ringan yang mampu meningkatkan representasi fitur dengan mempelajari apa (*channel attention*) dan di mana (*spatial attention*) bagian penting dari citra yang perlu difokuskan[5].

24 Integrasi teknologi deteksi objek seperti *YOLO* dengan mekanisme perhatian *CBAM* terbukti mampu meningkatkan akurasi dan efisiensi dalam mendeteksi penyakit tanaman berbasis citra. Pendekatan ini memperkuat fokus model pada area penting dan memperbaiki hasil klasifikasi tanpa menambah beban komputasi yang besar. Penelitian ini menunjukkan bahwa penerapan *CBAM* pada model *YOLO-Papaya* berhasil mencapai nilai mAP sebesar 86,2% dalam mendeteksi berbagai penyakit pada buah pepaya. Keberhasilan tersebut membuktikan bahwa kombinasi

YOLO dan CBAM efektif untuk diterapkan pula pada deteksi penyakit daun melon yang memiliki pola visual kompleks dan gejala[6].

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah diuraikan, maka rumusan masalah dalam penelitian ini adalah sebagai berikut:

1. Belum diketahuinya sejauh mana kinerja model yolov8 yang dikombinasikan dengan Convolutional Block Attention Modul (CBAM) dalam mendeteksi penyakit daun melon berdasarkan metrik, *precision*, *F1-score*, *recall*, dan *mean Average Precision (mAP)*.
2. Implementasi model YOLOv8 + CBAM pada aplikasi Android untuk mendukung deteksi penyakit daun melon secara cepat dan akurat.
3. Perbandingan kinerja model yang sudah di latih dengan model *You only look once* (YOLOv7) dalam mendeteksi penyakit daun melon.

1.3 Batasan Masalah

Agar penelitian ini lebih terarah dan memudahkan dalam pembahasan, maka ditetapkan beberapa batasan masalah sebagai berikut:

1. Penelitian ini hanya difokuskan pada deteksi penyakit daun melon, tidak mencakup identifikasi penyakit pada bagian tanaman lain.
2. Jenis penyakit daun melon yang menjadi objek penelitian meliputi Embun Bulu, Mosaik Virus, Penyakit Bercak Daun Bersudut, Layu Fusarium.
3. Dataset citra daun melon dikumpulkan secara langsung dari lapangan dan dilengkapi dengan data pendukung dari sumber terbuka yang relevan, dengan kondisi pencahayaan dan latar belakang yang bervariasi.

4. Model deteksi yang digunakan dibatasi pada pengembangan arsitektur YOLOv8 dengan integrasi *Convolutional Block Attention Module (CBAM)* sebagai metode peningkatan akurasi dan efisiensi deteksi.
5. Pengujian performa model dilakukan menggunakan metrik evaluasi *precision*, *recall*, dan *mean Average Precision (mAP)* tanpa membahas aspek optimasi hyperparameter lanjutan.
6. Hasil penelitian difokuskan pada analisis performa sistem deteksi dan kontribusinya terhadap peningkatan akurasi identifikasi penyakit daun melon, tanpa membahas aspek ekonomi atau kebijakan pertanian secara luas.
7. Implementasi model hanya terbatas pada platform Android, dan hanya mencakup fitur-fitur dasar seperti pengambilan gambar tanaman, deteksi penyakit, dan hasil klasifikasi. Platform lain seperti iOS tidak termasuk dalam penelitian ini.

1.4 Tujuan Penelitian

Berdasarkan rumusan masalah di atas, maka tujuan dari penelitian ini adalah sebagai berikut:

1. Untuk mengetahui kinerja model YOLOv8 yang dikombinasikan dengan *Convolutional Block Attention Module (CBAM)* dalam mendeteksi penyakit daun melon berdasarkan metrik *precision*, *recall*, dan *mean Average Precision (mAP)*.
2. Mengimplementasikan model ke dalam aplikasi Android menggunakan *TensorFlow Lite* agar dapat digunakan secara praktis oleh masyarakat.

3. Menganalisis perbedaan kinerja akurasi antara model YOLOv8 yang dikombinasikan dengan CBAM dan model YOLOv7 melalui uji banding.

1.5 Manfaat Penelitian

Dari tujuan penelitian yang sudah dijabarkan, maka manfaat dari penelitian ini adalah sebagai berikut:

1. Untuk mempermudah petani dan pengguna dalam memperoleh informasi mengenai kondisi kesehatan daun melon secara cepat, akurat, dan praktis melalui sistem deteksi berbasis citra digital melalui aplikasi android.
2. Untuk membantu proses identifikasi penyakit daun melon yang sulit dibedakan secara visual, dengan memanfaatkan teknologi pengolahan citra dan kecerdasan buatan berbasis *Deep Learning*.
3. Untuk melakukan deteksi dan klasifikasi jenis penyakit daun melon menggunakan metode YOLOv8 yang dioptimalkan dengan *Convolutional Block Attention Module (CBAM)* guna meningkatkan akurasi, efisiensi, dan fokus model terhadap area daun yang terinfeksi.

1.6 Sistematika Penelitian

Sistematika penulisan yang digunakan pada penulisan tugas akhir dengan judul “Sistem Deteksi Penyakit Daun Melon Menggunakan *You Only Look Once (YOLOv8)* dan *Convolutional Block Attention Module (CBAM)*” adalah sebagai berikut:

BAB I PENDAHULUAN

1 Bab ini membahas latar belakang masalah yang mencakup pada penelitian, perumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, dan sistematika penelitian laporan.

BAB II TINJAUAN LITERATUR

Bab ini terdapat pendahuluan dan berisi jurnal terdahulu yang relevan dengan judul penelitian, untuk menguraikan dan membandingkan penelitian terdahulu dengan penelitian yang penulis lakukan untuk melihat perbedaan terhadap penelitian.

BAB III METODE PENELITIAN

86 Bab ini membahas mengenai objek penelitian, tahapan dan langkah langkah dalam penelitian.

BAB VI HASIL DAN PEMBAHASAN

Berisi hasil penelitian dan pembahasan

BAB V PENUTUP

Berisi kesimpulan dan saran

73

BAB II

TINJAUAN LITERATUR

59

Bab ini berisi uraian mengenai berbagai informasi, teori, konsep, serta temuan dari penelitian terdahulu sebagai bahan pembanding. Kajian pustaka dilakukan untuk mengidentifikasi kekuatan dan kelemahan penelitian sebelumnya yang memiliki keterkaitan dengan topik yang dibahas. Melalui studi literatur, peneliti memperoleh pemahaman yang lebih komprehensif tentang isu yang diteliti, mengetahui sejauh mana penelitian tersebut telah dikembangkan, serta menentukan kontribusi yang dapat diberikan dalam penelitian ini.

2.1 Tanaman Melon

10

Melon merupakan tanaman buah semusim yang berasal dari lembah Persia, buah melon ini masuk ke Indonesia dan mulai dibudidayakan pada tahun 1970. Pada saat itu, melon menjadi buah yang bergensi tinggi dan sangat mahal, konsumennya juga sangat terbatas, hanya kalangan yang tergolong ekonomi tinggi. Namun buah yang mengandung banyak air tersebut kini sudah bisa dinikmati semua kalangan[7].

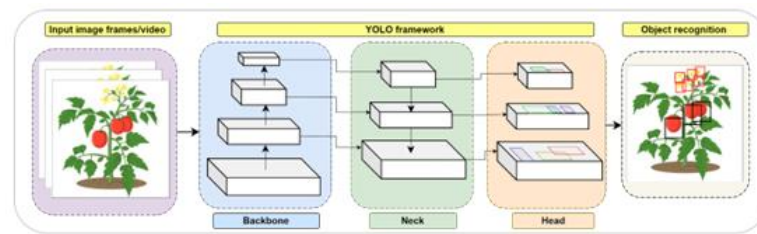
2.2 Tanaman Hidroponik

26

Hidroponik adalah metode bertanam tanpa memanfaatkan tanah sebagai media tanam, melainkan menggunakan bahan pengganti seperti rockwool, sabut kelapa, serat mineral, pasir, pecahan batu bata, atau serbuk kayu. Dalam sistem ini, tanaman disuplai dengan larutan mineral bernutrisi yang mengandung unsur hara yang dibutuhkan[8].

2.3 You Only Look Once (YOLO)

Metode *You Only Look Once (YOLO)* merupakan salah satu metode yang paling cepat dan akurat pada pendeteksian objek bahkan mampu melebihi hingga dua kali kemampuan algoritma lain[9]. Sederhananya sebuah instruksi pintar yang diberikan kepada program maupun mesin, salah satunya yaitu *Object detection* untuk mendeteksi citra buah menggunakan *You Only Look Once (YOLO)*. Metode yang dapat digunakan untuk pengenalan objek pada citra buah adalah *Deep Learning*. *You Only Look Once (YOLO)* merupakan salah satu model *Deep Learning* yang dapat digunakan untuk pengenalan objek [10].



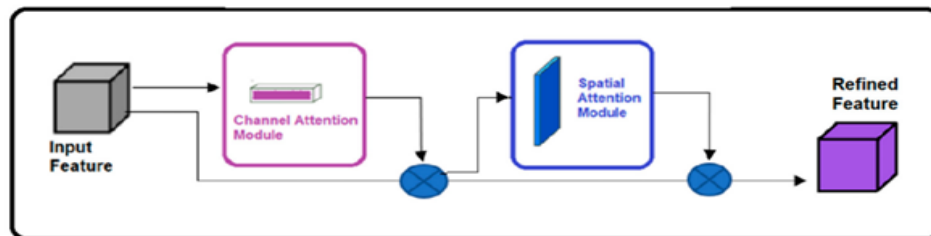
Gambar 2.1 Proses *You Only Look Once (YOLO)*[11]

2.4 Convolutional Block Attention Module (CBAM)

Convolutional Block Attention Module (CBAM) merupakan mekanisme perhatian (attention mechanism) yang dirancang untuk meningkatkan kemampuan representasi jaringan saraf konvolusional (CNN). CBAM bekerja dengan cara menekankan fitur-fitur penting pada data citra, sekaligus menekan bagian yang tidak relevan sehingga proses ekstraksi fitur menjadi lebih optimal. Modul ini memanfaatkan dua jenis perhatian sekaligus, yaitu perhatian saluran (*Channel Attention*) dan perhatian spasial (*Spatial Attention*)[12].

CBAM dapat diintegrasikan ke dalam berbagai arsitektur CNN maupun YOLO tanpa perubahan struktur signifikan karena bersifat modular dan ringan.

Penerapan CBAM pada beberapa lapisan jaringan mampu mengoptimalkan proses ekstraksi fitur dan secara signifikan meningkatkan performa deteksi pada model.



Gambar 2.2 Proses Convolutional Block Attention Module (CBAM)[6]

2.5 Penelitian Terdahulu

Penelitian terdahulu merupakan proses pencarian, mengumpulkan, dan analisis informasi dari berbagai sumber tertulis yang memiliki keterkaitan dengan topik atau permasalahan penelitian. Pada penelitian ini, peneliti memanfaatkan jurnal ilmiah yang memuat artikel penelitian dan studi kasus yang bertujuan untuk memberikan dasar teori yang kuat.

Tabel 2.1 Penelitian Terdahulu

No	Judul Penelitian	Metode Penelitian	Hasil Penelitian
1	Klasifikasi Penyakit pada Citra Daun Melon Menggunakan Algoritma Convolution Neural Network. Oleh: Mochammad Sholikhin, S.Kom, Reddy Alexandro H., S.Kom., M.Kom. (2022)[13]	Convolutional Neural Network (CNN).	Proses meliputi preprocessing, segmentasi citra, augmentasi data (<i>rotasi, zoom, pergeseran horizontal dan vertikal</i>), training, dan testing. Hasil pengujian menunjukkan sistem mampu bekerja dengan baik, dengan tingkat akurasi 89% pada aplikasi komputer (106 data sesuai dari 119 data uji) dan 90%

Tabel 2.1 Penelitian Terdahulu Lanjutan

No	Judul Penelitian	Metode Penelitian	Hasil Penelitian
2	Accuracy Optimization of Convolutional Neural Network Model in Chili Plant Disease Classification. Oleh: Muh.Rasyid Ridha,Risma Asriati, Dwi Yuli Prasetyo.(2025)[14]	CNN dengan arsitektur MobileNetV2	Model MobileNetV2 Custom mampu mengklasifikasikan 4 kelas daun cabai (sehat, daun keriting, bercak daun, daun menguning) dengan akurasi 92,94% dan AUC 0,9904. Nilai <i>precision</i> , <i>recall</i> , dan <i>F1-score</i> berada di atas 0,90, serta terbukti lebih baik dibandingkan CNN murni dan MobileNetV2 baseline. Model berhasil diimplementasikan ke aplikasi Android untuk deteksi penyakit secara real-time.
3	Evaluasi Augmentasi Data Pada Deteksi Penyakit Daun Tebu dengan yolov8. Oleh: Ibnu Topan Adib Amrulloh, Betha Nurina Sari, Tesa Nur Padilah[15].	Deteksi objek menggunakan YOLOv8	Penerapan data augmentasi terbukti meningkatkan kinerja YOLOv8 secara signifikan, ditunjukkan oleh kenaikan nilai mAP dari 40,3% (tanpa augmentasi) menjadi 50,5% (dengan augmentasi). Augmentasi membantu mengurangi overfitting dan meningkatkan kemampuan generalisasi model dalam mendeteksi penyakit daun tebu.
4	Penerapan YOLOv8 Dalam Deteksi Penyakit Daun Jambu Air Secara Real-time. Oleh: Nurul Safina, Muhammad Fikry, Cut Agusniar. (2024)[16].	Menggunakan metode YOLOv8 berbasis CNN	Hasil pengujian pada 70 citra menghasilkan precision 89%, <i>recall</i> 96%, accuracy 91%, <i>F1-score</i> 93%, dan mAP 89%. Sistem dapat melakukan deteksi secara real-time melalui webcam dengan menampilkan bounding box, tingkat kepercayaan,dan rekomendasi penanganan singkat. Penelitian membuktikan bahwa YOLOv8 efektif dan andal untuk deteksi dini penyakit daun jambu air berbasis web.

Tabel 2.1 Penelitian Terdahulu Lanjutan

No	Judul Penelitian	Metode Penelitian	Hasil Penelitian
5	Implementation of you only look once v8 algorithm in potato leaf disease detection system. Oleh: Bagus Kurniawan Ekhsanto, Bagus Adhi Kusuma, Adam Prayogo Kuncoro. (2024)[17].	YOU ONLY LOOK ONCE V8 (yolov8)	Dengan memanfaatkan teknologi YOU ONLY LOOK ONCE V8 (YOLOv8), melalui evaluasi menggunakan matriks <i>precision</i> dan <i>recall</i> , hasilnya menunjukkan tingkat keberhasilan yang signifikan, dengan akurasi <i>precision</i> masing-masing untuk <i>early_blight</i> sebesar 87%, tanaman sehat 81%, dan <i>late_blight</i> 97%. Sementara itu, hasil <i>recall</i> untuk ketiga kategori tersebut juga mencapai 87%, 81%, dan 97% secara berurutan. Dengan akurasi keseluruhan mencapai 88%, temuan ini menegaskan bahwa sistem deteksi yang dikembangkan berhasil dalam mengidentifikasi penyakit daun kentang dengan akurasi tinggi.
6	Deteksi dan klasifikasi hama potato beetle pada tanaman kentang menggunakan yolov8[18]	YOU ONLY LOOK ONCE V8 (yolov8)	Hasil penelitian menunjukkan model yang dikembangkan memiliki tingkat akurasi yang memadai dalam mendeteksi dan mengklasifikasikan potato beetle. Evaluasi model melibatkan pengukuran Mean Average Precision (<i>mAP</i>) dan <i>F1-Score</i> berdasarkan konsep Precision dan Recall. Dalam evaluasi kelas objek individu, model ini berhasil mendeteksi objek <i>guk (potato beetle)</i> dengan tingkat akurasi mencapai <i>precision</i> sebesar 88,1% dan <i>recall</i> sebesar 90,3%.
7	Implementasi Algoritma YOLOv8 (You Only Look Once) dalam Deteksi Penyakit Daun	YOLOv8-s	Hasil terbaik diperoleh pada skenario 70:20:10 dengan 100 <i>epoch</i> dan <i>batch size</i> 16, menghasilkan <i>precision</i> 0,994, <i>recall</i> 0,989, <i>mAP50</i> 0,990, dan <i>mAP50-95</i> 0,927.

Tabel 2.1 Penelitian Terdahulu Lanjutan

No	Judul Penelitian	Metode Penelitian	Hasil Penelitian
	Durian. Oleh: Raihan Restu Putra, Maimunah, Dimas Sasongko(2024) [19].		Model mampu mendeteksi empat kelas penyakit daun durian secara akurat dan cepat, sehingga layak diterapkan untuk deteksi dini penyakit tanaman durian.
8	Penerapan CBAM Pada Xception Untuk Klasifikasi Penyakit Daun Tebu. Oleh: SISILIA DEODORA SITEPU. (2025)[20].	Penerapan Convolutional Block Attention Module (CBAM) pada arsitektur Xception	Pendekatan eksperimen dilakukan dengan membandingkan performa model <i>Xception</i> tanpa CBAM dan <i>Xception</i> dengan CBAM. Hasil yang diperoleh menunjukkan bahwa penerapan CBAM meningkatkan akurasi model dari 88.33% menjadi 92.89%, dengan peningkatan signifikan pada <i>precision</i> , <i>recall</i> dan <i>F1-Score</i> , terutama pada kategori Rust yang mencatatkan <i>F1-Score</i> tertinggi 98.90%.
9	Yolo-Papaya: A Papaya Fruit Disease Detector and Classifier Using CNNsandConvolutional Block Attention Modules. Oleh: Jairo Lucas de Moraes, Jorcy de Oliveira Neto, Claudine Badue, Thiago Oliveira-Santos, & Alberto F. de Souza (2023) [6].	Menggunakan YOLOv7 yang dimodifikasi dengan Convolutional Block Attention Module (CBAM).	Model Yolo-Papaya mencapai performa terbaik dengan nilai mAP sebesar 86,2%, lebih tinggi dibandingkan YOLOv7 (83,8%), ResNet50 (83,4%), YOLO-R (82,3%), dan EfficientDet-B3 (67,1%). Hasil menunjukkan bahwa penambahan CBAM meningkatkan akurasi deteksi tanpa menambah waktu inferensi secara signifikan, sehingga efektif untuk aplikasi <i>real-time</i> di bidang pertanian.

Tabel 2.1 Penelitian Terdahulu Lanjutan

No	Judul Penelitian	Metode Penelitian	Hasil Penelitian
10	YOLOV5-CBAM-C3TR: An Optimized Model Based on Transformer Module and Attention Mechanism for Apple Leaf Disease Detection. Oleh: Meng Lv and Wen-Hao Su. (2024) [21].	YOLOv5 yang dioptimasi dengan Convolutional Block Attention Module (CBAM) dan Transformer Encoder (C3TR).	Model YOLOV5-CBAM-C3TR mampu meningkatkan performa deteksi penyakit daun apel secara signifikan dengan $mAP@0.5$ sebesar 73,4%, $precision$ 70,9%, dan $recall$ 69,5%. Model ini juga mencapai akurasi rata-rata 92,4% pada pengujian 208 citra di lingkungan nyata, serta menunjukkan kemampuan yang baik dalam membedakan penyakit daun apel yang memiliki karakteristik visual serupa.

Berdasarkan hasil perbandingan jurnal disajikan tabel 2.1 dapat dijelaskan sebagai berikut:

Penelitian menyimpulkan bahwa penerapan *Convolutional Neural Network* (CNN) untuk mengklasifikasikan penyakit pada citra daun melon. Dataset yang digunakan berjumlah 4.000 citra daun melon yang telah melalui proses preprocessing, segmentasi, dan augmentasi data. Hasil pengujian menunjukkan bahwa model CNN mampu melakukan klasifikasi dengan akurasi 89% pada aplikasi komputer dan 90% pada aplikasi smartphone, sehingga dapat disimpulkan bahwa metode CNN efektif dan layak digunakan sebagai alat bantu diagnosis penyakit daun melon secara otomatis.

Penelitian oleh Muh.Rasyid Ridha,Risma Asriati, Dwi Yuli Prasetyo (2025) dari Universitas Islam Indragiri melakukan klasifikasi penyakit cabai menggunakan *Convolutional Neural Network* (CNN) karena identifikasi manual masih kurang akurat. Sistem mengklasifikasikan empat kelas daun, yaitu daun sehat, daun

keriting, bercak daun, dan daun menguning. Model dibangun menggunakan arsitektur *MobileNetV2* dengan *transfer learning* dan *fine-tuning*. Hasil pengujian menunjukkan akurasi 92,94%, sehingga metode CNN dengan *MobileNetV2* dinilai efektif untuk klasifikasi penyakit daun cabai secara otomatis.

Penelitian ini mengevaluasi pengaruh data augmentasi terhadap kinerja YOLOv8 dalam mendeteksi penyakit daun tebu menggunakan empat kelas (sehat, mosaik, karat daun, dan redrot). Dengan menerapkan augmentasi berupa horizontal flipping, jittering scale, dan random brightness, nilai *mAP* meningkat dari 40,3% menjadi 50,5% dibandingkan pelatihan tanpa augmentasi. Hasil ini menunjukkan bahwa augmentasi data mampu meningkatkan kemampuan generalisasi model serta mengurangi *overfitting* dalam deteksi penyakit daun tebu.

Penelitian ini menerapkan YOLOv8 untuk mendeteksi penyakit daun jambu air secara *real-time* berbasis web menggunakan dataset 754 citra yang diperluas melalui augmentasi hingga 2089 citra. Model menunjukkan kinerja tinggi dengan *precision* 89%, *recall* 96%, *accuracy* 91%, dan *mAP* 89%, sehingga efektif untuk mendukung deteksi dini penyakit tanaman.

Penelitian ini menunjukkan bahwa YOLOv8 efektif untuk mendeteksi penyakit daun kentang (*healthy*, *early blight*, dan *late blight*) menggunakan dataset 750 citra, dengan kinerja baik ditunjukkan oleh akurasi 88% serta *precision* 87% (*early blight*), 81% (*healthy*), 97% (*late blight*) serta *recall* 87%, 81%, dan 97%, sehingga mendukung deteksi dini penyakit tanaman secara akurat.

Penelitian ini menerapkan YOLOv8 untuk mendeteksi dan mengklasifikasikan hama *potato beetle* pada tanaman kentang menggunakan data

35
2
citra yang dibagi menjadi data latih, validasi, dan uji. Hasil evaluasi menunjukkan model mencapai mAP50 sebesar 81,8%, dengan *precision* 78,1% dan *recall* 89,8%, serta pada kelas utama *potato beetle* memperoleh *precision* 88,1% dan *recall* 90,3%, sehingga membuktikan bahwa YOLOv8 efektif untuk mendukung deteksi hama kentang secara otomatis.

40
Penelitian ini menunjukkan bahwa YOLOv8-s efektif dalam mendeteksi penyakit daun durian menggunakan 420 citra, dengan kinerja terbaik pada skenario 70:20:10 yang menghasilkan *precision* 99,4%, *recall* 98,9%, *mAP50* 99,0%, dan *mAP50-95* 92,7%, sehingga mendukung deteksi dini penyakit tanaman.

199
Penelitian ini membuktikan bahwa penambahan CBAM pada model *Xception* meningkatkan akurasi klasifikasi penyakit daun tebu dari 88,33% menjadi 92,89%, serta memperbaiki nilai *precision*, *recall*, dan *F1-score*. CBAM membuat model lebih fokus pada fitur penting gambar sehingga hasil klasifikasi lebih akurat, meskipun waktu pelatihan menjadi lebih lama.

Penelitian ini mengembangkan model Yolo-Papaya yang menggabungkan YOLOv7 dengan CBAM untuk mendeteksi penyakit pada buah pepaya menggunakan dataset 23.158 citra, dan hasilnya menunjukkan peningkatan kinerja hingga mAP 86,2%, sehingga lebih akurat dan efektif dibandingkan YOLOv7 standar.

Penelitian ini mengembangkan model YOLOv5-CBAM-C3TR yang menggabungkan YOLOv5 dengan *mekanisme attention* (CBAM) dan modul *Transformer* untuk mendeteksi penyakit daun apel, dan hasilnya menunjukkan

peningkatan kinerja dengan nilai *mAP* mencapai sekitar 94%, sehingga lebih akurat dan efektif dibandingkan YOLOv5 standar dalam mendeteksi penyakit daun apel.

2.5 Rangkuman

Berbagai penelitian terdahulu menunjukkan bahwa metode deep learning seperti CNN dan model deteksi objek berbasis YOLO (YOLOv5, YOLOv7, dan YOLOv8) terbukti mampu menghasilkan akurasi, *precision*, *recall*, dan *mAP* yang tinggi dalam klasifikasi serta deteksi penyakit dan hama tanaman. CNN unggul dalam melakukan klasifikasi citra daun secara detail, sedangkan YOLO efektif dalam mendeteksi objek secara cepat dan real-time. Selain itu, penggunaan teknik data *augmentation*, *transfer learning*, serta penambahan mekanisme *attention* seperti CBAM dan modul *Transformer* terbukti dapat meningkatkan fokus model terhadap fitur penting citra sehingga kinerja model menjadi lebih optimal. Penelitian lain yang menggunakan model EfficientNet juga menunjukkan hasil yang sangat baik dalam klasifikasi penyakit tanaman. Model *EfficientNet-B2* memperoleh akurasi validasi sebesar 97,77% dan F1-Score sebesar 98%, sehingga menunjukkan bahwa arsitektur EfficientNet memiliki kemampuan yang tinggi dalam mengenali pola penyakit pada citra daun tanaman. Meskipun demikian, metode-metode tersebut juga memiliki keterbatasan, seperti kebutuhan komputasi yang relatif tinggi, waktu pelatihan yang lebih lama, serta risiko *overfitting* pada arsitektur yang kompleks apabila jumlah data terbatas, sehingga diperlukan optimasi model agar lebih efisien dan layak diterapkan pada sistem deteksi penyakit tanaman secara praktis.

Meninjau keterbatasan tersebut, penelitian terdahulu menunjukkan bahwa pengembangan model YOLO yang dikombinasikan dengan mekanisme *attention*

seperti CBAM mampu meningkatkan akurasi dan kestabilan deteksi objek secara signifikan. YOLO unggul dalam kecepatan dan kemampuan deteksi *real-time*, sedangkan CBAM membantu model lebih fokus pada fitur penting baik pada dimensi kanal maupun spasial. Berbagai penelitian melaporkan bahwa kombinasi YOLO + CBAM mampu meningkatkan kinerja dengan nilai mAP yang umumnya berada pada kisaran 86% hingga 94%, bahkan lebih tinggi dibandingkan YOLO standar. Peningkatan ini juga terlihat pada nilai *precision* dan *recall* yang menjadi lebih stabil dan akurat. Meskipun demikian, penerapan YOLO + CBAM masih terbatas pada objek dan dataset tertentu. Oleh karena itu, penelitian ini dilakukan untuk menguji dan memvalidasi efektivitas kombinasi YOLO + CBAM dalam meningkatkan performa deteksi penyakit tanaman, khususnya pada data lapangan dengan variasi pencahayaan, sudut pengambilan, dan latar belakang yang kompleks.

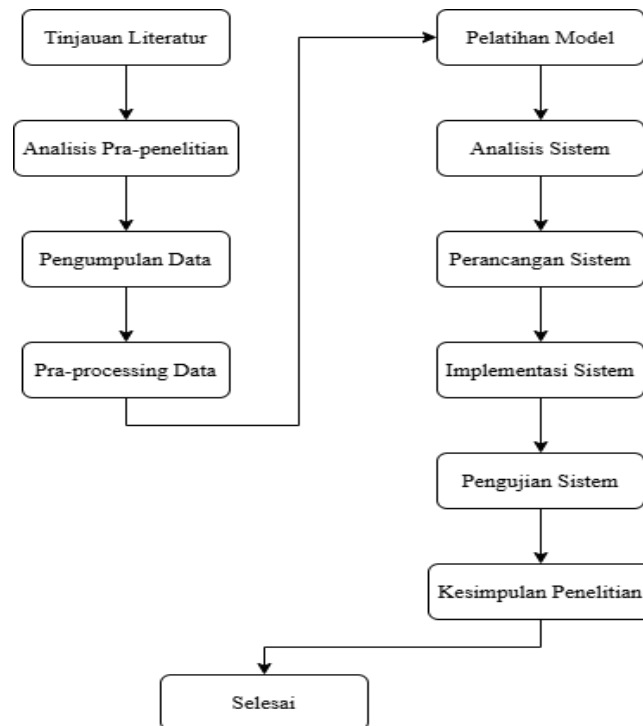
BAB III

METODE PENELITIAN

Bab ini menguraikan tahapan dan prosedur yang digunakan dalam pelaksanaan penelitian. Isi bab mencakup jenis penelitian, rancangan penelitian, tahapan pengumpulan dataset, proses pelabelan, metode analisis model, serta proses evaluasi akurasi. Seluruh tahapan disusun secara sistematis agar tujuan penelitian dapat dicapai secara optimal. Penelitian ini menerapkan pendekatan kuantitatif, karena berfokus pada pengolahan data berbasis angka, yakni citra yang telah diberi label sesuai kategorinya, serta analisis performa melalui metrik evaluasi seperti akurasi, presisi, dan recall. Metode penelitian yang digunakan ini meliputi proses akuisisi dataset, prapengolahan data, pelatihan model YOLO (*You Only Look Once*) dengan kombinasi CBAM (*Convolutional Block Attention Module*), pengujian performa model, dan interpretasi hasil untuk mengukur tingkat efektivitas sistem.

3.1 Kerangka Penelitian

Kerangka penelitian ini dirancang untuk menggambarkan alur sistematis dalam pengembangan aplikasi deteksi penyakit tanaman daun melon menggunakan algoritma Yolo (*You Only Look Once*) dengan kombinasi penambahan CBAM (*Convolutional Block Attention Module*). Secara visual tahapan-tahapan dalam penelitian ini di sajikan pada gambar 3.1



Gambar 3.1 Kerangka Penelitian

3.1.1 Tinjauan Literatur

Tinjauan literatur dilakukan untuk pemahaman yang mendalam mengenai model yang digunakan dalam mendeteksi penyakit tanaman daun melon. Literatur yang dikumpulkan dari berbagai sumber seperti jurnal dan artikel. Proses pengumpulan dilakukan dengan menggunakan kata kunci: “penyakit tanaman melon”, “klasifikasi penyakit tanaman”, “*Yolo (You Only Look Once)*”, “*CBAM (Convolutional Block Attention Module)*” dan kata kunci lain yang relevan.

3.1.2 Analisis Pra-penelitian

Analisis pra-penelitian dilakukan sebagai tahap awal untuk merumuskan masalah, tujuan, serta pendekatan yang tepat dalam pengembangan sistem deteksi penyakit tanaman melon. Permasalahan utama yang diidentifikasi dalam penelitian ini adalah sulitnya membedakan kondisi daun melon yang sehat dan yang terinfeksi penyakit hanya melalui pengamatan visual, sehingga diperlukan sebuah sistem

otomatis yang mampu mengklasifikasikan dan mendeteksi penyakit daun melon secara akurat. Berdasarkan permasalahan tersebut serta kebutuhan penerapan di kondisi lapangan, penelitian ini memilih arsitektur YOLO yang dikombinasikan dengan modul CBAM, karena mampu menekankan fitur penting pada citra sehingga proses deteksi penyakit daun melon dapat dilakukan secara cepat, akurat, dan real-time. Selain itu, peneliti melakukan survei awal mengenai ketersediaan dataset daun melon, identifikasi ciri visual pembeda antara daun sehat dan berpenyakit, serta peluang penerapan model deteksi ke dalam aplikasi Android sebagai solusi praktis di lapangan.

3.1.3 Pengumpulan Data

Tahapan pengumpulan data dilakukan untuk mendapatkan citra penyakit melon dari empat penyakit, yaitu embun bulu, busuk bakteri, bercak daun bersudut, mosaic virus, layu fusarium sebagai bahan pelatihan model klasifikasi citra. Data citra diperoleh di usaha perkebunan milik bapak Ahmad Jamil dengan alamat di jalan Prof M Yamin SH, GG Bunga Padi Parit 15 No 46. Pengambilan gambar dilakukan dengan menggunakan kamera Samsung A36 5G, saat mengambil gambar dilakukan secara outdoor dengan pencahayaan alami agar menampilkan kondisi nyata.

3.1.4 Pra-processing data

Pra-processing data adalah tahap awal yang dilakukan sebelum data digunakan untuk melatih model, dengan tujuan menyiapkan citra dalam format dan kondisi yang optimal agar arsitektur YOLO yang dipadukan dengan modul CBAM mampu mengenali pola penyakit daun melon secara lebih efektif. Pada tahap ini dilakukan pelabelan kategori penyakit pada citra daun melon serta augmentasi gambar untuk

meningkatkan keragaman data dan mendukung performa deteksi model secara maksimal.



Gambar 3.2 Data training penyakit tanaman melon bercak daun bersudut, mosaik virus, layu fusarium, dan embun bulu[22].

Penyakit pada tanaman melon seperti embun bulu, layu fusarium, mosaik virus, dan bercak daun bersudut dapat menyebabkan penurunan pertumbuhan dan kualitas tanaman. Gejala yang muncul umumnya berupa perubahan warna daun, bercak, layu, hingga kerusakan jaringan daun. Penyakit tersebut dapat menyebar melalui tanah, benih, maupun vektor serangga, serta berkembang cepat pada kondisi lingkungan yang mendukung[23].

3.1.5 Pelatihan Model

Dalam tahap ini pelatihan model dilakukan secara lokal menggunakan perangkat laptop/komputer dengan sistem windows. Proses pelatihan menggunakan bahasa pemrograman *Python* melalui *framework Ultralytics YOLOv8*, yang

menyediakan fasilitas lengkap untuk pelatihan deteksi objek secara cepat dan teroptimasi. Modul CBAM ditambahkan ke arsitektur YOLO dengan menempatkan pada bagian backbone, karena backbone itu tahapan ekstraksi fitur utama. Sebelum proses pelatihan dimulai, dataset dibagi menjadi tiga bagian, yaitu 70% sebagai data pelatihan (*training set*), 20% sebagai data validasi (*validation set*), dan 10% sebagai data pengujian (*test set*).

3.1.6 Analisis Sistem

Analisis sistem dilakukan untuk menentukan kebutuhan teknis yang diperlukan dalam pengembangan sistem deteksi penyakit daun melon berbasis Android. Pada tahap ini ditetapkan komponen perangkat keras maupun perangkat lunak yang paling sesuai untuk menunjang proses penelitian, mulai dari pelatihan model YOLO, konversi model, hingga persiapan implementasi.

Berikut ini merupakan rincian perangkat keras dan perangkat lunak yang digunakan:

Tabel 3.1 Spesifikasi Perangkat Keras (Hardware)

Komponen	Spesifikasi
Merek & Tipe	Axioo HYPE 5
Prosesor	Intel(R) Core (TM) i5-1235U
GPU	Intel Iris Xe Graphics
RAM	16 GB
Internal	512 SSD
Sistem Operasi	Windows 11 Pro
Perangkat Tambahan	Smartphone Android

Tabel 3.2 Perangkat Lunak (Software) yang digunakan

Nama Perangkat Lunak	Fungsi
Google Collab	Pelatihan Model
Anaconda Navigator	Untuk melakukan Labeling
TensorFlow & TFLite	Framework untuk membangun dan mengkonversi model ke format .tflite
Android Studio	Platform pengembangan aplikasi Android dan integrasi model.

3.1.7 Perancangan Sistem

Tahapan perancangan sistem ini untuk menggambarkan alur kerja serta struktur deteksi penyakit daun melon berbasis Android sebelum tahap implementasi di jalankan. Proses perancangan ini meliputi antar muka pengguna (*User Interface*), alur interaksi pengguna (*User Flow*), serta pemodelan fungsional sistem dalam bentuk diagram.

Aplikasi ini dirancang dengan antarmuka yang sederhana dan mudah dipahami untuk memastikan kenyamanan bagi semua pengguna. Sebelum pengguna masuk ke halaman utama, aplikasi menampilkan dua halaman splash screen. Halaman pertama menampilkan logo sebagai identitas aplikasi, sedangkan halaman kedua berisi petunjuk singkat mengenai cara penggunaan. Setelah tahap tersebut, pengguna diarahkan menuju halaman utama yang menyediakan fitur utama untuk melakukan klasifikasi citra penyakit daun melon.

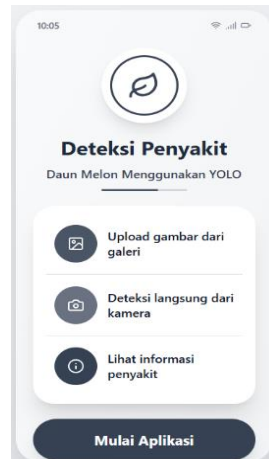
3.1.7.1 Desain Antarmuka Pengguna (*User Interface*)

Guna mendukung kemudahan pengguna aplikasi klasifikasi penyakit daun melon, antarmuka pengguna dirancang dengan prinsip kesederhanaan dan kejelasan tampilan. Aplikasi ini terdiri dari beberapa halaman utama, yaitu tampilan awal pembuka, petunjuk penggunaan, proses deteksi, dan halaman detail deteksi. Seluruh elemen disusun agar dapat berfungsi optimal pada perangkat Android dan tetap ramah bagi pengguna awam.

a. Tampilan Awal Aplikasi

Gambar ini menunjukkan halaman utama aplikasi Deteksi Penyakit Daun Melon Menggunakan YOLO yang menyediakan fitur unggah gambar dari

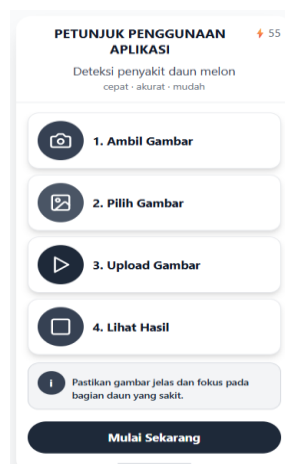
galeri, deteksi melalui kamera, dan informasi penyakit. Pengguna dapat memulai penggunaan aplikasi melalui tombol Mulai Aplikasi.



Gambar 3.3 Tampilan Awal Aplikasi

b. Halaman Petunjuk Aplikasi

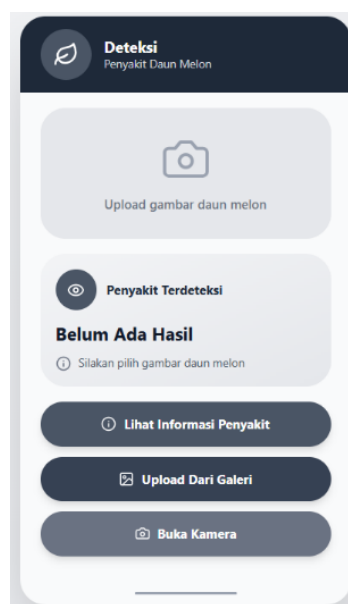
Tampilan halaman petunjuk penggunaan aplikasi deteksi penyakit daun melon yang menampilkan langkah-langkah penggunaan, yaitu mengambil gambar, memilih gambar, mengunggah gambar, dan melihat hasil deteksi. Di bagian bawah terdapat tombol "Mulai Sekarang" untuk memulai proses identifikasi penyakit pada daun melon.



Gambar 3.4 Halaman Petunjuk Aplikasi

c. Halaman Aplikasi Deteksi

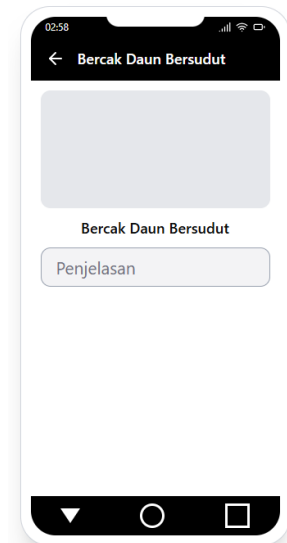
Halaman Gambar ini menampilkan halaman **Deteksi Penyakit Daun Melon** yang digunakan untuk mengunggah gambar daun melon dan melihat hasil deteksi. Pada tampilan ini belum terdapat hasil deteksi karena pengguna belum memilih gambar. Aplikasi juga menyediakan tombol untuk melihat informasi penyakit, mengunggah gambar dari galeri, dan membuka kamera.



Gambar 3.5 Tampilan Halaman Deteksi

d. Halaman Detail Deteksi

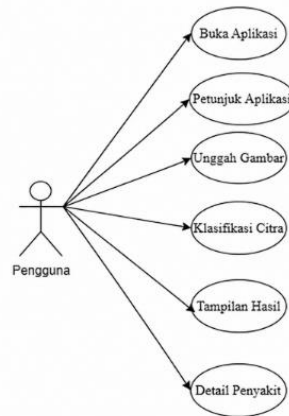
Halaman detail penyakit menampilkan area pratinjau gambar dibagian atas, kemudian diikuti dengan nama penyakit yang terdeteksi. Pada bagian bawah, disediakan ruang “Penjelasan” yang berfungsi sebagai tempat untuk menampilkan deskripsi mengenai penyakit tersebut dan dampak yang ditimbulkan pada tanaman.



Gambar 3.6 Tampilan Detail Penyakit

3.1.7.2 Diagram Use Case

Diagram *use case* digunakan untuk merepresentasikan hubungan interaksi antara pengguna sebagai aktor dengan sistem dalam menjalankan fungsi-fungsi utama aplikasi deteksi penyakit daun melon. Pada sistem ini, aktor yang terlibat adalah pengguna, yaitu individu yang menggunakan aplikasi untuk melakukan identifikasi penyakit berdasarkan citra daun melon. Use case yang ditampilkan meliputi proses pengambilan atau pemilihan gambar melalui kamera maupun galeri, pelaksanaan proses klasifikasi, serta peninjauan hasil klasifikasi yang dihasilkan oleh sistem.



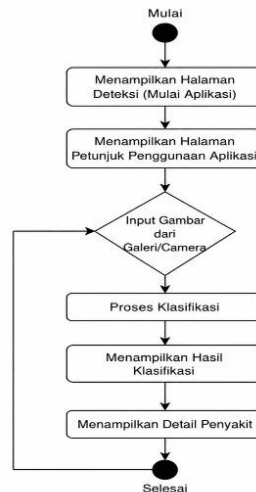
Gambar 3.7 Diagram Use Case Sistem Deteksi Penyakit Daun Melon

Use case diagram merepresentasikan hubungan antara pengguna dan sistem aplikasi deteksi penyakit daun melon berbasis Android. Pengguna berperan sebagai aktor utama yang berinteraksi langsung dengan sistem untuk menjalankan seluruh fungsi aplikasi. Aktor utama adalah pengguna yang dapat melakukan lima aktivitas utama, yaitu menuju halaman petunjuk aplikasi penggunaan, mengambil gambar melalui kamera, memilih gambar dari galeri, melihat hasil klasifikasi penyakit, serta melihat detail penyakit. Diagram ini menunjukkan bahwa fitur aplikasi disusun secara terstruktur untuk memudahkan pengguna memperoleh informasi diagnosis penyakit tanaman secara cepat dan sistematis.

3.1.7.3 Diagram Aktivitas

Diagram aktivitas ini digunakan untuk menggambarkan alur kerja pada sistem deteksi penyakit daun melon dalam aplikasi android, mulai pengguna membuka aplikasi sampai menampilkan hasil klasifikasi. Alur ini menjelaskan urutan aktivitas yang dilakukan sistem berdasarkan interaksi pengguna dan proses internal aplikasi.

Berikut merupakan diagram aktivitas sistem:



Gambar 3.8 Diagram Aktivitas

Diagram aktivitas pada aplikasi deteksi penyakit daun melon berbasis Android menggambarkan urutan proses kerja sistem secara sistematis dari awal hingga akhir. Alur dimulai ketika aplikasi dijalankan dan sistem menampilkan halaman deteksi tampilan mulai sebagai tampilan awal, kemudian pengguna diarahkan menuju halaman petunjuk penggunaan aplikasi. Pada tahap berikutnya, pengguna memasukkan citra daun melon yang di peroleh melalui kamera atau galeri sebagai data masukan. Sistem selanjutnya menjalankan proses klasifikasi citra menggunakan model YOLO yang telah dilatih. Hasil klasifikasi kemudian ditampilkan kepada pengguna, dilengkapi dengan informasi lanjutan berupa detail penyakit yang teridentifikasi. Diagram ini menunjukkan bahwa alur kerja aplikasi dirancang secara runtun dan berorientasi pada kemudahan pengguna dalam memperoleh hasil deteksi penyakit secara jelas dan efisien.

3.1.8 Implementasi Sistem

Tahap implementasi sistem ini merupakan tahapan penerapan dari hasil analisis dan perancangan yang telah disusun sebelumnya. Pada fase ini,

179 pengembangan sistem dilakukan dengan menggunakan metode *Waterfall*, yaitu model pengembangan perangkat lunak yang dilaksanakan secara bertahap dan berurutan, mulai dari analisis kebutuhan, perancangan sistem, implementasi, hingga pengujian. Proses implementasi diawali dengan pelatihan model YOLO menggunakan dataset penyakit daun melon yang telah melalui tahap praproses. Pelatihan model dilakukan pada lingkungan *Google Collab* dengan menerapkan arsitektur YOLOv8 yang dikombinasikan dengan modul *Convolutional Block Attention Modul (CBAM)* serta memanfaatkan *framework TensorFlow*, sehingga dihasilkan model klasifikasi yang optimal dan memiliki tingkat akurasi yang baik. 180 Selanjutnya, model yang telah dilatih dikonversi ke dalam format *TensorFlow Lite (.tflite)* agar dapat diintegrasikan dan dijalankan secara efisien pada perangkat berbasis Android. 94

Selanjutnya tahap pengembangan aplikasi berbasis Android dengan menggunakan Android Studio, yang meliputi integrasi model *TensorFlow Lite (TFLite)* pada sisi *backend* serta perancangan antarmuka pengguna (*frontend*) yang sederhana dan mudah dipahami. 1 Aplikasi dikembangkan agar dapat menerima input berupa gambar dari kamera atau galeri, kemudia menjalankan proses inferensi untuk menampilkan hasil klasifikasi penyakit daun melon. 1 Proses implementasi dilakukan secara terstruktur mengikuti alur metode *Waterfall*, sehingga menjadi landasan yang jelas untuk pelaksanaan tahap pengujian sistem pada fase selanjutnya.

1 3.1.9 Pengujian Sistem

Tahap pengujian sistem dilakukan untuk menilai kinerja dan tingkat keakuratan sistem deteksi penyakit daun melon berbasis Android yang telah dikembangkan. 1 Pengujian ini menggunakan dataset pengujian (*testing*) yang telah

dipisahkan dari dataset pelatihan sebelumnya. Model YOLO yang telah dilatih dievaluasi menggunakan matrik performa seperti *accuracy*, *precision*, *recall*, dan *F1-score* guna mengukur kemampuan sistem dalam mengklasifikasi jenis penyakit daun melon secara tepat. Selain itu, evaluasi juga dilakukan dengan menggunakan confusion matrix untuk menganalisis sebaran hasil prediksi serta mengidentifikasi kesalahan klasifikasi, seperti *false positif* dan *false negative*. Pengujian sistem turut mencakup uji coba langsung melalui aplikasi Android dengan memanfaatkan fitur input citra dari kamera maupun galeri, sehingga dapat dipastikan bahwa proses klasifikasi berjalan dengan baik dan hasil prediksi ditampilkan secara real-time sesuai dengan fungsi yang diharapkan.

3.1.10 Kesimpulan Penelitian

Pada tahap ini dilakukan penarikan kesimpulan sebagai rangkuman akhir penelitian dengan mengacu pada tujuan yang telah ditetapkan. Kesimpulan disusun berdasarkan hasil analisis sampai pengujian sistem yang telah dilakukan.

BAB VI

HASIL DAN PEMBAHASAN

Bab ini menyajikan hasil yang diperoleh dari proses pelatihan dan pengujian model deteksi penyakit pada daun tanaman melon menggunakan algoritma YOLO dengan kombinasi *Convolutional Block Attention Module (CBAM)*. Penelitian ini menggunakan pendekatan kuantitatif dengan memanfaatkan dataset citra daun melon yang mencakup beberapa kategori penyakit, yaitu embun bulu, layu fusarium, mosaik virus, dan bercak daun bersudut. Dataset diperoleh dari pengumpulan data mandiri untuk mencakup variasi kondisi pencahayaan dan latar belakang. Tahapan analisis dimulai dari *pra-pemrosesan* data, pelabelan citra, pelatihan model YOLO-CBAM, hingga evaluasi performa menggunakan metrik seperti *precision*, *recall*, *mAP*, dan *loss* guna memberikan gambaran menyeluruh terhadap kinerja model.

4.1 Hasil

Bab ini menyajikan hasil dari serangkaian proses pelatihan, pengujian, dan evaluasi model deteksi penyakit tanaman melon berbasis algoritma YOLO dengan kombinasi *Convolutional Block Attention Module (CBAM)*. Penelitian ini bertujuan mengembangkan model yang mampu mendeteksi berbagai jenis penyakit daun melon secara otomatis melalui citra digital.

4.1.1 Pengumpulan Data

Tahap awal dalam pengembangan model deteksi dilakukan dengan mengumpulkan dataset citra daun melon. Dataset tersebut terdiri dari beberapa kategori penyakit, yaitu embun bulu, layu fusarium, mosaik virus, dan bercak daun bersudut. Proses pengambilan gambar dilakukan secara langsung menggunakan kamera smartphone Samsung Galaxy A36 dengan variasi kondisi pencahayaan,

191 sudut pengambilan, serta latar belakang yang berbeda agar data yang diperoleh lebih beragam dan model dapat bekerja lebih adaptif. Setelah seluruh citra terkumpul, data dikelompokkan berdasarkan jenis penyakit untuk menjaga keseimbangan jumlah data pada setiap kelas sehingga proses pelatihan model tidak cenderung pada satu kelas tertentu. Selanjutnya, citra yang telah dikumpulkan memasuki tahap pra-pemrosesan guna menyeragamkan format dan kualitas gambar sebelum dilakukan pelabelan serta pelatihan model YOLO-CBAM.

59 4.1.2 Pra-processing Data

209 *Pra-processing* merupakan tahap awal dalam menyiapkan data sebelum
117 digunakan dalam proses pelatihan model. Dataset yang digunakan dalam penelitian ini berjumlah 4.000 citra daun melon yang diperoleh langsung dari pengambilan data lapangan menggunakan kamera smartphone Samsung Galaxy A36. Dataset tersebut terdiri dari empat kelas penyakit, yaitu embun bulu, layu fusarium, mosaik virus, dan bercak daun bersudut, dengan masing-masing kelas berjumlah 1.000 citra. Tahapan pra-processing dilakukan untuk memastikan kualitas dan keseragaman data sebelum digunakan pada proses pelabelan dan pelatihan model YOLO-CBAM, dengan tahapan sebagai berikut.

55 4.1.2.1 Pelabelan dataset Gambar

33 Proses pelabelan bounding box dilakukan menggunakan aplikasi
5 *LabelImg*. Setiap citra daun melon dianotasi dengan *class_id* dan koordinat *bounding box* untuk menandai area penyakit pada daun. Hasil pelabelan kemudian disimpan dalam format YOLO dan digunakan pada proses pelatihan model di Google Colab. Pada tahap ini dilakukan proses anotasi dataset untuk

memberikan informasi mengenai posisi serta jenis penyakit yang terdapat pada daun melon.

Langkah-langkah pelabelan dilakukan sebagai berikut:

1. Membuka Gambar pada *LabelImg*

Setiap citra daun melon dimuat ke dalam aplikasi *LabelImg*. Gambar yang digunakan terdiri dari beberapa jenis penyakit daun melon, yaitu embun bulu, layu fusarium, mosaik virus, dan bercak daun bersudut.

2. Membuat *Bounding Box*

Proses selanjutnya dilakukan dengan menggambar kotak pembatas (*bounding box*) secara manual pada bagian daun yang menunjukkan gejala penyakit. *Bounding box* digunakan untuk menandai area objek yang akan dideteksi oleh model.

3. Menentukan Kelas Objek (*Class ID*)

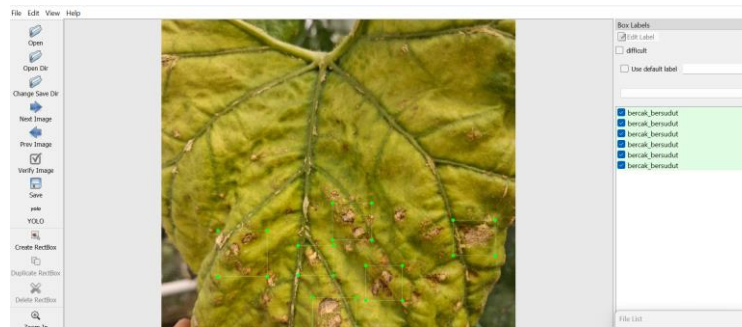
Setelah *bounding box* dibuat, setiap objek diberikan label kelas (*class_id*) sesuai dengan jenis penyakit pada daun melon. Pada penelitian ini, pembagian kelas dilakukan sebagai berikut: kelas 0 untuk penyakit bercak daun bersudut, kelas 1 untuk embun bulu, kelas 2 untuk layu fusarium, dan kelas 3 untuk mosaik virus. Penentuan *class_id* ini bertujuan agar model dapat membedakan setiap jenis penyakit selama proses pelatihan dan pengujian.

4. Menyimpan Label dalam Format YOLO

Hasil anotasi disimpan dalam file berekstensi .txt dengan format YOLO, di mana setiap baris berisi informasi:

class_id x_center y_center width height

Nilai koordinat tersebut telah dinormalisasi terhadap ukuran gambar dalam rentang 0 hingga 1. File label disimpan dengan nama yang sama seperti file gambar dan ditempatkan pada folder label sesuai struktur dataset YOLO sebelum digunakan pada proses pelatihan model di Google Colab.



Gambar 4.1 Proses Labeling Dataset

4.1.2.2 Augmentasi Data

Augmentasi data merupakan tahap untuk menambah variasi citra pada dataset agar model dapat mengenali penyakit daun melon dalam berbagai kondisi. Proses ini dilakukan dengan memodifikasi gambar asli tanpa mengubah informasi utama pada objek, sehingga data pelatihan menjadi lebih beragam dan membantu meningkatkan kemampuan model dalam mendeteksi penyakit secara lebih stabil.

Beberapa teknik augmentasi yang digunakan dalam penelitian ini meliputi:

1. Rotasi

Citra daun melon diberikan variasi rotasi kecil secara acak hingga $\pm 5\%$ untuk membantu model mengenali objek pada berbagai posisi dan sudut pengambilan gambar. Teknik ini membuat model lebih adaptif terhadap perubahan orientasi daun saat proses deteksi penyakit.

2. Flip Horizontal

Teknik flip horizontal dilakukan dengan membalik gambar secara mendatar dengan probabilitas 50%. Augmentasi ini bertujuan agar model tidak hanya mengenali satu arah objek saja, sehingga kemampuan deteksi menjadi lebih fleksibel terhadap perubahan posisi daun.

3. Perubahan Skala dan Translasi

Pada tahap ini, ukuran objek diperbesar atau diperkecil hingga 50% serta posisi objek digeser hingga 10% dari ukuran gambar. Teknik tersebut membantu model mengenali penyakit daun melon meskipun ukuran maupun posisi objek pada citra berbeda-beda.

4. Penyesuaian Warna dan Pencahayaan

Variasi warna dan pencahayaan diterapkan melalui perubahan nilai HSV, yaitu hue sebesar 1,5%, saturation sebesar 70%, dan value sebesar 40%. Teknik ini bertujuan agar model dapat mengenali pola penyakit daun melon pada berbagai kondisi cahaya saat pengambilan gambar di lapangan.

Dengan penerapan berbagai teknik augmentasi tersebut, dataset menjadi lebih variatif sehingga model dapat belajar dari berbagai kondisi citra. Hal ini membuat performa model lebih stabil dan mampu mendeteksi penyakit daun melon dengan lebih baik pada data baru di luar data pelatihan.

4.1.2.3 Normalisasi dan Resize

Sebelum citra daun melon digunakan dalam proses pelatihan model, dilakukan tahap resize dan normalisasi untuk memastikan seluruh data memiliki ukuran serta format yang seragam sehingga dapat diproses dengan baik oleh model. Tahap ini bertujuan meningkatkan efisiensi pelatihan dan membantu model dalam mengenali pola penyakit pada daun melon secara lebih optimal.

1. Resize (Merubah Ukuran Gambar menjadi 640×640 piksel)

Seluruh citra daun melon diubah ukurannya menjadi 640×640 piksel agar sesuai dengan ukuran input yang digunakan pada model. Penyamaan ukuran gambar dilakukan supaya proses ekstraksi fitur berjalan lebih konsisten dan model dapat membaca setiap citra dengan struktur yang sama. Selain itu, penggunaan ukuran 640×640 dipilih karena mampu memberikan keseimbangan antara kualitas citra dan efisiensi komputasi selama proses pelatihan.

2. Normalisasi Nilai Piksel

Setelah proses resize, citra kemudian melalui tahap normalisasi dengan mengubah rentang nilai piksel menjadi skala 0 hingga 1. Proses ini dilakukan secara otomatis oleh *framework* YOLO saat pelatihan model berlangsung. Awalnya, gambar yang diambil dari kamera *smartphone* memiliki nilai warna piksel dalam format RGB dengan rentang angka 0-255.

Contohnya:

- Nilai piksel (0) warna sangat gelap/hitam menjadi 0.0
- Nilai piksel (255) warna sangat terang/putih menjadi 0.1

Nah, YOLO tidak langsung memakai angka besar 0-255 tersebut. Sistem akan mengubahnya menjadi angka yang lebih kecil dalam rentang 0-1 dengan cara membagi semua nilai piksel dengan 255. Normalisasi bertujuan membantu model memproses data secara lebih stabil, mempercepat proses pembelajaran, serta meningkatkan performa model dalam mendeteksi penyakit daun melon.

4.1.3 Pengolahan Data

4.1.3.1 Struktur Pemisahan Folder Dataset

Dataset yang digunakan dalam penelitian ini disusun dalam struktur folder yang terorganisasi untuk mendukung proses pelatihan model. Dataset terdiri dari 4.000 citra daun melon yang diperoleh melalui pengambilan data lapangan dan dikelompokkan ke dalam empat kategori penyakit. Setiap kategori memiliki jumlah data yang sama, yaitu 1.000 citra, sehingga distribusi data lebih seimbang dan dapat membantu model mempelajari karakteristik setiap penyakit dengan baik. Struktur folder yang teratur juga memudahkan proses pengelolaan data, pelabelan, serta pembagian dataset ke dalam data pelatihan (*training*), validasi (*validation*), dan pengujian (*testing*) sebelum digunakan pada proses pelatihan model. Masing-masing kombinasi disimpan dalam folder terpisah. Berikut adalah struktur folder awal:

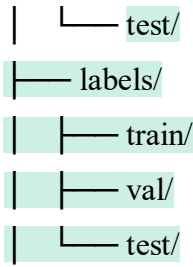
dataset/

- |— Bercak_bersudut/
- |— Embun_bulu/
- |— Layu_fusarium/
- |— Mosaik_Virus/

Dataset kemudian dibagi menjadi tiga subset: train (70%), validation (20%), dan test (10%), dengan distribusi kelas yang seimbang agar tidak terjadi class imbalance. Struktur direktori pelatihan model disusun sebagai berikut:

dataset/

- |— images/
 - |— train/
 - |— val/



Folder images berisi file gambar (.jpg, .png, dll) yang sudah dikelompokkan berdasarkan pembagian train, validation, dan test dan Folder labels berisi file anotasi dalam format .txt, di mana setiap file memiliki nama yang sama dengan file gambar dan berisi informasi bounding box beserta ID kelas dari objek daun melon yang terdapat pada gambar tersebut.

4.1.3.2 Penetapan Label Kelas (class_id)

Model YOLO membutuhkan label dalam bentuk angka (numerik), sehingga setiap kelas dikodekan menjadi class ID yang konsisten. Penetapan class ID juga dicantumkan dalam file classes.txt yang digunakan dalam konfigurasi pelatihan. Berikut adalah daftar penetapan class ID untuk setiap kelas:

Tabel 4.1 Pemberian class_id

class_id	Label
0	bercak bersudut
1	embun bulu
2	layu fusarium
3	mosaik virus

Tabel 4.1 menunjukkan pemberian class_id yang digunakan pada proses pelabelan dataset penyakit daun melon. Setiap jenis penyakit diberikan nilai numerik yang berbeda untuk memudahkan model dalam mengenali dan membedakan objek selama proses pelatihan dan pengujian. Pada penelitian ini, class_id 0 digunakan untuk penyakit bercak daun bersudut, class_id 1 untuk

embun bulu, *class_id* 2 untuk layu fusarium, dan *class_id* 3 untuk mosaik virus. Pemberian *class_id* ini menjadi acuan dalam proses anotasi menggunakan *LabelImg* serta digunakan oleh model untuk melakukan klasifikasi dan deteksi penyakit pada daun melon.

4.1.3.3 Format Anotasi (Yolo Format)

Setiap file label disimpan dalam *format .txt* dengan nama yang sama seperti file gambar. Format isi file mengikuti standar YOLO sebagai berikut:

`<class_id> <x_center> <y_center> <width> <height>`

Keterangan:

- *class_id*: nomor kelas (0-3)
- *x_center*, *y_center*: titik tengah *bounding box*
- *width*: lebar *bounding box* yang mengelilingi objek. Nilainya dinormalisasi terhadap lebar gambar.
- *height*: tinggi *bounding box* yang mengelilingi objek. Nilainya dinormalisasi terhadap tinggi gambar.

File	Edit	View
0 0.51896875 0.31696875 0.8102656250000001 0.56696875		

Gambar 4.2 Hasil Pelabelan

4.1.3.4 Jumlah Dataset

Dataset yang digunakan dalam penelitian ini sebanyak 4.000 citra gambar daun melon yang sudah di klasifikasikan ke dalam 4 kelas penyakit daun. Masing-masing kelas memiliki jumlah gambar sebanyak 1.000 citra daun, dengan distribusi data yang seimbang. Hal tersebut bertujuan untuk menghindari ketidak

seimbangan terhadap setiap kelas, yang dapat mempengaruhi performa kinerja model. Berikut adalah distribusi dataset perkelas:

Tabel 4.2 Distribusi Dataset Gambar Penyakit Daun Melon

No	Jenis Penyakit	Jumlah Gambar	Ciri-ciri Penyakit
1	Bercak Bersudut	1000	Bercak bersudut, menguning, daun berlubang.
2	Embun Bulu	1000	Terdapat lapisan atau bercak putih seperti tepung pada permukaan daun.
3	Layu Fusarium	1000	Daun layu, menguning.
4	Mosaik Virus	1000	Daun belang-belang, pucat, pertumbuhan terhambat.

Tabel 4.2 menunjukkan distribusi dataset citra penyakit daun melon yang digunakan dalam penelitian. Dataset terdiri atas empat kategori penyakit, yaitu bercak bersudut, embun bulu, layu fusarium, dan mosaik virus. Masing-masing kategori memiliki jumlah data yang sama, yaitu sebanyak 1.000 gambar, sehingga total keseluruhan dataset yang digunakan mencapai 4.000 gambar.

Setiap kategori penyakit dilengkapi dengan karakteristik visual yang berbeda sebagai dasar dalam proses pelabelan dan pelatihan model. Penyakit bercak bersudut ditandai dengan munculnya bercak-bercak pada permukaan daun yang sering disertai perubahan warna menjadi kekuningan dan kerusakan jaringan daun. Embun bulu memiliki ciri khas berupa lapisan atau bercak putih menyerupai tepung pada permukaan daun. Layu fusarium ditandai dengan kondisi daun yang layu dan menguning akibat gangguan pada sistem pembuluh tanaman. Sementara itu, mosaik virus menunjukkan gejala berupa pola belang-belang pada daun, warna yang tampak pucat, serta pertumbuhan tanaman yang terhambat.

Jumlah data yang relatif besar dan distribusi yang seimbang pada setiap kelas memungkinkan model untuk mempelajari karakteristik masing-masing penyakit

secara lebih optimal. Selain itu, variasi kondisi citra, seperti perbedaan pencahayaan, sudut pengambilan gambar, tekstur daun, dan latar belakang, dapat membantu meningkatkan kemampuan model dalam mengenali pola penyakit pada data baru. Dengan demikian, model diharapkan mampu menghasilkan performa deteksi yang lebih akurat dan stabil ketika diterapkan pada kondisi lapangan yang sebenarnya.

4.1.4 Pengembangan Model Arsitektur YOLOv8n dan YOLOv8n+CBAM

Model yang digunakan dalam penelitian ini adalah YOLOv8n (*nano*), yaitu salah satu varian dari keluarga YOLO (*You Only Look Once*) yang dikembangkan oleh Ultralytics. YOLOv8n merupakan model deteksi objek berbasis *Convolutional Neural Network (CNN)* yang dirancang untuk melakukan deteksi objek secara cepat dan efisien. Pada penelitian ini, YOLOv8n digunakan sebagai model dasar untuk mendeteksi penyakit pada daun melon berdasarkan citra yang telah dilabeli.

Selain model dasar YOLOv8n, penelitian ini juga mengembangkan model YOLOv8n dengan penambahan *Convolutional Block Attention Module (CBAM)*. CBAM merupakan mekanisme perhatian (*attention mechanism*) yang terdiri dari *Channel Attention* dan *Spatial Attention*. Penambahan CBAM bertujuan untuk meningkatkan kemampuan model dalam mengekstraksi fitur-fitur penting pada citra daun, seperti bercak penyakit, perubahan warna, dan pola kerusakan jaringan daun.

Dengan adanya modul CBAM, model diharapkan mampu memberikan fokus yang lebih baik pada area yang mengandung gejala penyakit sehingga dapat meningkatkan performa deteksi dibandingkan dengan model YOLOv8n standar. Selanjutnya, kedua model akan dilatih menggunakan dataset yang sama dan

dibandingkan berdasarkan nilai *precision*, *recall*, *mAP*, serta akurasi untuk mengetahui pengaruh penambahan CBAM terhadap kinerja model.

4.1.4.1 Arsitektur YOLO

YOLOv8n (You Only Look Once version 8 nano) merupakan salah satu varian terkecil dari keluarga YOLOv8 yang dikembangkan oleh *Ultralytics*. Model ini dirancang untuk menghasilkan performa deteksi objek yang cepat dengan jumlah parameter yang relatif sedikit, sehingga cocok digunakan pada perangkat dengan sumber daya komputasi terbatas. Model ini sangat sesuai untuk digunakan dalam skenario waktu nyata (*real-time*), seperti dalam pendeteksian penyakit pada daun melon yang dilakukan pada penelitian ini.

a. Input Model

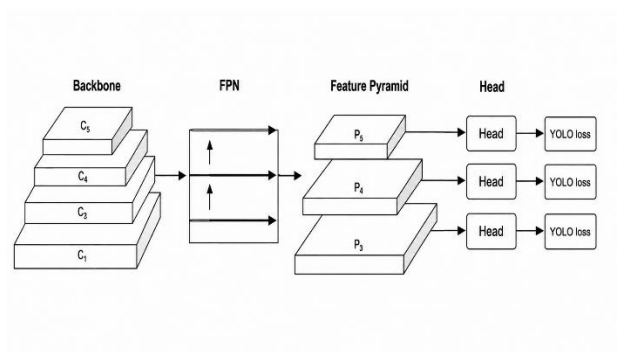
Input dari model berupa citra digital yang telah dikumpulkan dan disimpan dalam format gambar. Sebelum digunakan pada proses pelatihan, seluruh citra terlebih dahulu melalui tahap pre-processing untuk memastikan keseragaman data dan kesesuaian dengan kebutuhan model. Pada penelitian ini, gambar diubah ukurannya (*resize*) menjadi 640×640 piksel. Selain itu, diterapkan teknik augmentasi data yang meliputi rotasi, flip, perubahan skala (*scaling*), translasi, serta penyesuaian warna dan pencahayaan. Setelah melalui tahap tersebut, citra siap digunakan sebagai input pada arsitektur YOLOv8. Secara umum, arsitektur YOLOv8 terdiri dari tiga komponen utama, yaitu *Backbone*, *Neck*, dan *Head*.

1. Backbone Berfungsi untuk mengekstraksi fitur-fitur penting dari gambar input. Backbone terdiri dari beberapa lapisan konvolusi (*Conv Block*) bertingkat (C1 hingga C5) yang secara bertahap menangkap informasi visual mulai dari bentuk sederhana hingga pola kompleks.

2. **Neck** berfungsi untuk menggabungkan fitur yang diperoleh dari berbagai tingkat kedalaman pada Backbone. YOLOv8n menggunakan mekanisme *Feature Pyramid Network (FPN)* dan *Path Aggregation Network (PAN)* untuk menghasilkan feature map multiskala. Penggabungan fitur ini memungkinkan model mendeteksi objek dengan ukuran yang berbeda-beda, baik objek kecil, sedang, maupun besar.

3. **Head** adalah bagian yang menghasilkan output akhir. Fitur yang telah diolah oleh FPN akan diproses di sini untuk menghasilkan prediksi berupa:

- **Bounding box**: Koordinat lokasi objek pada gambar,
- **Confidence score**: Tingkat keyakinan model terhadap prediksi objek,
- **Class label**: Kelas atau kategori objek yang dikenali.



Gambar 4.3 Arsitektur YOLOv8

Pada Gambar 4.3 ditampilkan struktur lengkap dari arsitektur YOLO yang digunakan dalam penelitian ini. Model menerima input berupa gambar digital yang diproses melalui lapisan *Backbone*, yang tersusun dari serangkaian *Conv Block* untuk mengekstraksi fitur visual secara bertahap, mulai dari fitur beresolusi tinggi dengan informasi detail (C1 dan C2) hingga fitur beresolusi rendah dengan informasi semantik yang lebih dalam (C3, C4, dan C5).

Fitur-fitur dari tahap C3, C4, dan C5 kemudian disalurkan ke bagian FPN (*Feature Pyramid Network*), yang berfungsi menggabungkan fitur dari beberapa skala (*multi-scale features*). Pada tahap ini, fitur semantik dari level yang lebih dalam (C5) digabungkan secara bertahap dengan fitur resolusi tinggi dari level yang lebih dangkal (C4 dan C3), sehingga menghasilkan tiga level fitur *pyramid*, yaitu P3, P4, dan P5. Penggabungan multi-skala ini memungkinkan model untuk mendeteksi objek dengan ukuran yang beragam, baik objek kecil (dideteksi pada P3), objek berukuran menengah (P4), maupun objek berukuran besar (P5).

Selanjutnya, setiap level fitur *pyramid* (P3, P4, P5) diproses oleh bagian *Head* secara terpisah untuk menghasilkan prediksi akhir berupa tiga output utama, yaitu label kelas objek (*class*), posisi objek (*bounding box*), dan tingkat keyakinan (*confidence score* atau *objectness*).

Berdasarkan arsitektur tersebut, model menghasilkan tiga jenis output utama. Output pertama berupa *bounding box*, yang menunjukkan lokasi area daun yang terindikasi penyakit dalam bentuk persegi panjang pada gambar. Output kedua berupa *class label*, yang menentukan jenis penyakit pada daun melon yang terdeteksi, misalnya bercak daun bersudut, embun bulu, layu fusarium, mosaik virus, dan sebagainya. Output ketiga berupa *confidence score*, yaitu nilai probabilitas antara 0 sampai 1 yang menunjukkan keyakinan model terhadap klasifikasi penyakit tersebut.

4.1.4.2 Arsitektur YOLOv8n+CBAM

Arsitektur YOLOv8 yang telah dimodifikasi dengan penambahan modul CBAM (*Convolutional Block Attention Module*). Modul CBAM disisipkan setelah

proses ekstraksi fitur pada *backbone*, tepatnya pada setiap level fitur C3, C4, dan C5, sebelum fitur tersebut diteruskan ke bagian *neck* (*FPN/PAN*). CBAM terdiri dari dua mekanisme attention yang bekerja secara berurutan, yaitu *Channel Attention* dan *Spatial Attention*. Penambahan modul CBAM bertujuan untuk meningkatkan fokus model terhadap fitur penting, mengingat gejala penyakit pada daun melon sering berukuran kecil dan menyerupai area daun sehat. CBAM membantu model memberi bobot lebih besar pada *channel* dan area *spasial* yang relevan, sehingga akurasi deteksi meningkat tanpa menambah beban komputasi yang signifikan.

a. Modifikasi Arsitektur YOLOv8n dengan CBAM

Arsitektur YOLOv8n pada penelitian ini dimodifikasi dengan menyisipkan modul CBAM (*Convolutional Block Attention Module*) pada bagian keluaran *backbone*, tepatnya sebelum fitur diteruskan ke tahap *neck* (*FPN/PAN*). Modifikasi ini bertujuan agar model lebih fokus pada fitur-fitur penting yang merepresentasikan gejala penyakit pada daun (seperti bercak, embun, atau perubahan warna) dan menekan fitur latar belakang yang tidak relevan.

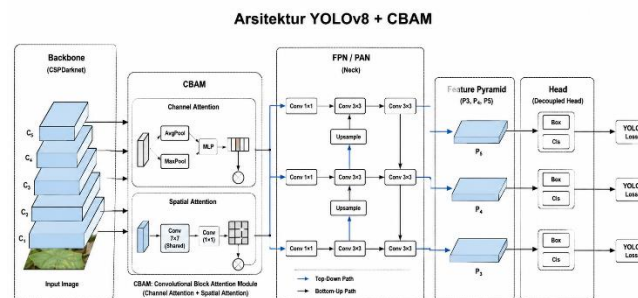
Pada arsitektur YOLOv8n standar, *backbone CSPDarknet* menghasilkan lima level fitur (C1-C5) dari resolusi tinggi ke rendah. Pada model yang dikembangkan, setiap level fitur (C1-C5) yang akan digunakan dilewatkan terlebih dahulu ke dalam modul CBAM sebelum masuk ke tahap *neck*. Dengan penambahan ini, setiap feature map diperkaya dengan informasi *atensi kanal* (*channel attention*) dan *atensi spasial* (*spatial attention*) sebelum dilakukan proses penggabungan fitur multi-skala pada *FPN/PAN*.

Alasan pemilihan posisi penyisipan CBAM setelah backbone adalah:

1. Fitur keluaran *backbone* masih bersifat *raw feature* sehingga penguatan atensi pada tahap ini akan diteruskan dan diperkuat lagi pada proses fusion di neck.
2. Penyisipan setelah *backbone* tidak mengubah struktur internal backbone, sehingga bobot *pre-trained* pada *backbone* tetap dapat dimanfaatkan (*transfer learning*).
3. Beban komputasi tambahan relatif kecil karena CBAM merupakan modul lightweight.

b. Struktur Arsitektur YOLOv8n+CBAM

Secara umum, arsitektur YOLOv8n+CBAM yang dibangun terdiri atas lima komponen utama yaitu:



Gambar 4.4 Arsitektur Yolo+CBAM

Alur kerja arsitektur dapat dijabarkan sebagai berikut:

1. Tahap *Backbone*: Citra input diproses oleh backbone menghasilkan lima feature map bertingkat (C_1 , C_2 , C_3 , C_4 , C_5), masing-masing dengan resolusi yang semakin mengecil namun representasi semantik yang semakin tinggi.
2. Tahap CBAM: Setiap feature map yang akan digunakan pada tahap selanjutnya dilewatkan ke modul CBAM, yang terdiri atas dua sub-modul berurutan, yaitu *Channel Attention* dan *Spatial Attention*.

3. Tahap *Neck (FPN/PAN)*: Fitur hasil CBAM diproses melalui:

- Jalur *Top-Down* (ditandai garis biru): fitur level dalam (semantik tinggi, resolusi rendah) di-*upsample* lalu digabungkan (*concat*) dengan fitur level dangkal melalui operasi $Conv\ 1\times 1 \rightarrow Concat \rightarrow Conv\ 3\times 3$.
- Jalur *Bottom-Up* (ditandai garis hitam): fitur hasil *top-down* diperkuat kembali dengan informasi detail spasial dari level resolusi tinggi.

4. Tahap *Feature Pyramid*: Proses *neck* menghasilkan tiga *output feature map* yaitu P3 (resolusi tinggi, untuk objek kecil), P4 (resolusi sedang), dan P5 (resolusi rendah, untuk objek besar).

5. Tahap *Head (Decoupled Head)*: Setiap level P3, P4, P5 diproses oleh dua cabang terpisah:

- Cabang Box: memprediksi koordinat bounding box.
- Cabang Cls: memprediksi probabilitas kelas objek.

c. Output Model

Keluaran akhir dari arsitektur YOLOv8n+CBAM berupa hasil deteksi objek pada citra daun, dengan setiap objek terdeteksi ditampilkan dalam bentuk *bounding box* yang dilengkapi label kelas penyakit beserta tingkat kepercayaan (*confidence score*).

4.1.5 Pelatihan dan Evaluasi Model

Pelatihan dan evaluasi model merupakan tahap yang penting karena menentukan keberhasilan sistem dalam mengenali detail penyakit pada daun melon berdasarkan gambar. Model yang digunakan adalah YOLOv8n + CBAM, model ini

dilatih menggunakan dataset yang sudah di kumpulkan dengan pengambilan langsung di lapangan sebanyak 1000 per kelas.

a. Proses Pelatihan Model

Pelatihan model dilakukan menggunakan perangkat laptop dengan sistem operasi windows. Proses pelatihan menggunakan Bahasa pemograman *Python* dan *Framework Ultralytics YOLOv8*. Konfigurasi pelatiha model disajikan pada table berikut:

Tabel 4.3 Parameter Pelatihan Model YOLOv8 + CBAM

Parameter	Nilai
Epoch	100
Batch Size	16
Image Size	640x640 piksel
Optimizer	AdamW
Dataset Split	70:20:10

Tabel 4.3 Parameter model ini dirancang untuk mengoptimalkan kinerja model dalam mengenali jenis penyakit daun melon, pelatihan dilakukan selama 100 epoch yang berarti model akan memproses seluruh data pelatihan sebanyak 100 kali untuk memperkuat generalisasi. Ukuran batch yang digunakan adalah 16, artinya dalam setiap iterasi pelatihan model memproses 16 gambar ssekaligus. Seluruh gambar diubah ukurannya menjadi 640x640 piksel, ukuran yang sesuai dengan standar input pada YOLOv8. Sebagai algoritma optimasi, digunakan *AdamW (Adaptive Moment Estimation with Weight Decay)*, yaitu optimizer yang memperbarui bobot model secara adaptif berdasarkan nilai loss untuk meningkatkan proses pembelajaran model. Selain itu, dataset dibagi menjadi tiga subset dengan rasio 70:20:10, yakni 70% untuk data pelatihan (*train*), 20% untuk

data validasi (*validation*), dan 10% untuk data pengujian (*test*). Perintah yang digunakan pada terminal Google colab untuk menjalankan pelatihan model yaitu:

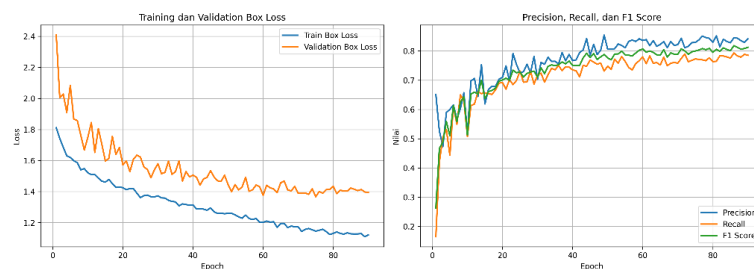
```
yolo task=detect mode=train model=yolov8n_cbam.yaml data=data.yaml  
epochs=100 imgsz=640 batch=16 optimizer=AdamW
```

Hasil dari proses pelatihan model yaitu:

- File *best.pt* yang berisi bobot model terbaik dari hasil training,
- File *last.pt* bobot model pada *epoch* terakhir proses pelatihan,
- Grafik performa model (*mAP*, *loss*, *precision*, *recall*) yang otomatis tersimpan.

b. Hasil Pelatihan Model

Untuk mengetahui kinerja model selama proses pelatihan, dilakukan pemantauan terhadap perubahan nilai *loss*, *precision*, *recall* dan *F1-score* pada setiap *epoch*. Visualisasi perkembangan matrik tersebut ditampilkan pada gambar 4.5 sebagai dasar untuk mengevaluasi proses pembelajaran model YOLOv8n+CBAM.



Gambar 4.5 Grafik Training, Validation Box Loss, Precision, Recal dan F1 Score.

Berdasarkan Gambar 4.5 proses pelatihan model YOLOv8n+CBAM menunjukkan hasil yang baik. Grafik *Training Box Loss* dan *Validation Box Loss*

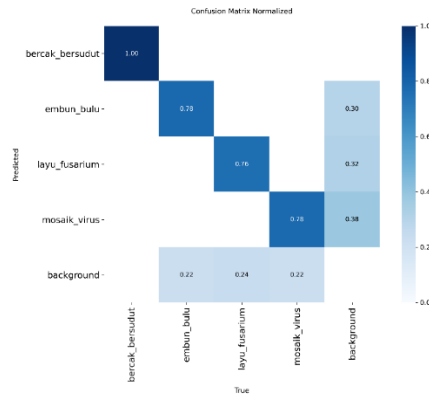
mengalami penurunan secara bertahap hingga akhir pelatihan, yang menandakan bahwa model semakin mampu mempelajari pola penyakit daun melon. Sementara itu, nilai *Precision*, *Recall*, dan *F1-Score* terus meningkat dan mulai stabil pada *epoch* akhir, menunjukkan peningkatan kemampuan model dalam mendeteksi objek secara akurat. Secara keseluruhan, tren penurunan *loss* dan peningkatan metrik evaluasi menunjukkan bahwa proses pelatihan berlangsung secara konvergen dan menghasilkan model dengan kemampuan generalisasi yang baik.

c. Evaluasi Model

Subbab ini menyajikan hasil evaluasi model YOLOv8n+CBAM yang telah dilatih untuk mendeteksi penyakit daun melon. Evaluasi dilakukan berdasarkan hasil pelatihan setiap *epoch*, grafik *loss*, *precision*, *recall*, dan *F1-score*, serta metrik *Precision*, *Recall*, *mAP50*, dan *mAP50-95*. Hasil deteksi pada data uji juga ditampilkan untuk menunjukkan performa model dalam mengenali setiap kelas penyakit.

1. Confusional Matrik

Confusional Matrik digunakan untuk mengevaluasi kinerja deteksi objek dengan membandingkan hasil prediksi model terhadap label yang sebenarnya pada dataset validasi. Matriks ini menyajikan jumlah prediksi yang benar (*tru positives*), jumlah prediksi yang salah (*false positives*), serta objek yang gagal terdeteksi (*false negatives*) untuk setiap kelas objek yang diprediksi. Berikut adalah *hasil Confusion Matrik* dapat dilihat pada Gambar 4.6.



Gambar 4.6 Confusion Matrik

Untuk akurasi *Confusion Matrik* dilakukan dengan menerapkan perhitungan persamaan 1 pada rumus berikut[24]:

		Actual Values	
		Positive (1)	Negative (0)
Predicted Values	Positive (1)	TP	FP
	Negative (0)	FN	TN

Gambar 4.7 Perhitungan Confusio Matrik

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \times 100\%$$

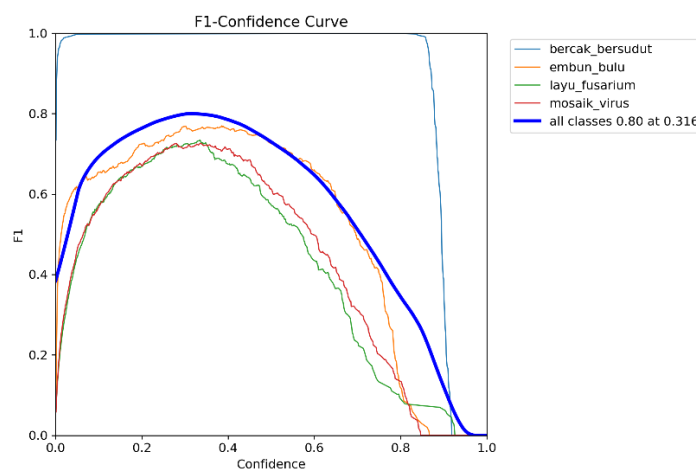
$$Accuracy = \frac{100 + 78 + 76 + 78}{22 + 24 + 22 + 100 + 78 + 76 + 78}$$

$$= \frac{332}{400} = 0,83$$

$$= 0,83 \times 100\% = 83\%$$

2. F1-Confidence Curve

F1-Confidence Curve adalah grafik yang menunjukkan hubungan antara nilai F1-score dengan confidence yang digunakan oleh model dalam proses deteksi. F1-score adalah matrik yang menggabungkan nilai precision dan recall, sehingga dapat menggambarkan keseimbangan kemampuan model dalam mendeteksi objek secara tepat sekaligus menemukan seluruh objek yang seharusnya terdeteksi. Melalui kurva ini dapat diketahui pada tingkat confidence berapa model menghasilkan performa terbaik. Nilai threshold yang dipilih dari kurva tersebut dapat digunakan sebagai acuan untuk memperoleh keseimbangan antara precision dan recall, sehingga model mampu memberikan hasil deteksi yang lebih optimal sesuai dengan kebutuhan penelitian.



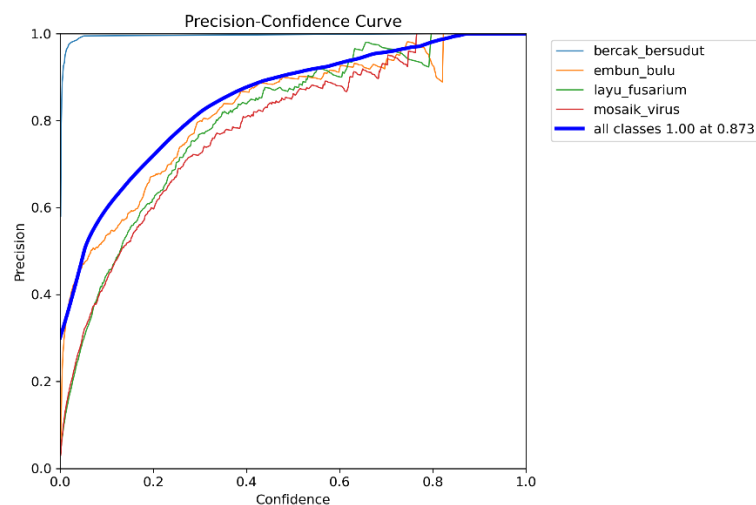
Gambar 4.8 F1-Confidence Curve

Grafik pada gambar 4.8 F1-Confidence Curve menunjukkan hubungan antara nilai confidence dengan nilai F1-score. Berdasarkan grafik, model memperoleh F1-score terbaik sebesar 0,80 pada confidence threshold 0,316. Hal ini menunjukkan bahwa pada nilai threshold tersebut keseimbangan antara precision

dan *recall* berada pada kondisi optimal. Kurva kelas bercak_bersudut memiliki nilai F1 tertinggi dibandingkan kelas lainnya.

3. Precision-Confidence Curve

Precision-Confidence Curve merupakan kurva yang menunjukkan hubungan antara nilai *precision* dan *confidence threshold* pada model deteksi. Kurva ini digunakan untuk melihat perubahan tingkat ketepatan prediksi ketika nilai *confidence* dinaikkan atau diturunkan. Melalui kurva ini, dapat ditentukan nilai *threshold* yang paling sesuai agar model menghasilkan prediksi dengan tingkat ketepatan yang optimal.

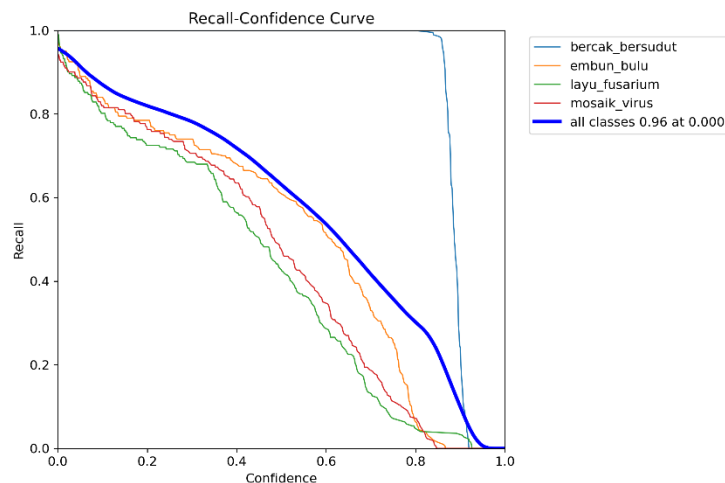


Gambar 4.9 Precision-Confidence Curve

Grafik *Precision-Confidence Curve* menunjukkan perubahan nilai *precision* terhadap *confidence threshold*. Terlihat bahwa nilai *precision* meningkat seiring bertambahnya *confidence threshold*. Pada seluruh kelas, model mencapai *precision* sebesar 1,00 pada *confidence* sekitar 0,873, yang artinya hampir seluruh prediksi yang dihasilkan benar.

4. Recall-Confidence Curve

Recall-Confidence Curve menggambarkan kemampuan model dalam menemukan seluruh objek pada berbagai nilai *confidence threshold*. Grafik ini menunjukkan bahwa perubahan nilai *confidence* memengaruhi jumlah objek yang berhasil dideteksi. Semakin tinggi *threshold* yang digunakan, model menjadi lebih selektif sehingga nilai *recall* umumnya menurun karena sebagian objek tidak lagi terdeteksi.



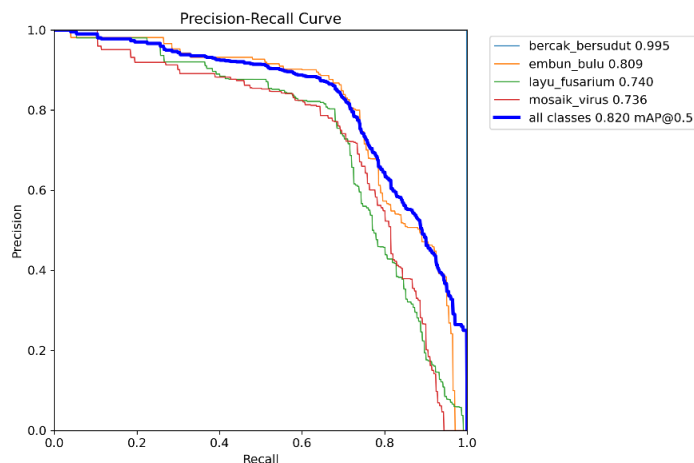
Gambar 4.10 Recall-Confidence Curve

Gambar di atas menunjukkan grafik nilai *recall* terhadap nilai *confidence*. Dari hasil pelatihan yang diperoleh, nilai *recall* mencapai rata-rata maksimal sebesar 0,96 pada nilai *confidence* 0,00. Ini menunjukkan bahwa pada tingkat keyakinan yang sangat rendah (0,0), model mampu mendeteksi hampir seluruh objek penyakit daun melon.

5. Precision-Recall Curve

Precision-Recall Curva merupakan grafik yang menunjukkan hubungan antara *precision* dan *recall* pada berbagai nilai *confidence threshold*. *Precision*

menggambarkan tingkat ketepatan model dalam mendeteksi, sedangkan *recall* menunjukkan kemampuan model dalam menemukan seluruh objek yang ada pada gambar. Melalui grafik ini, dapat diketahui keseimbangan antara *precision* dan *recall* sehingga dapat ditentukan nilai *threshold* yang memberikan performa deteksi paling optimal.



Gambar 4.11 Precision-Recall Curve

Gambar diatas menunjukkan hubungan antara nilai *precision* dan *recall* pada model yang telah dilatih. Semakin mendekati sudut kanan atas, maka semakin baik kemampuan model dalam mendeteksi. Berdasarkan hasil pengujian, model memperoleh nilai *mAP@0,5* sebesar 0,820 (82,0%) untuk seluruh kelas, yang menunjukkan bahwa model memiliki kemampuan deteksi yang baik.

Berdasarkan kurva setiap kelas, penyakit bercak_bersudut memiliki performa terbaik dengan *mAP* sebesar 0,995, diikuti embun_bulu sebesar 0,809, layu_fusarium sebesar 0,740, dan mosaik_virus sebesar 0,736. Perbedaan nilai tersebut menunjukkan bahwa model lebih mudah mengenali penyakit bercak_bersudut dibandingkan tiga kelas penyakit lainnya. Secara keseluruhan, hasil *Precision-Recall Curve* menunjukkan bahwa model mampu memberikan

keseimbangan yang baik antara ketepatan deteksi (*precision*) dan kemampuan menemukan seluruh objek penyakit (*recall*), sehingga model layak digunakan untuk proses deteksi penyakit daun.

d. Perbandingan dengan *You Only Look Once (YOLOv7)*

Sebagai model pembanding, penelitian ini menggunakan YOLOv7 untuk mengetahui perbedaan performa deteksi terhadap model YOLOv8 yang telah ditambahkan mekanisme *Convolutional Block Attention Module (CBAM)*. Proses pelatihan dan pengujian dilakukan menggunakan dataset dan pembagian data yang sama agar hasil evaluasi dapat dibandingkan secara objektif. Evaluasi dilakukan menggunakan matrik *precision*, *recall*, *mAP@50*, dan *mAP@50-95*. Hasil evaluasi model YOLOv7 pada data uji disajikan pada Tabel 4.4 Perbandingan kinerja model.

Tabel 4.4 Perbandingan Kinerja Model

Matrik	YOLOv7-tiny	YOLOv8+CBAM
Precision	57,30%	82,92%
Recall	49,20%	77,32%
F1-score	52,93%	79,99%
mAP@0.5	52,20%	82,00%
mAP@0.5:0.95	21,50%	50,52%

Berdasarkan Tabel 4.4, model YOLOv8+CBAM menunjukkan performa yang lebih baik dibandingkan model YOLOv7 pada seluruh matrik evaluasi. Model YOLOv7 memperoleh nilai *precision* sebesar 57,30%, *recall* sebesar 49,20%, *F1-score* sebesar 52,93%, *mAP@0.5* sebesar 52,20%, dan *mAP@0.5:0.95* sebesar 21,50%.

Peningkatan performa tersebut menunjukkan bahwa penambahan modul CBAM (*Convolutional Block Attention Modul*) mampu meningkatkan kemampuan model dalam mengekstraksi fitur penting pada citra daun melon. *Mekanisme attention* pada CBAM membantu model memberikan perhatian lebih pada informasi yang relevan, baik pada dimensi kanal (*channel attention*) maupun lokasi objek (*spatial attention*), sehingga proses deteksi menjadi lebih akurat. Dengan demikian, model YOLOv8+CBAM terbukti memiliki kinerja yang lebih baik dibandingkan model YOLOv7 dalam mendeteksi penyakit daun melon.

Tabel 4.5 Perbandingan Nilai mAP@0.5 per Kelas Penyakit

Kelas	YOLOv7-tiny	YOLOv8n+CBAM
Bercak Bersudut	36,60%	99,50%
Embun Bulu	58,90%	80,90%
Layu Fusarium	56,60%	74,00%
Mosaik Virus	56,90%	73,60%

Berdasarkan Tabel 4.5, model YOLOv8+CBAM memperoleh $mAP@0.5$ yang lebih tinggi pada seluruh kelas penyakit dibandingkan model YOLOv7. Peningkatan paling signifikan terjadi pada kelas bercak bersudut, yaitu dari 36,60% menjadi 99,50%, yang menunjukkan bahwa penambahan CBAM mampu meningkatkan kemampuan model dalam mengenali karakteristik penyakit yang sebelumnya sulit dibedakan. Pada kelas embun bulu, nilai $mAP@0.5$ meningkat dari 58,90% menjadi 80,90%. Sementara itu, pada kelas layu fusarium dan mosaik virus, nilai $mAP@0.5$ juga mengalami peningkatan menjadi 74,00% dan 73,60%.

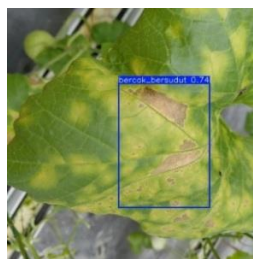
Secara keseluruhan, hasil tersebut menunjukkan bahwa penambahan modul CBAM memberikan kontribusi positif terhadap kemampuan model dalam memfokuskan perhatian pada fitur-fitur penting dari setiap kelas penyakit,

sehingga meningkatkan akurasi deteksi dibandingkan model YOLOv7 tanpa mekanisme attention. Hal ini memperkuat bahwa model YOLOv8 + CBAM merupakan model yang lebih efektif untuk mendeteksi penyakit daun melon pada penelitian ini.

f. Test Model

Setelah proses pelatihan selesai dan diperoleh model terbaik (*best.pt*), dilakukan pengujian menggunakan citra daun melon di luar dataset pelatihan dan validasi. Pengujian dilakukan melalui Google Colab dengan memanggil model menggunakan fungsi predict pada *Ultralytics YOLOv8* untuk mengetahui kemampuan model dalam mendeteksi penyakit daun melon pada data baru. Berikut adalah hasil-hasil nya:

1. Bercak Bersudut



Gambar 4.12 Hasil Deteksi Bercak Bersudut

2. Embun Bulu



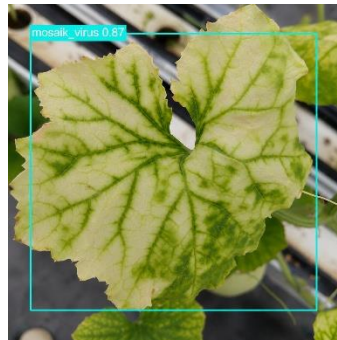
Gambar 4.13 Hasil Deteksi Embun Bulu

3. Layu Fusarium



Gambar 4.14 Hasil Deteksi Layu Fusarium

4. Mosaik Virus



Gambar 4.15 Hasil Deteksi Mosaik Virus

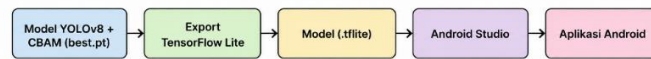
4.1.6 Implementasi Sistem

Setelah proses pelatihan dan evaluasi model klasifikasi penyakit daun melon selesai, langkah selanjutnya adalah mengimplementasikan model tersebut ke dalam aplikasi android. Implementasi ini bertujuan untuk mempermudah pengguna dan mendeteksi penyakit daun melon secara langsung melalui perangkat seluler.

a. Integrasi Model Yolo ke Android

Setelah model selesai dilatih dan memberikan hasil evaluasi yang memuaskan, langkah selanjutnya adalah mengintegrasikan model tersebut kedalam aplikasi android. Proses integrasi ini diawali dengan konversi model yang telah disimpan

dalam format *best.pt* menjadi format *.tflite* menggunakan *TransFlow Lite Converter*. Format *.tflite* lebih ringan dan dirancang khusus untuk digunakan dalam perangkat mobile seperti Android, sehingga lebih efisien dan cepat dijalankan.



Gambar 4.16 Alur Integrasi Model YOLO ke Aplikasi Android

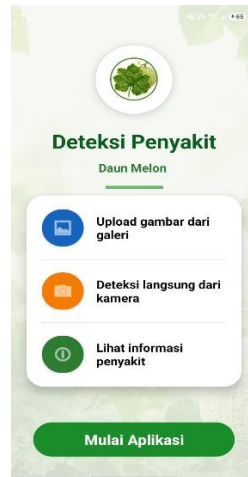
Gambar 4.16 berikut menampilkan alur proses integrasi model dari tahap pelatihan hingga dapat digunakan pada aplikasi android. Model yang telah dikonversi ke *.tflite* kemudian dimasukkan ke dalam folder *assets/* di Aplikasi Studio. Selanjutnya, dilakukan proses pemanggilan model dan inferensi hasil deteksi di dalam kode aplikasi menggunakan *API TensorFlow Lite*.

b. Fitur dan Tampilan Aplikasi

Aplikasi Android yang dikembangkan dirancang dengan antarmuka yang sederhana dan mudah dipahami, untuk memudahkan pengguna dalam mendeteksi penyakit daun melon secara cepat. Berikut adalah tampilan-tampilan utama dalam aplikasi:

1. Tampilan Awal

Ketika aplikasi pertama kali dijalankan, pengguna akan di tampilkan halaman awal untuk memulai aplikasi menuju ke halaman selanjutnya.



Gambar 4.17 Tampilan Awal Aplikasi Android

2. Tampilan Info Petunjuk Penggunaan

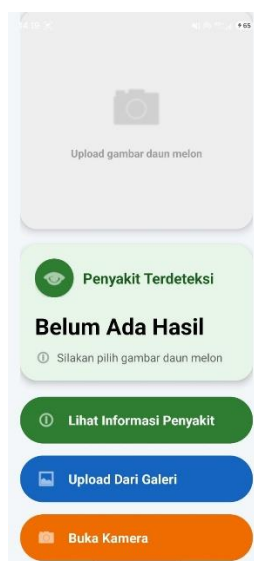
Halaman petunjuk penggunaan aplikasi yang berfungsi sebagai panduan awal bagi pengguna sebelum melakukan proses deteksi penyakit daun melon. Halaman ini dirancang dengan tampilan yang sederhana dan mudah dipahami sehingga pengguna dapat mengetahui tahapan pengguna aplikasi secara berurutan.



Gambar 4.18 Petunjuk Penggunaan Aplikasi Android

3. Tampilan Halaman Utama

Halaman utama merupakan pusat utama interaksi antara pengguna dengan sistem deteksi penyakit daun melon. Pada halaman ini, pengguna dapat langsung menggunakan fitur-fitur utama aplikasi. Tampilan yang minimalis dan intuitif dibuat agar mudah digunakan oleh semua kalangan pengguna, termasuk masyarakat umum.



Gambar 4 19 Halaman Utama Aplikasi Android

Fitur-fitur yang tersedia pada halaman antara lain:

a. Preview Gambar

Area kosong di bagian tengah digunakan untuk menampilkan gambar yang dipilih atau diambil secara langsung oleh pengguna. Gambar ini akan diproses oleh model YOLO yang telah ditanamkan dalam aplikasi.

b. Hasil Deteksi

Setelah gambar diproses, hasil yang terdeteksi akan ditampilkan secara langsung di bawah gambar. Teks “Penyakit Terdeteksi” akan berubah menjadi hasil kelas yang di prediksi.

c. Tombol “Buka Kamera”

Ketika diklik pada bagian buka kamera maka secara langsung akan mengambil gambar baru secara langsung.

d. Tombol “Upload dari Galeri”

Memberikan opsi kepada pengguna untuk memilih gambar dari galeri penyimpanan perangkat.

e. Tombol “Lihat Informasi Penyakit”

Jika sudah ada gambar yang terdeteksi maka pengguna akan di arahkan untuk melihat informasi penyakit yang lebih lanjut, tapi jika belum ada gambar yang terdeteksi maka tombol lihat informasi penyakit itu tidak berfungsi.

4. Tampilan Halaman Informasi Penyakit

Tersedia juga fitur tambahan berupa halaman informasi penyakit, yang dapat di klik pada tombol lihat informasi penyakit. Halaman ini memberikan informasi lebih mendalam tentang penyakit yang terdeteksi seperti, nama penyakit, kepercayaan, pengertian penyakit, cara penanganan penyakit baik dalam proses penanaman Hidroponik ataupun Lahan Tanah.



Gambar 4 20 Tampilan Halaman Informasi Penyakit

4.1.7 Pengujian Sistem

Dalam tahap ini, dilakukan dua jenis pengujian, yaitu pengujian sistem menggunakan metode *black-box testing* untuk memastikan fungsionalitas antarmuka berjalan dengan baik, serta pengujian performa model dalam aplikasi menggunakan data uji untuk memastikan akurasi model tetap terjaga setelah integrasi.

a. Pengujian Fungsional (*Black-Box Testing*)

Metode *black-box testing* dilakukan untuk menguji fungsi-fungsi utama aplikasi tanpa memperlihatkan struktur internal kode. Fokusnya adalah pada input yang diberikan pengguna dan output yang dihasilkan aplikasi. Berikut adalah tabel pengujiannya:

Tabel 4.6 Black Box Testing Aplikasi Deteksi Penyakit Daun Melon

No	Fitur yang Diuji	Input	Hasil yang Diharapkan	Status
1	Ambil gambar dari kamera	Gambar hasil kamera	Gambar berhasil dimuat dan diklasifikasikan	Ya
2	Upload gambar dari galeri	Gambar citra daun dari galeri	Gambar berhasil dimuat dan diklasifikasikan	Ya
3	Hasil deteksi muncul	Gambar valid	Label penyakit sesuai gambar muncul di layer	Ya
4	tombol informasi penyakit	Tekan informasi penyakit	Halaman info aplikasi tampil dengan benar	Ya

b. Pengujian Kepercayaan Model Menggunakan Data Uji

Pada tahap ini, dilakukan pengujian sistem menggunakan data uji berjumlah 80 citra daun, yang terdiri dari 20 citra daun per kelas. Tujuan dari pengujian ini yaitu untuk melihat kemampuan sistem dalam mengklasifikasi citra yang tidak pernah dilihat sebelumnya oleh model (*unseen data*), serta mengevaluasi performa aplikasi saat digunakan secara nyata di perangkat Android. Sebagai representasi hasil pengujian, ditampilkan satu contoh prediksi setiap kelas dari aplikasi Android, sehingga setiap kondisi dapat diwakili.

1

Tabel 4.7 Hasil Prediksi Model pada Data Uji

No	Jenis Penyakit	Gambar Citra Daun	Hasil Prediksi	Keterangan Tambahan
1	Bercak Bersudut		Bercak Bersudut	Prediksi sesuai
2	Embun Bulu		Embun Bulu	Prediksi sesuai
3	Layu Fusarium		Layu Fusarium	Prediksi sesuai

Lanjutan Tabel 4.7 Hasil Prediksi Model pada Data Uji

No	Jenis Penyakit	Gambar Citra Daun	Hasil Prediksi	Keterangan Tambahan
4	Mosaik Virus		Mosaik Virus	Prediksi sesuai

Berdasarkan tabel di atas dapat dilihat bahwa sistem deteksi citra penyakit daun melon berhasil mengidentifikasi kategori penyakit dengan tepat. Setiap citra yang diambil mewakili setiap jenis penyakit daun melon, yaitu bercak bersudut, embun bulu, layu fusarium, mosaik virus. Hasil prediksi menunjukkan bahwa model dapat bekerja secara konsisten dalam mendeteksi penyakit daun melon. Keakuratan sistem dalam pengujian ini bahwa penambahan CBAM (*Convolutional Block Attention Module*) yang diterapkan mampu membantu model mengenali fitur penting dari citra.

Selain itu, prediksi yang tepat pada berbagai kondisi pengujian juga mencerminkan bahwa integritas model YOLO+CBAM dalam aplikasi Android berjalan optimal dan dapat dijalankan dalam dunia nyata. Hal ini membuktikan bahwa aplikasi yang dikembangkan tidak hanya akurat di lingkungan pelatihan, namun juga mampu mempertahankan performa ketika digunakan secara langsung untuk mengklasifikasikan citra baru dari pengguna.

4.2 Pembahasan

4.2.1 Pembahasan Hasil Pelatihan YOLOv8+CBAM

Berdasarkan hasil pelatihan yang telah dilakukan, model YOLOv8 yang dikombinasikan dengan *Convolutional Block Attention Module (CBAM)* menunjukkan performa yang baik dalam mendeteksi penyakit daun melon. Hal ini ditunjukkan oleh penurunan nilai training loss dan validation loss selama proses pelatihan, serta peningkatan nilai *precision*, *recall*, *F1-score*, dan *mean Average Precision (mAP)* hingga mencapai kondisi yang stabil pada *epoch* akhir. Hasil tersebut menunjukkan bahwa model mampu mempelajari karakteristik dari setiap kelas penyakit daun melon sehingga proses deteksi dapat dilakukan dengan baik.

Kemampuan model dalam mendeteksi objek dipengaruhi oleh arsitektur YOLOv8 yang dirancang sebagai metode *one-stage object detection*, sehingga proses lokalisasi objek dan klasifikasi dapat dilakukan dalam satu tahapan. Selain itu, penggunaan CBAM membantu model memberikan perhatian pada fitur-fitur yang lebih relevan pada citra daun, sehingga proses ekstraksi fitur menjadi lebih terarah. Hasil yang diperoleh pada penelitian ini menunjukkan bahwa kombinasi YOLOv8 dan CBAM mampu menghasilkan performa deteksi yang baik pada dataset penyakit daun melon.

4.2.2 Pembahasan Hasil Evaluasi Model

Evaluasi model dilakukan menggunakan metrik *precision*, *recall*, *F1-score*, dan *mean Average Precision (mAP)*. Berdasarkan hasil evaluasi, model mampu mendeteksi empat jenis penyakit daun melon dengan tingkat ketepatan yang baik. Nilai *precision* menunjukkan bahwa sebagian besar objek yang terdeteksi merupakan prediksi yang benar, sedangkan nilai *recall* menunjukkan kemampuan

model dalam menemukan objek penyakit yang terdapat pada citra. Nilai *F1-score* menggambarkan keseimbangan antara *precision* dan *recall*, sedangkan nilai *mAP* digunakan untuk mengukur performa model secara keseluruhan dalam mendeteksi lokasi dan kelas objek.

Hasil evaluasi juga didukung oleh kurva *Precision-Confidence*, *Recall-Confidence*, *F1-Confidence*, dan *Precision-Recall* yang menunjukkan bahwa model memiliki performa yang relatif stabil pada berbagai nilai *confidence threshold*. Berdasarkan hasil tersebut, model dinilai mampu memberikan hasil deteksi yang konsisten sehingga dapat digunakan untuk mendukung proses identifikasi penyakit daun melon.

4.2.3 Pembahasan Confusion Matrik

Hasil *confusion matrix* menunjukkan bahwa sebagian besar citra berhasil diprediksi sesuai dengan kelas sebenarnya. Hal ini ditunjukkan oleh dominasi nilai pada diagonal utama matriks yang menunjukkan jumlah prediksi benar lebih banyak dibandingkan prediksi yang salah. Meskipun demikian, masih terdapat beberapa citra yang mengalami kesalahan prediksi maupun tidak terdeteksi sebagai objek penyakit.

Kesalahan tersebut kemungkinan dipengaruhi oleh kemiripan gejala visual antarpenyakit, variasi pencahayaan saat pengambilan gambar, serta ukuran area penyakit yang relatif kecil. Faktor-faktor tersebut dapat menyebabkan model mengalami kesulitan dalam membedakan karakteristik beberapa kelas penyakit. Namun, secara keseluruhan hasil *confusion matrix* menunjukkan bahwa model

memiliki kemampuan yang baik dalam membedakan empat jenis penyakit daun melon.

4.2.4 Pembahasan Implementasi Model Pada Android

Model terbaik (*best.pt*) yang diperoleh dari proses pelatihan kemudian dikonversi ke format *TensorFlow Lite* (*TensorFlow Lite Model*) agar dapat diimplementasikan pada aplikasi Android. Hasil implementasi menunjukkan bahwa model dapat dijalankan dengan baik pada perangkat Android untuk melakukan deteksi penyakit daun melon menggunakan gambar yang diambil melalui kamera maupun gambar yang dipilih dari galeri.

Pengujian menggunakan citra baru yang tidak termasuk dalam dataset pelatihan menunjukkan bahwa aplikasi tetap mampu mendeteksi jenis penyakit daun melon dengan baik. Hal tersebut menunjukkan bahwa model memiliki kemampuan generalisasi yang baik sehingga tidak hanya bekerja pada data pelatihan, tetapi juga mampu mengenali pola penyakit pada citra baru.

4.2.5 Pembahasan Perbandingan dengan You Only Look Once (YOLOv7)

Berdasarkan Tabel 4.4, dilakukan perbandingan antara model YOLOv7 dan YOLOv8 yang dikombinasikan dengan *Convolutional Block Attention Modul* (CBAM) menggunakan dataset penyakit daun melon yang sama. Perbandingan ini bertujuan untuk mengetahui perbedaan performa kedua model dalam mengenali penyakit daun melon berdasarkan matrik evaluasi yang digunakan serta kemampuan masing-masing model.

Hasil pengujian menunjukkan bahwa model YOLOv7 memperoleh nilai *precision* sebesar 57,30%, *recall* sebesar 49,20%, *F1-score* sebesar 52,93%, *mAP@0.5*

sebesar 52,20%, dan $mAP@0.5:0.95$ sebesar 21,50%. Sementara itu, model YOLOv8-CBAM memperoleh precision sebesar 82,92%, recall sebesar 77,32%, F1-score sebesar 79,99%, $mAP@0.5$ sebesar 82,00%. Selain mampu mengenali jenis penyakit, model ini juga dapat menampilkan bounding box pada area daun yang terdeteksi mengalami penyakit, sehingga pengguna dapat mengetahui lokasi penyakit secara langsung.

Dari hasil evaluasi antara dua model tersebut untuk mendeteksi penyakit daun melon lebih tepat menggunakan YOLOv8+CBAM, dikarenakan model YOLOv8+CBAM memiliki nilai evaluasi lebih tinggi dan lebih akurat dari pada YOLOv7 tanpa CBAM.

BAB V

PENUTUP

5.1 Kesimpulan

Berdasarkan hasil penelitian mengenai Deteksi Penyakit Daun Melon Menggunakan YOLOv8 dikombinasikan dengan *Convolutional Block Attention Module (CBAM)* Berbasis Android, maka dapat disimpulkan sebagai berikut:

1. Model YOLOv8 yang dikombinasikan dengan *Convolutional Block Attention Module (CBAM)* berhasil diterapkan untuk mendeteksi empat jenis penyakit daun melon, yaitu bercak bersudut, embun bulu, layu fusarium, dan mosaik virus. Berdasarkan hasil evaluasi, model memperoleh precision sebesar 82,92%, recall sebesar 77,32%, mAP@0.5 sebesar 82,00%, dan Matrik 83%. Hasil tersebut menunjukkan bahwa model memiliki kemampuan yang baik dalam mendeteksi penyakit daun melon.
2. Model YOLOv8-CBAM berhasil diimplementasikan ke dalam aplikasi Android melalui proses konversi ke format *TensorFlow Lite (.tflite)*. Aplikasi yang dikembangkan mampu melakukan deteksi penyakit daun melon menggunakan gambar dari kamera maupun galeri serta menampilkan hasil deteksi berupa nama penyakit, nilai *confidence*, dan *bounding box* pada area daun yang terdeteksi. Hal ini menunjukkan bahwa model dapat digunakan secara praktis sebagai media bantu identifikasi penyakit daun melon.
3. Hasil perbandingan menunjukkan bahwa model YOLOv7 memperoleh precision sebesar 57,30%, recall sebesar 49,20%, *F1-score* sebesar 52,93%, mAP@0.5 sebesar 52,20%, mAP@0.5:0.95 sebesar 21,50%. Hasil tersebut menunjukkan bahwa model YOLOv8+CBAM memiliki performa deteksi yang lebih baik dibanding YOLOv7 pada seluruh matrik evaluasi. Peningkatan

tersebut menunjukkan bahwa penambahan modul *Convolution Block Attention Module* (CBAM) mampu meningkatkan kemampuan model dalam mengekstraksi fitur penting sehingga menghasilkan deteksi penyakit daun melon yang lebih akurat. Dengan demikian, model YOLOv8+CBAM lebih sesuai untuk diimplementasikan pada aplikasi android dalam penelitian ini.

5.2 Saran

Berdasarkan hasil penelitian yang telah dilakukan, beberapa saran yang dapat diberikan untuk penelitian selanjutnya adalah sebagai berikut:

1. Menambahkan jumlah serta variasi dataset penyakit daun melon dengan kondisi pencahayaan, sudut pengambilan gambar, dan tingkat keparahan penyakit yang lebih beragam agar model memiliki kemampuan generalisasi yang lebih baik.
2. Mengembangkan penelitian dengan menggunakan arsitektur YOLO versi terbaru atau mengombinasikan model dengan metode attention lainnya sebagai pembanding untuk meningkatkan performa deteksi penyakit daun melon.
3. Mengembangkan aplikasi Android dengan menambahkan fitur informasi mengenai gejala, penyebab, dan penanganan setiap penyakit daun melon sehingga aplikasi tidak hanya berfungsi sebagai alat deteksi, tetapi juga sebagai media edukasi bagi masyarakat dan petani.
4. Mengoptimalkan performa aplikasi agar proses deteksi menjadi lebih cepat dan stabil pada berbagai jenis perangkat Android.

DAFTAR PUSTAKA

- [1] C. A. Wangsa and U. M. Nusantara, "Deteksi Penyakit Alternaria ..., Christophorus Augusta Wangsa, Universitas Multimedia Nusantara," pp. 1–5, 2023.
- [2] H. A. Susanto, A. Himawan, and E. N. Kristalisasi, "Kajian Penyakit Layu *Fusarium oxysporum* pada Tanaman Melon (*Cucumis melo* L .) Hidroponik di Greenhouse," vol. 7, no. 1, pp. 87–97, 2023.
- [3] N. Izhar, "Bertanam Melon," vol. 38, no. 10, pp. 1801–1804, 1997.
- [4] L. Sinaga, N. Zahara, P. Tanaman, F. Pertanian, and U. Bengkulu, "Kajian Patogen Penyebab Penyakit Pada Tanaman Melon (*Cucumis melo* L.) di Bengkulu 1,2," vol. 18, no. 1, pp. 22–25, 2022.
- [5] S. Woo, J. Park, J. Y. Lee, and I. S. Kweon, "CBAM: Convolutional block attention module," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 11211 LNCS, pp. 3–19, 2018, doi: 10.1007/978-3-030-01234-2_1.
- [6] J. L. de Moraes, J. de Oliveira Neto, C. Badue, T. Oliveira-Santos, and A. F. de Souza, "Yolo-Papaya: A Papaya Fruit Disease Detector and Classifier Using CNNs and Convolutional Block Attention Modules," *Electron.*, vol. 12, no. 10, 2023, doi: 10.3390/electronics12102202.
- [7] B. Durroh, M. Y. Dawud, D. Fakultas, and P. Universitas, "Jurnal Sosiologi Pertanian dan Agribisnis STRATEGI PENGENDALIAN HAMA DAN PENYAKIT PADA BUDIDAYA TANAMAN MELON (*Cucumis Melo* . L)," vol. 4, no. 2, 2022.
- [8] Y. M. Nindi Apriliani , Muhammad Zainal Fanani, "BUDIDAYA DAN ANALISIS USAHA MELON (*Cucumis melo* L .)," vol. 4, pp. 1106–1128, 2025.
- [9] L. Rahma, H. Syaputra, A. H. Mirza, and S. D. Purnamasari, "Objek Deteksi Makanan Khas Palembang Menggunakan Algoritma YOLO (You Only Look Once)," vol. 2, no. 3, 2021.
- [10] D. H. Saputra *et al.*, "OBJECT DETECTION UNTUK MENDETEKSI CITRA BUAH BUAHAN," vol. 2, no. 2, pp. 70–80, 2023.
- [11] C. M. Badgujar, A. Poulouse, and H. Gan, "Agricultural Object Detection with You Look Only Once (YOLO) Algorithm : A Bibliometric and Systematic Literature Review," pp. 1–30, 2023.
- [12] T. H. Ruihao Dong, Aya Shiraiwa, Achara Pawasut, Kesaraporn Sreechun, "Diagnosis of Citrus Greening Using Artificial Intelligence : A Faster Region-Based Convolutional Neural Network Approach," 2024.
- [13] M. Sholikhin, S. Kom, R. A. H, S. Kom, and M. Kom, "KLASIFIKASI PENYAKIT PADA CITRA DAUN MELON MENGGUNAKAN ALGORITMA CONVOLUTION NEURAL NETWORK," vol. 7, no. 1, pp.

525–530, 2022.

- [14] M. R. Ridha, R. Asriati, and D. Y. Prasetyo, “Accuracy Optimization of Convolutional Neural Network Model in Chili Plant Disease Classification,” vol. X, no. X, 2025, doi: 10.15408/aism.vxix.xxxxx.
- [15] I. Topan *et al.*, “EVALUASI AUGMENTASI DATA PADA DETEKSI PENYAKIT DAUN TEBU DENGAN YOLOV8,” vol. 8, no. 4, pp. 7547–7552, 2024.
- [16] S. Real-time, N. Safina, M. Fikry, and C. Agusniar, “Penerapan YOLOv8 Dalam Deteksi Penyakit Daun Jambu Air,” vol. 17, no. 2, pp. 154–164, 2025.
- [17] B. K. Ekhsanto, B. A. Kusuma, A. P. Kuncoro, C. S. Faculty, and U. A. Purwokerto, “IMPLEMENTATION OF YOU ONLY LOOK ONCE V8 ALGORITHM IN POTATO IMPLEMENTASI ALGORITMA YOU ONLY LOOK ONCE V8 PADA SISTEM,” vol. 5, no. 4, pp. 125–132, 2024.
- [18] A. Dwi, S. Putri, A. P. Tampubolon, R. F. Prayata, and S. K. Nisa, “DETEKSI DAN KLASIFIKASI HAMA POTATO BEETLE PADA TANAMAN KENTANG MENGGUNAKAN YOLOV8,” vol. 11, no. 4, 2024, doi: 10.25126/jtiik.1148092.
- [19] R. R. Putra and D. Sasongko, “Implementasi Algoritma YOLO V8 (You Only Look Once) Dalam Deteksi Penyakit Daun Durian,” vol. 6, no. 3, pp. 0–9, 2024, doi: 10.47065/bits.v6i3.6136.
- [20] S. D. SITEPU, “Penerapan CBAM Pada Xception Untuk Klasifikasi Penyakit Daun Tebu,” 2025.
- [21] M. Lv and W. Su, “YOLOV5-CBAM-C3TR : an optimized model based on transformer module and attention mechanism for apple leaf disease detection,” no. January, pp. 1–13, 2024, doi: 10.3389/fpls.2023.1323301.
- [22] D. Buah, D. A. N. Florikultura, D. Jen, D. L. Hortikultura, and K. E. M. E. Pertanian, “Direktorat buah dan florikultura direktorat jen dera l hortikultura kem enterian pertanian 2016,” 2016.
- [23] S. Nelly Saptayanti, S. Ami Cahyani Ratnaningrum, and S. Evy Octavia, *Buku Pedoman Organisme Pengganggu Tumbuhan Pengelolaan Secara Ramah Lingkungan Pada Tanaman Melon*. 2015.
- [24] TEGAR RAMADHANI, “APLIKASI DETEKSI PENYAKIT PADA DAUN TANAMAN SINGKONG MENGGUNAKAN ALGORITMA YOU ONLY LOOK ONCE (YOLO) V8 BERBASIS ANDROID,” 2024.

LAMPIRAN

