

Turnitin LLCC

2213020019_VIRGINIA ABUK KLAU

 TI-12

Document Details

Submission ID

trn:oid::2945:396338062

Submission Date

Jun 30, 2026, 10:46 AM GMT+7

Download Date

Jun 30, 2026, 10:52 AM GMT+7

File Name

2213020019_VIRGINIA ABUK KLAU.pdf

File Size

1.9 MB

93 Pages

18,140 Words

114,179 Characters

18% Overall Similarity

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

Filtered from the Report

- Bibliography

Top Sources

- 8%  Internet sources
- 7%  Publications
- 17%  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

- 8% Internet sources
- 7% Publications
- 17% 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	journal.ipm2kpe.or.id	<1%
2	Publication	Ananda Rizki Dani, Irma Handayani. "Classification of Yogyakarta Batik Motifs Usi...	<1%
3	Student papers	UPN Veteran Yogyakarta on 2026-05-25	<1%
4	Student papers	Universitas Islam Riau on 2026-05-13	<1%
5	Student papers	Universitas Sumatera Utara on 2026-06-04	<1%
6	Internet	repository.ub.ac.id	<1%
7	Student papers	UPN Veteran Yogyakarta on 2026-05-04	<1%
8	Internet	ojs.stmik-banjarbaru.ac.id	<1%
9	Student papers	Universitas Dian Nuswantoro on 2026-01-09	<1%
10	Internet	ejurnal.methodist.ac.id	<1%
11	Student papers	UPN Veteran Yogyakarta on 2025-10-28	<1%

12	Student papers	UPN Veteran Yogyakarta on 2026-05-05	<1%
13	Student papers	UPN Veteran Yogyakarta on 2026-01-06	<1%
14	Student papers	Universitas Putera Batam on 2026-02-03	<1%
15	Student papers	Universitas Islam Riau on 2026-06-19	<1%
16	Student papers	Universitas Sumatera Utara on 2025-05-23	<1%
17	Student papers	Universitas Islam Riau on 2025-09-02	<1%
18	Student papers	Universitas Islam Riau on 2026-06-19	<1%
19	Student papers	UPN Veteran Yogyakarta on 2025-10-01	<1%
20	Student papers	Perpustakaan on 2025-08-07	<1%
21	Internet	dspace.uii.ac.id	<1%
22	Student papers	Universitas Pancasila on 2026-06-26	<1%
23	Student papers	Universitas Pendidikan Muhammadiyah Sorong on 2026-06-29	<1%
24	Student papers	Universitas Pancasila on 2025-08-07	<1%
25	Student papers	Universitas Islam Riau on 2026-05-18	<1%

26	Student papers	Universitas Pamulang on 2026-06-22	<1%
27	Internet	docplayer.info	<1%
28	Student papers	UPN Veteran Yogyakarta on 2026-04-22	<1%
29	Student papers	Exeed College on 2025-12-30	<1%
30	Student papers	Universitas Khairun on 2026-05-11	<1%
31	Student papers	STT PLN on 2025-08-11	<1%
32	Student papers	UPN Veteran Yogyakarta on 2026-04-30	<1%
33	Student papers	Universitas Pancasila on 2026-03-04	<1%
34	Internet	digilib.iain-palangkaraya.ac.id	<1%
35	Internet	ejournal.uigm.ac.id	<1%
36	Publication	Iqbal May Aryanto, Syaiful Mansur, Ayu Sintianingrum, Ayang Kinasih, Eko Hari...	<1%
37	Student papers	Universitas Tarumanagara on 2025-12-18	<1%
38	Student papers	Telkom University on 2026-01-09	<1%
39	Internet	repositori.unimma.ac.id	<1%

40	Internet	repository.unpkediri.ac.id	<1%
41	Internet	repository.upstegal.ac.id	<1%
42	Student papers	UPN Veteran Yogyakarta on 2026-06-04	<1%
43	Internet	ejournal.uniramalang.ac.id	<1%
44	Publication	I Dewa Gede Agung Wibhisana Natih, Made Windu Antara Kesiman, I Made Gede ...	<1%
45	Student papers	School of Business and Management ITB on 2025-07-14	<1%
46	Student papers	Sultan Agung Islamic University on 2025-06-03	<1%
47	Student papers	Universitas Muhammadiyah Purwokerto on 2025-08-15	<1%
48	Student papers	STT PLN on 2025-08-11	<1%
49	Student papers	UPN Veteran Yogyakarta on 2026-05-29	<1%
50	Internet	text-id.123dok.com	<1%
51	Student papers	Indonesia Police Academy - University of Police Science on 2025-11-22	<1%
52	Publication	Lintan Zhuliani, Refliani Marsela. "Rancang Bangun Sistem Informasi Manajemen..."	<1%
53	Student papers	Universitas Islam Indonesia on 2026-06-15	<1%

54	Student papers	Universitas Sumatera Utara on 2026-06-17	<1%
55	Internet	ejournal.omahtabing.com	<1%
56	Internet	indojurnal.com	<1%
57	Publication	I Gede Dhananjaya, Kadek Yota Ernanda Aryanto, I Gede Partha Sindu. "Pemanfa...	<1%
58	Student papers	Perpustakaan on 2025-08-06	<1%
59	Student papers	Police Academy – University of Police Science on 2025-11-26	<1%
60	Student papers	Universitas Pertahanan RI on 2026-06-18	<1%
61	Student papers	Universitas Tarumanagara on 2025-12-17	<1%
62	Student papers	Universitas Terbuka on 2025-12-04	<1%
63	Internet	digilib.uinsby.ac.id	<1%
64	Internet	digilib.widyagama.ac.id	<1%
65	Student papers	Tarumanagara University on 2026-06-21	<1%
66	Student papers	Telkom University on 2026-06-18	<1%
67	Student papers	Telkom University on 2026-06-30	<1%

68	Student papers	Universitas Islam Indonesia on 2026-04-20	<1%
69	Student papers	Universitas Islam Riau on 2025-12-15	<1%
70	Internet	digilib.uin-suka.ac.id	<1%
71	Student papers	Udayana University on 2024-12-02	<1%
72	Student papers	Universitas Muslim Indonesia on 2024-05-07	<1%
73	Student papers	Academic Library Consortium on 2025-01-07	<1%
74	Publication	Tri Ahmad Jabar JABAR, Rudi Heriansyah, Evi Purnamasari. "Pengenalan Pola Hur...	<1%
75	Student papers	UPN Veteran Yogyakarta on 2026-06-02	<1%
76	Student papers	Universitas Khairun on 2026-06-08	<1%
77	Student papers	Universitas Pamulang on 2026-05-04	<1%
78	Student papers	Universitas Tarumanagara on 2025-12-23	<1%
79	Publication	Nurfalah Nurfalah, Iiz Izmuiddin. "Dampak Dinamika Persaingan Harga Beli Sayur...	<1%
80	Student papers	STT PLN on 2023-07-26	<1%
81	Student papers	Universitas Indraprasta PGRI on 2026-03-01	<1%

82	Student papers	Universitas Negeri Semarang - iTh on 2024-07-14	<1%
83	Internet	digilib.uns.ac.id	<1%
84	Internet	journal.isas.or.id	<1%
85	Internet	repo.itera.ac.id	<1%
86	Internet	santika.upnjatim.ac.id	<1%
87	Student papers	Binus University International on 2026-06-08	<1%
88	Student papers	Institut Agama Islam Al-Zaytun Indonesia on 2026-06-25	<1%
89	Publication	Intan Sulistyawati, Devi Permatasari. "Efektivitas Sistem Pengendalian Internal d...	<1%
90	Student papers	Konsorsium PTS Batch 5 on 2026-06-24	<1%
91	Student papers	STT PLN on 2023-07-20	<1%
92	Student papers	Universitas Brawijaya on 2017-01-24	<1%
93	Student papers	Universitas Putera Batam on 2026-02-03	<1%
94	Internet	core.ac.uk	<1%
95	Internet	informatika.unpkediri.ac.id	<1%

96	Publication	Anwar Solihin, Jannus Maurits Nainggolan, Dikpride Despa. "Karakteristik Peluah...	<1%
97	Publication	Mahardika Wildan Fitrian, Wiwi Widayani, Heri Sismoro, Nur Aini, Dwi Nurani. "Pr...	<1%
98	Student papers	Universitas Islam Riau on 2025-10-17	<1%
99	Student papers	Universitas PGRI Semarang on 2023-07-24	<1%
100	Student papers	Universitas Pertahanan RI on 2026-06-19	<1%
101	Student papers	Universitas Putera Batam on 2021-03-15	<1%
102	Internet	ejurnal.stmik-budidarma.ac.id	<1%
103	Internet	eprint.ivet.ac.id	<1%
104	Internet	jmiki.aptirmik.or.id	<1%
105	Internet	proceeding.unpkediri.ac.id	<1%
106	Internet	repository.unair.ac.id	<1%
107	Internet	www.biorxiv.org	<1%
108	Internet	www.coursehero.com	<1%
109	Publication	Hanna Willa Dhany, Aminuddin Indra Permana, Fahmi Izhari, Andhika Pratama Gi...	<1%

110	Student papers	Perpustakaan on 2025-09-02	<1%
111	Student papers	Rijksuniversiteit Groningen - Tii on 2026-04-16	<1%
112	Student papers	UIN Raden Intan Lampung on 2026-05-26	<1%
113	Student papers	UPN Veteran Yogyakarta on 2026-05-29	<1%
114	Student papers	Universitas Bengkulu on 2026-04-14	<1%
115	Student papers	Universitas Malikussaleh on 2026-04-28	<1%
116	Student papers	Universitas Tarumanagara on 2025-12-09	<1%
117	Internet	docobook.com	<1%
118	Internet	eprints.uny.ac.id	<1%
119	Internet	etheses.uin-malang.ac.id	<1%
120	Internet	nairobiultrabooks.odoo.com	<1%
121	Internet	www.asifcomputers.com	<1%
122	Internet	www.courtsmammouth.mu	<1%
123	Student papers	Institut Teknologi Sumatera on 2026-01-12	<1%

124	Student papers	Konsorsium PTS Batch 5 on 2026-06-24	<1%
125	Publication	Nurhayati Nurhayati, Aditya Kusuma. "PERANCANGAN SISTEM DETEKSI KEMATAN...	<1%
126	Student papers	Sriwijaya University on 2025-11-24	<1%
127	Student papers	UPN Veteran Yogyakarta on 2026-05-26	<1%
128	Student papers	UPN Veteran Yogyakarta on 2026-06-04	<1%
129	Student papers	Universitas Brawijaya on 2017-01-14	<1%
130	Student papers	Universitas Brawijaya on 2018-01-09	<1%
131	Internet	eprints.umpo.ac.id	<1%
132	Internet	id.scribd.com	<1%
133	Internet	journals.upi-yai.ac.id	<1%
134	Internet	lib.unnes.ac.id	<1%
135	Internet	repository.ut.ac.id	<1%
136	Internet	www.researchgate.net	<1%
137	Internet	www.slideshare.net	<1%

138	Publication	Aprilian Gevindo, Syafri Arlis. "Smoker Detection Using the You Only Look Once (Y...	<1%
139	Publication	Irfan Ibrahim, Maulana Fitra Ramadhani, Muhammad Ridho, M. Wisnu Adjie Pra...	<1%
140	Publication	Kharisma Monika Dian Pertiwi. "Implementasi dan Analisa Performa Berbagai Pla...	<1%
141	Student papers	Nottingham Trent University on 2023-12-10	<1%
142	Publication	Nurussakinah Nurussakinah, Muhammad Faisal. "Klasifikasi Penyakit Diabetes M...	<1%
143	Student papers	Perpustakaan on 2025-08-08	<1%
144	Student papers	Politeknik Negeri Bandung on 2017-07-21	<1%
145	Publication	Ridho Sholehurrohman, Igit Sabda Ilman, Mita Anggraaeni, Rahman Taufik, Muh...	<1%
146	Student papers	SDM Universitas Gadjah Mada on 2021-10-08	<1%
147	Student papers	Sultan Agung Islamic University on 2026-01-24	<1%
148	Student papers	Telkom University on 2026-05-06	<1%
149	Student papers	UPN Veteran Yogyakarta on 2026-01-06	<1%
150	Student papers	Universitas Brawijaya on 2016-06-13	<1%
151	Student papers	Universitas Brawijaya on 2016-12-22	<1%

152	Student papers	Universitas Brawijaya on 2018-11-26	<1%
153	Student papers	Universitas Brawijaya on 2020-02-12	<1%
154	Student papers	Universitas Brawijaya on 2023-12-28	<1%
155	Student papers	Universitas Islam Negeri Sumatera Utara on 2025-09-16	<1%
156	Student papers	Universitas Islam Riau on 2025-12-04	<1%
157	Student papers	Universitas Islam Riau on 2026-06-19	<1%
158	Student papers	Universitas Jember on 2025-06-16	<1%
159	Student papers	Universitas Muhammadiyah Kotabumi on 2025-08-13	<1%
160	Student papers	Universitas Muhammadiyah Makassar on 2026-06-25	<1%
161	Student papers	Universitas Negeri Padang on 2026-05-01	<1%
162	Student papers	Universitas Sumatera Utara on 2026-03-09	<1%
163	Student papers	Universitas Sumatera Utara on 2026-06-10	<1%
164	Student papers	Universiti Kebangsaan Malaysia on 2017-06-15	<1%
165	Internet	edisonzero.blogspot.com	<1%

166	Internet	eprints.uns.ac.id	<1%
167	Internet	eprints.unsri.ac.id	<1%
168	Internet	id.123dok.com	<1%
169	Internet	juti.if.its.ac.id	<1%
170	Internet	nanopdf.com	<1%
171	Internet	repository.iainbengkulu.ac.id	<1%
172	Internet	repository.umsu.ac.id	<1%
173	Internet	simki.unpkediri.ac.id	<1%
174	Internet	simons-ratschlag.de	<1%
175	Internet	www.theseus.fi	<1%
176	Publication	Baehaqi Wahyu Kurniawan, Pungkas Subarkah, Dinar Mustofa. "Perbandingan N...	<1%
177	Student papers	Indonesia Police Academy - University of Police Science on 2025-11-22	<1%
178	Student papers	Middlesex University on 2024-01-21	<1%
179	Publication	Salamun Salamun, Firman Wazir. "RANCANG BANGUN SISTEM PENGENALAN WAJ...	<1%

180	Student papers	Sriwijaya University on 2023-07-07	<1%
181	Student papers	Universitas Bengkulu on 2024-09-25	<1%
182	Student papers	Universitas Brawijaya on 2018-01-04	<1%
183	Student papers	Universitas Pancasila on 2026-02-21	<1%
184	Student papers	Universitas Sumatera Utara on 2026-06-04	<1%
185	Student papers	Universitas Tarumanagara on 2025-12-09	<1%
186	Student papers	Universitas Tarumanagara on 2025-12-15	<1%
187	Internet	adoc.pub	<1%
188	Publication	Irwan Argi Himawan, Tedy Rismawan, Suhardi Suhardi. "SISTEM KEAMANAN SEPE...	<1%
189	Student papers	Sriwijaya University on 2026-05-04	<1%
190	Student papers	UIN Raden Intan Lampung on 2025-07-17	<1%
191	Student papers	UNIVERSITAS BUDI LUHUR on 2026-01-30	<1%
192	Student papers	UPN Veteran Yogyakarta on 2026-06-18	<1%
193	Student papers	Universitas Borneo Tarakan on 2025-06-25	<1%

194	Student papers	
Universitas Brawijaya on 2016-11-07		<1%
195	Student papers	
Universitas Khairun on 2026-05-19		<1%
196	Student papers	
Universitas Muhammadiyah Buton on 2026-06-05		<1%
197	Student papers	
iGroup on 2014-02-19		<1%

BAB I

PENDAHULUAN

A. Latar Belakang Masalah

Batik merupakan salah satu warisan budaya bangsa Indonesia yang telah diakui oleh UNESCO sebagai *Intangible Cultural Heritage of Humanity* sejak tahun 2009. Batik memiliki keragaman motif dan warna yang sangat tinggi. Setiap daerah di Indonesia memiliki ciri khas motif dan warna yang mencerminkan nilai *filosofis*, sejarah, serta identitas budaya lokal (Agustin & Latifah, 2025). Keberagaman tersebut menjadikan batik tidak hanya memiliki nilai *estetika*, tetapi juga nilai *edukatif* dan kultural yang penting untuk dilestarikan.

Proses identifikasi, klasifikasi, dan dokumentasi motif batik secara manual memerlukan keahlian khusus dan waktu yang cukup lama serta rentan terhadap *subjektivitas*. Kesulitan identifikasi semakin tinggi pada motif yang memiliki ornamen rumit, kemiripan visual antarkelas yang tinggi, variasi pola dalam kelas, dan komposisi warna yang beragam (Sutresno, 2024; Suyahman & Hapsari, 2025). Berdasarkan kondisi tersebut, hipotesis penelitian ini adalah model YOLOv8 dapat digunakan untuk mendeteksi motif batik secara otomatis. Performa model diukur menggunakan nilai *precision*, *recall*, *mAP50*, dan *mAP50–95* sebagai indikator kemampuan model dalam mengenali dan melokalisasi motif batik (Sutresno, 2024). Selanjutnya, kemampuan sistem dalam menentukan warna dominan diuji menggunakan algoritma *K-Means* untuk mengelompokkan piksel warna pada area motif yang terdeteksi. Pusat kluster dengan jumlah anggota terbanyak digunakan sebagai warna dominan, kemudian direpresentasikan dalam nilai RGB dan dikonversi menjadi kode heksadesimal (Molada-Tebar, Marqués-Mateu, Lerma, & Westland, 2020).

Permasalahan lain yang muncul adalah penentuan warna dominan batik secara *objektif*. Warna yang terekam pada citra batik dapat mengalami variasi akibat teknik pewarnaan, kondisi pencahayaan saat pengambilan gambar,

179 kualitas kamera, dan kompleksitas pola pada kain. Kondisi tersebut menyebabkan penentuan warna dominan melalui pengamatan visual berpotensi menghasilkan penilaian yang berbeda. Padahal, fitur warna merupakan salah satu karakteristik visual yang dapat digunakan untuk membedakan dan mengklasifikasikan motif batik sehingga perlu diidentifikasi secara terukur untuk mendukung proses dokumentasi dan pengarsipan batik secara digital (Winarno, Hadikurniawati, Septiarini, & Hamdani, 2022).

97 Perkembangan Artificial Intelligence (AI), khususnya pada bidang computer vision dan deep learning, memungkinkan proses pengenalan pola dan analisis citra dilakukan secara otomatis. Salah satu pendekatan yang dapat digunakan adalah object detection, yaitu metode yang mampu mengenali jenis objek sekaligus menentukan lokasinya dalam citra. Model You Only Look Once (YOLO) dikenal memiliki kemampuan melakukan deteksi objek dalam satu tahap pemrosesan sehingga dapat menghasilkan label kelas, nilai confidence, dan bounding box pada objek yang terdeteksi (Vijayakumar & Vairavasundaram, 2024).

125 Penelitian mengenai YOLO dan YOLOv8 sebelumnya lebih banyak membahas performa deteksi objek secara umum, sedangkan penerapannya pada motif batik masih menghadapi tantangan berupa pola berulang, detail visual yang halus, kemiripan antarmotif, latar belakang yang kompleks, serta variasi warna yang tinggi ((Vijayakumar & Vairavasundaram, 2024; Mao & Hong, 2025). Penelitian terdahulu mengenai motif batik juga umumnya berfokus pada klasifikasi citra secara keseluruhan sehingga belum mampu menunjukkan lokasi motif secara spesifik melalui bounding box. Analisis warna yang dilakukan terhadap keseluruhan citra berpotensi dipengaruhi oleh warna latar belakang dan bagian nonmotif. Kondisi tersebut menunjukkan adanya kebutuhan terhadap pendekatan yang dapat mendeteksi lokasi motif sekaligus menganalisis warna dominan pada area motif yang telah terdeteksi.

4 Berdasarkan kesenjangan penelitian tersebut, penelitian ini mengembangkan sistem otomatis yang mengintegrasikan YOLOv8 dan K-Means Clustering. YOLOv8 digunakan untuk mendeteksi serta

11 mengidentifikasi jenis dan lokasi motif batik, sedangkan *K-Means Clustering* digunakan untuk mengelompokkan *piksel* dan mengekstraksi warna dominan dari *area* motif hasil deteksi. Hasil deteksi motif ditampilkan dalam bentuk *bounding box*, label kelas, dan nilai *confidence*, sedangkan hasil ekstraksi warna dominan ditampilkan dalam bentuk nilai RGB dan kode heksadesimal.

Kebaruan penelitian ini terletak pada *integrasi* YOLOv8 untuk mendeteksi motif batik dengan *K-Means Clustering* untuk mengekstraksi warna dominan secara khusus berdasarkan area motif yang terdeteksi, bukan berdasarkan keseluruhan citra. Penelitian berjudul “Deteksi Motif dan Warna Dominan Batik Menggunakan YOLOv8” diharapkan menghasilkan sistem yang dapat mendukung proses identifikasi, dokumentasi, dan digitalisasi batik Indonesia. Sistem yang dikembangkan juga diharapkan dapat menjadi alat bantu dalam mengenali motif dan memperoleh informasi warna dominan secara terukur berdasarkan hasil pemrosesan citra.

50 B. Identifikasi Masalah

Berdasarkan latar belakang di atas, maka dapat diidentifikasi beberapa permasalahan sebagai berikut:

1. Belum tersedianya sistem otomatis yang mampu mengenali motif batik secara *spesifik* dan akurat.
2. Sulitnya menentukan warna dominan batik secara *objektif* akibat variasi warna dan kompleksitas pola.
3. Proses klasifikasi motif batik masih dilakukan secara manual dan bergantung pada *subjektivitas* pengamat.
4. Diperlukan metode berbasis deep learning yang mampu melakukan deteksi motif dan analisis warna dominan secara terintegrasi.

68 3 C. Rumusan Masalah

51 Berdasarkan identifikasi masalah di atas, maka rumusan masalah dalam penelitian ini adalah:

1. Bagaimana membangun aplikasi deteksi motif dan warna dominan batik

berbasis Streamlit yang mengintegrasikan YOLOv8 dan *K-Means Clustering*?

2. Bagaimana menerapkan metode YOLOv8 untuk mendeteksi dan mengidentifikasi jenis serta lokasi motif batik pada citra digital?
3. Bagaimana menerapkan *K-Means Clustering* untuk menentukan warna dominan pada area motif batik yang telah terdeteksi?
4. Bagaimana performa model YOLOv8 dalam mendeteksi motif batik berdasarkan nilai *precision, recall, mAP50, dan mAP50-95*?

D. Batasan Masalah

Agar penelitian lebih terarah dan fokus, maka batasan masalah dalam penelitian ini adalah:

1. *Dataset* yang digunakan berjumlah 586 citra batik yang terdiri atas lima kelas, yaitu Batik Bali, Batik Betawi, Batik Garutan, Batik Parang, dan Batik Sekar.
2. *Dataset* dibagi menjadi 513 citra data latih, 49 citra data validasi, dan 24 citra data uji. Pembagian tersebut setara dengan 87,54% data latih, 8,36% data validasi, dan 4,10% data uji.
3. Citra yang digunakan memiliki resolusi asli minimum sebesar 640 piksel dan resolusi maksimum sebesar 640 piksel. Seluruh citra kemudian diubah ukurannya menjadi 640×640 piksel sebelum digunakan dalam proses pelatihan dan pengujian model.
4. Format citra yang digunakan dibatasi pada JPG, JPEG, dan PNG. Citra yang tidak dapat dibaca, rusak, atau tidak menampilkan motif batik secara jelas tidak digunakan dalam dataset.
5. Model deteksi objek yang digunakan adalah YOLOv8n dari pustaka *Ultralytics* versi 8.4.53. Model dilatih selama 50 epoch dengan *batch size* 16, ukuran input 640×640 piksel, *optimizer AdamW*, dan bobot awal pretrained YOLOv8n.
6. Lingkungan pemrograman menggunakan *Python* versi 3.12.0, *OpenCV* versi 4.13.0.92, *NumPy* versi 2.4.4, *scikit-learn* versi 1.8.0, dan *Streamlit*

versi 1.57.0.

7. YOLOv8 digunakan untuk mendeteksi kelas dan lokasi motif batik dalam bentuk *bounding box*, sedangkan *K-Means Clustering* digunakan untuk mengekstraksi warna dominan dari area motif yang telah terdeteksi.
8. Jumlah cluster pada proses *K-Means Clustering* ditetapkan sebanyak tiga cluster. Cluster dengan jumlah piksel terbanyak dipilih sebagai warna dominan dan ditampilkan dalam bentuk nilai RGB serta kode heksadesimal.
9. Evaluasi *performa* deteksi motif dibatasi pada nilai *precision*, *recall*, *mAP50*, dan *mAP50–95*. Penelitian tidak membahas perbandingan seluruh varian YOLO atau perbandingan dengan seluruh algoritma deteksi objek lainnya.
10. Aplikasi dikembangkan menggunakan Streamlit dan dijalankan secara lokal pada sistem operasi Windows 11 Home Single Language versi 22H2 dengan arsitektur 64-bit. Perangkat pengujian menggunakan prosesor 11th Gen Intel Core i5-11400H 2,70 GHz, RAM 8 GB, NVIDIA GeForce RTX 3050 Laptop GPU 4 GB, dan penyimpanan SSD 477 GB. Akses sistem dibatasi pada satu pengguna yang mengoperasikan aplikasi secara lokal tanpa autentikasi dan tanpa penggunaan secara bersamaan. Sistem hanya menerima satu citra pada setiap proses analisis dan tidak mencakup pengolahan video secara waktu nyata.
11. Penelitian difokuskan pada deteksi motif dan ekstraksi warna dominan. Penelitian tidak membahas keaslian batik, teknik produksi, bahan kain, jenis pewarna, nilai *filosofis* motif, maupun penentuan daerah asal batik secara otomatis.
12. Ketepatan warna dominan dibatasi pada hasil pengelompokan piksel dalam ruang warna RGB. Hasil warna dapat dipengaruhi oleh pencahayaan, kualitas kamera, bayangan, dan kondisi pengambilan citra.

E. Tujuan Penelitian

Berdasarkan rumusan masalah yang telah ditetapkan, tujuan penelitian ini

adalah sebagai berikut:

1. Merancang, mengimplementasikan, dan menguji fungsionalitas aplikasi berbasis Streamlit yang mengintegrasikan YOLOv8 untuk deteksi motif batik dan *K-Means Clustering* untuk ekstraksi warna dominan.
2. Menguji kemampuan YOLOv8 dalam mendeteksi jenis dan lokasi motif batik pada citra digital melalui keluaran berupa label kelas, bounding box, dan nilai confidence.
3. Mengukur hasil ekstraksi warna dominan menggunakan *K-Means Clustering* berdasarkan jumlah piksel pada setiap cluster di dalam area motif hasil deteksi serta menyajikannya dalam bentuk nilai RGB dan kode heksadesimal.
4. Mengukur performa model YOLOv8 dalam mendeteksi motif batik berdasarkan nilai *precision*, *recall*, *mAP50*, dan *mAP50–95*.

F. Manfaat Penelitian

Adapun manfaat yang diharapkan dari penelitian ini adalah sebagai berikut:

1. Manfaat Teoretis

Penelitian ini diharapkan dapat memperkaya kajian dalam bidang Teknik Informatika, khususnya *computer vision*, *deteksi objek*, dan pengolahan citra digital. Hasil penelitian dapat menjadi *referensi* mengenai penerapan YOLOv8 untuk mendeteksi motif batik serta integrasinya dengan *K-Means Clustering* untuk mengekstraksi warna dominan dari area *objek* yang terdeteksi.

2. Manfaat Praktis

Penelitian ini diharapkan dapat menghasilkan aplikasi yang membantu pengrajin batik dan pihak terkait dalam mengenali motif serta memperoleh informasi warna dominan secara otomatis. Sistem yang dikembangkan dapat digunakan sebagai alat bantu awal dalam kegiatan dokumentasi, pengelompokan, dan penyajian informasi citra batik secara digital.

BAB II

LANDASAN TEORI

A. Teori dan Penelitian Terdahulu

1. Landasan Teori

Sistem yang dikembangkan dalam penelitian ini mengintegrasikan YOLOv8 dan *K-Means Clustering* untuk melakukan deteksi motif serta ekstraksi warna dominan pada citra batik. YOLOv8 digunakan untuk mengekstraksi *fitur visual*, menentukan kelas motif, dan menghasilkan *koordinat bounding box*. Area yang berada di dalam *bounding box* selanjutnya digunakan sebagai *Region of Interest (ROI)* dan diproses menggunakan *K-Means Clustering* untuk memperoleh warna dominan.

Landasan teori penelitian difokuskan pada konsep teknologi dan mekanisme algoritma yang mendukung sistem, meliputi pengolahan citra digital, deteksi objek, mekanisme YOLO, *arsitektur YOLOv8*, *K-Means Clustering*, analisis warna dominan, serta *evaluasi model deteksi objek*.

a. Pengolahan Citra Digital

Pengolahan citra digital merupakan proses komputasi terhadap citra untuk menyesuaikan bentuk data, memperbaiki kualitas citra, *mengekstraksi fitur*, dan menghasilkan informasi yang dapat dianalisis oleh sistem. Citra digital direpresentasikan sebagai kumpulan *piksel* yang tersusun dalam baris dan kolom. Setiap piksel memiliki nilai intensitas atau komponen warna yang dapat diproses menggunakan algoritma komputer (Whardana & Aziz, 2025).

Tahapan pengolahan citra dalam penelitian ini meliputi:

1. *Akuisisi citra*, yaitu proses memperoleh citra melalui kamera atau sumber data digital.
2. *Pra-pemrosesan*, yaitu proses mengubah ukuran citra, melakukan normalisasi nilai *piksel*, mengurangi *derau* atau *denoising*, dan melakukan *augmentasi data*.
3. *Anotasi data*, yaitu proses memberikan label kelas dan *koordinat*

bounding box pada motif batik.

4. Deteksi objek, yaitu proses menentukan jenis dan lokasi motif menggunakan YOLOv8.
5. Ekstraksi warna, yaitu proses mengelompokkan *piksel* pada area hasil deteksi untuk menentukan warna dominan.

Tahapan tersebut merupakan bagian dari image processing yang mengubah citra masukan menjadi informasi digital. Melalui proses tersebut, karakteristik visual berupa garis, tepi, tekstur, bentuk, pola pengulangan, dan warna dapat diproses secara komputasional (Dani & Handayani, 2024).

b. Roboflow

Roboflow merupakan *platform* berbasis *web* yang dapat digunakan untuk membantu pengelolaan dataset dalam penelitian deteksi objek. *Platform* tersebut mendukung proses pemberian label kelas dan pembuatan *anotasi bounding box* pada objek yang terdapat dalam citra. Dataset yang telah dianotasi selanjutnya dapat dibagi menjadi data latih, data validasi, dan data uji untuk digunakan dalam proses pengembangan model YOLOv8 (Himawan, Rachmawanto, & Sari, 2025). Dalam penelitian ini, *Roboflow* digunakan untuk memberikan label kelas dan *anotasi bounding box* pada lima kelas motif, yaitu Batik Bali, Batik Betawi, Batik Garutan, Batik Parang, dan Batik Sekar. Anotasi tersebut menjadi ground truth yang menunjukkan jenis serta lokasi motif pada citra dan digunakan dalam proses pelatihan, validasi, serta pengujian model YOLOv8n.

c. Deteksi Objek

Deteksi objek merupakan proses untuk mengenali kelas objek sekaligus menentukan lokasi objek dalam suatu citra. Hasil deteksi umumnya terdiri atas label kelas, nilai *confidence*, dan koordinat *bounding box*. Deteksi *objek* berbeda dengan klasifikasi citra karena klasifikasi hanya menghasilkan kelas citra secara keseluruhan, sedangkan deteksi objek juga menunjukkan posisi *objek* yang dikenali

(Mao & Hong, 2025).

Koordinat bounding box dinyatakan sebagai:

$$B = (x_1, y_1, x_2, y_2)$$

Keterangan:

B = *koordinat bounding box*

x_1 dan y_1 = koordinat sudut kiri atas

x_2 dan y_2 = koordinat sudut kanan bawah\

Dalam penelitian ini, deteksi objek digunakan untuk menentukan kelas dan lokasi motif batik. *Area* yang berhasil dideteksi selanjutnya digunakan sebagai *Region of Interest*. Pembatasan *area* tersebut membuat proses *ekstraksi* warna hanya dilakukan pada bagian citra yang *relevan* dan mengurangi pengaruh warna latar belakang atau bagian nonmotif (Nalendra, Fisena, & Herdianto, 2025).

d. YOLO

You Only Look Once atau YOLO merupakan keluarga model deteksi objek berbasis *deep learning* yang melakukan klasifikasi dan pelokalan objek dalam satu tahap pemrosesan. Pendekatan tersebut disebut *single-stage detection*. Model memproses citra melalui satu jaringan untuk menghasilkan kelas objek dan koordinat lokasinya secara langsung (Terven, Córdova-Esparza, & Romero-González, 2023; Vijayakumar & Vairavasundaram, 2024). Istilah yang digunakan dalam bagian ini adalah *single-stage detection*, bukan *single shot detection*. Penulisan tersebut dipilih untuk membedakan mekanisme satu tahap pada YOLO dari nama model *Single Shot MultiBox Detector* atau SSD.

Model YOLO menghasilkan beberapa informasi utama, yaitu:

1. label kelas *objek*;
2. *probabilitas* kelas;
3. nilai *confidence*; dan
4. *koordinat bounding box*.

Keluaran awal model dapat menghasilkan beberapa *bounding box*

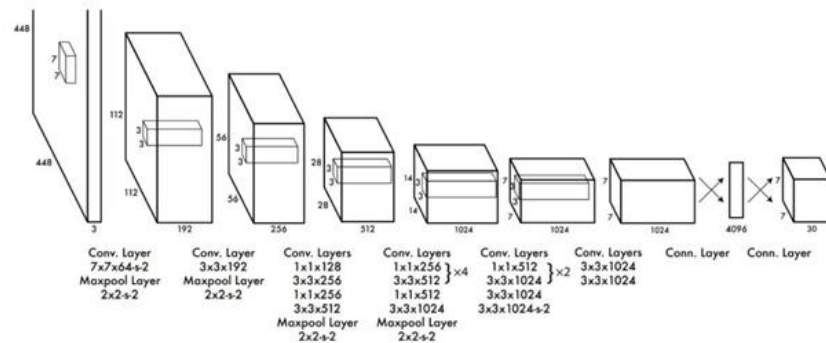
yang mendeteksi objek yang sama. *Prediksi* dengan nilai *confidence* rendah terlebih dahulu dihapus. Kotak yang masih saling bertumpang tindih kemudian diproses menggunakan *Non-Maximum Suppression* (NMS). Proses NMS mempertahankan kotak dengan *skor* tertinggi dan menghapus kotak lain yang memiliki tingkat tumpang tindih tinggi pada objek yang sama (Vijayakumar & Vairavasundaram, 2024).

e. Arsitektur dan Mekanisme Ultralytics YOLOv8

Ultralytics menyediakan paket atau *framework* yang mendukung berbagai tahapan pengembangan model, seperti pemuatan *dataset*, pelatihan, validasi, inferensi, dan penyimpanan atau ekspor model. YOLOv8 merupakan *arsitektur deteksi objek* yang dijalankan melalui *framework* tersebut. Dengan demikian, *Ultralytics* berfungsi sebagai lingkungan pengembangan dan pemrosesan, sedangkan YOLOv8 berfungsi sebagai model yang mempelajari karakteristik visual serta mendeteksi objek pada citra.

YOLOv8 dikembangkan oleh *Ultralytics* dan diperkenalkan pada tahun 2023. Arsitektur ini tersedia dalam beberapa varian, yaitu YOLOv8n, YOLOv8s, YOLOv8m, YOLOv8l, dan YOLOv8x. Dalam penelitian ini digunakan varian YOLOv8n karena merupakan varian paling ringan dan memiliki kebutuhan sumber daya komputasi yang lebih rendah dibandingkan varian lainnya sehingga sesuai digunakan pada perangkat dengan kemampuan komputasi terbatas (Yaseen, 2024). Model tersebut dilatih untuk mempelajari karakteristik visual lima kelas motif, yaitu Batik Bali, Batik Betawi, Batik Garutan, Batik Parang, dan Batik Sekar.

Arsitektur YOLOv8 terdiri atas tiga komponen utama, yaitu *backbone*, *neck*, dan *detection head*. Hubungan ketiga komponen tersebut ditunjukkan secara *eksplisit* pada Gambar 2.1.



Gambar 2. 1 Arsitektur YOLOv8

Sumber: <https://encord.com/blog/yolo-object-detection-guide/>
 Sumber: Diadaptasi dari Ultralytics (2023) dan (Terven et al., 2023)

1. Pra-pemrosesan Citra

Citra yang dimasukkan ke dalam model terlebih dahulu diperiksa format dan keterbacaannya. Citra kemudian disesuaikan menjadi ukuran 640×640 piksel agar memiliki dimensi yang seragam dengan ukuran masukan YOLOv8n. Nilai *piksel* citra dinormalisasi dan diubah menjadi bentuk tensor. Tensor tersebut menjadi masukan bagi jaringan *konvolusi* YOLOv8. *Pra-pemrosesan* tidak berfungsi mengenali kelas motif, tetapi memastikan citra memiliki bentuk dan skala nilai yang dapat diproses secara konsisten oleh model.

2. Ekstraksi Fitur pada Backbone

Bagian backbone berfungsi mengekstraksi fitur visual dari citra masukan dan mengubahnya menjadi beberapa peta fitur. Backbone YOLOv8 dibangun menggunakan lapisan konvolusi, modul C2f, dan modul *Spatial Pyramid Pooling Fast* atau SPPF (Hussain, 2024).

Lapisan awal pada backbone mengekstraksi fitur tingkat rendah, seperti:

- garis;
- tepi objek;
- perubahan intensitas;
- arah pola; dan

e) bentuk dasar *ornamen*.

Lapisan yang lebih dalam mempelajari *fitur* yang lebih kompleks, seperti tekstur kain, susunan ornamen, pola pengulangan, kemiripan bentuk, dan karakteristik visual yang membedakan setiap kelas motif.

Modul C2f membagi aliran *fitur* ke dalam beberapa jalur. Sebagian *fitur* diteruskan, sedangkan sebagian lainnya diproses melalui beberapa *blok konvolusi*. Seluruh hasil tersebut kemudian digabungkan kembali. Mekanisme ini membantu mempertahankan informasi dari lapisan sebelumnya dan memperluas aliran gradien selama proses pelatihan.

Pada bagian akhir backbone terdapat modul *Spatial Pyramid Pooling Fast (SPPF)*. Modul SPPF melakukan proses *pooling* pada beberapa cakupan *area*. Mekanisme tersebut memperluas bidang reseptif jaringan sehingga model dapat memahami *detail* lokal dan susunan pola secara lebih luas.

3. Penggabungan Fitur pada Neck

Peta *fitur* yang dihasilkan oleh *backbone* diteruskan ke bagian *neck*. Bagian *neck* menggabungkan fitur dari beberapa tingkat kedalaman melalui proses *upsampling*, *concatenation*, dan modul C2f.

Penggabungan tersebut menghasilkan tiga tingkat peta fitur, yaitu:

- a) P3/8 untuk *objek* berukuran *relatif* kecil;
- b) P4/16 untuk *objek* berukuran sedang; dan
- c) P5/32 untuk *objek* berukuran lebih besar.

Angka 8, 16, dan 32 menunjukkan tingkat pengurangan *resolusi* peta *fitur* dibandingkan citra masukan. P3 menyimpan informasi spasial yang lebih rinci, sedangkan P5 membawa informasi semantik yang lebih kuat. Penggabungan *fitur* multiskala membantu model mendeteksi motif dengan ukuran dan bentuk yang berbeda.

4. Prediksi pada Detection Head

Bagian *detection head* menerima *peta fitur* P3, P4, dan P5. YOLOv8 menggunakan *mekanisme anchor-free detection*, sehingga model tidak bergantung pada ukuran *anchor box* yang telah ditentukan sebelumnya.

Pada *detection head*, proses klasifikasi dan *regresi* lokasi dilakukan melalui cabang yang berbeda. Cabang klasifikasi menghitung kemungkinan kelas motif, sedangkan cabang *regresi* menentukan *koordinat* dan ukuran *bounding box*.

Hasil deteksi untuk objek ke- i ditulis sebagai:

$$D_i = (c_i, s_i, x_{1i}, y_{1i}, x_{2i}, y_{2i})$$

Keterangan:

D_i = hasil *deteksi objek* ke- i

c_i = kelas motif yang diprediksi

s_i = nilai *confidence*

x_{1i} dan y_{1i} = koordinat sudut kiri atas *bounding box*

x_{2i} dan y_{2i} = koordinat sudut kanan bawah *bounding box*

5. Penyaringan Hasil Deteksi

Pada tahap *inferensi*, YOLOv8 dapat menghasilkan beberapa kandidat deteksi pada *area* yang sama. Sistem menghapus prediksi dengan nilai *confidence* di bawah ambang yang ditentukan.

Prediksi yang masih saling bertumpang tindih diproses menggunakan *Non-Maximum Suppression*. Kotak dengan *skor* tertinggi dipertahankan, sedangkan kotak lain yang mendeteksi objek yang sama dihapus.

Hasil akhir YOLOv8 terdiri atas:

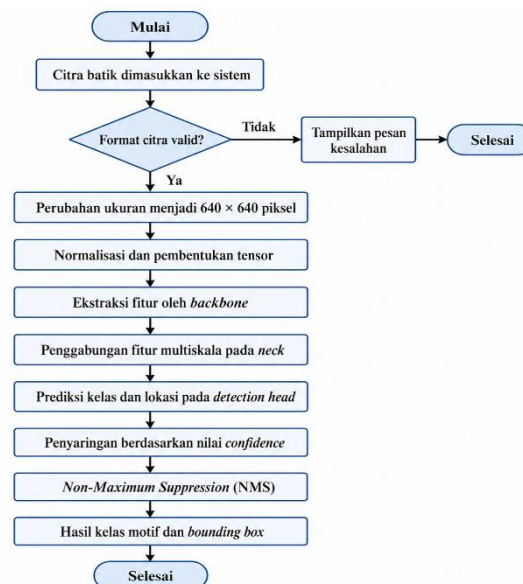
- nama kelas motif;
- nilai *confidence*;
- koordinat bounding box*; dan
- citra hasil deteksi.

Koordinat bounding box digunakan untuk mengambil *Region of*

Interest dari citra asli. ROI tersebut menjadi masukan untuk proses *K-Means Clustering*.

6. Bagan Alir Pemrosesan YOLOv8

Proses *inferensi* YOLOv8 meliputi validasi citra, penyesuaian ukuran, *ekstraksi fitur* melalui *backbone* dan *neck*, serta *prediksi objek* pada *detection head*. Hasil deteksi kemudian disaring menggunakan nilai *confidence* dan *Non-Maximum Suppression*. Alur proses tersebut ditunjukkan pada Gambar 2.2.



Gambar 2.2. Bagan Alir Pemrosesan Citra Menggunakan YOLOv8

Sumber: Diolah peneliti berdasarkan Ultralytics (2023) dan Terven et al. (2023)

f. Mekanisme K-Means Clustering pada Scikit-Learn

Scikit-Learn merupakan pustaka pembelajaran mesin yang menyediakan implementasi algoritma *K-Means* melalui kelas *K-Means*. Dalam penelitian ini, kelas *K-Means* digunakan untuk mengelompokkan nilai warna piksel pada area motif hasil deteksi berdasarkan kedekatan karakteristik warnanya. *K-Means* dapat digunakan untuk mengelompokkan piksel citra batik digital berdasarkan warna, sedangkan karakteristik bentuk dan tekstur

diperoleh melalui metode *ekstraksi fitur* yang berbeda (Zain, 2024). Oleh karena itu, *K-Means* dalam penelitian ini tidak digunakan untuk mengekstraksi bentuk, tekstur, atau pola motif karena proses *ekstraksi fitur visual* tersebut dilakukan oleh YOLOv8. *Fitur* yang diproses *K-Means* adalah tiga komponen warna pada setiap *piksel*, yaitu merah, hijau, dan biru

1. Pengambilan *Region of Interest*

YOLOv8 menghasilkan *koordinat bounding box*:

$$B = (x_1, y_1, x_2, y_2)$$

Koordinat tersebut digunakan untuk memotong citra asli. Hasil pemotongan disebut *Region of Interest*. Dengan *mekanisme* ini, *K-Means* hanya memproses *piksel* di dalam *area* motif dan tidak memproses seluruh citra.

2. Penyusunan Data *Piksel*

ROI memiliki tinggi H *piksel* dan lebar W *piksel*. Jumlah seluruh *piksel* dihitung dengan persamaan:

$$N = H \times W$$

Keterangan:

N = jumlah seluruh *piksel* pada ROI

H = tinggi ROI

W = lebar ROI

Setiap *piksel* direpresentasikan oleh tiga nilai warna:

$$x_i = (R_i, G_i, B_i)$$

Keterangan:

x_i = *piksel* ke- i

R_i = nilai kanal merah

G_i = nilai kanal hijau

B_i = nilai kanal biru

Seluruh *piksel* kemudian diubah menjadi data dua dimensi berukuran $N \times 3$. Setiap baris mewakili satu *piksel* dan ketiga kolom mewakili nilai merah, hijau, dan biru.

3. Inisialisasi Centroid

Jumlah *cluster* dalam penelitian ini ditetapkan sebanyak tiga.

Setiap *cluster* mempunyai satu pusat yang disebut *centroid*.

Pusat *cluster* ke-*j* ditulis sebagai:

$$\mu_j = (\mu R_j, \mu G_j, \mu B_j)$$

Keterangan:

μR_j = nilai merah pada *centroid* ke-*j*

μG_j = nilai hijau pada *centroid* ke-*j*

μB_j = nilai biru pada *centroid* ke-*j*

Scikit-Learn menentukan pusat awal berdasarkan parameter inisialisasi yang digunakan dalam kode program. Parameter *init*, *n_init*, *max_iter*, *tol*, dan *random_state* harus dituliskan sesuai dengan konfigurasi aplikasi yang sebenarnya.

4. Pengelompokan Piksel

Setiap *piksel* dibandingkan dengan seluruh *centroid*. Jarak antara *piksel* ke-*i* dan *centroid* ke-*j* dihitung menggunakan jarak *Euclidean*:

$$d(x_i, \mu_j) = \sqrt{[(R_i - \mu R_j)^2 + (G_i - \mu G_j)^2 + (B_i - \mu B_j)^2]}$$

Keterangan:

$d(x_i, \mu_j)$ = jarak piksel ke-*i* terhadap *centroid* ke-*j*

R_i, G_i, B_i = nilai RGB piksel ke-*i*

$\mu R_j, \mu G_j, \mu B_j$ = nilai RGB *centroid* ke-*j*

Piksel dimasukkan ke dalam *cluster* yang memiliki jarak paling kecil. *Piksel* dengan karakteristik warna yang berdekatan akan dikelompokkan dalam *cluster* yang sama.

5. Pembaruan Centroid

Setelah seluruh piksel memperoleh label *cluster*, nilai *centroid* diperbarui menggunakan rata-rata nilai RGB seluruh anggota *cluster*. Pembaruan tersebut dilakukan secara terpisah terhadap kanal merah, hijau, dan biru. Nilai merah pada *centroid* baru diperoleh dari rata-rata nilai merah seluruh anggota *cluster*. Proses

yang sama dilakukan pada kanal hijau dan biru.

6. Pengujian Konvergensi

Proses pengelompokan piksel dan pembaruan *centroid* dilakukan secara berulang. Proses berhenti ketika perubahan posisi *centroid* berada di bawah nilai toleransi atau jumlah iterasi maksimum telah tercapai. *K-Means* berusaha meminimalkan jumlah kuadrat jarak *piksel* terhadap pusat *clusternya*. Nilai tersebut disebut inerti. Nilai *inertia* yang lebih kecil menunjukkan bahwa *piksel* dalam satu *cluster* mempunyai karakteristik warna yang semakin berdekatan.

7. Pemilihan Warna Dominan

Setelah proses *K-Means* selesai, *Scikit-Learn* menghasilkan label *cluster* untuk setiap *piksel* dan nilai *centroid* setiap *cluster*. Sistem menghitung jumlah *piksel* yang menjadi anggota masing-masing *cluster*. *Cluster* dengan jumlah *piksel* terbesar dipilih sebagai *cluster* dominan. Nilai *centroid cluster* tersebut ditetapkan sebagai warna dominan:

Warna dominan = ($R_{dominan}$, $G_{dominan}$, $B_{dominan}$)

Nilai RGB dibulatkan menjadi bilangan bulat dalam rentang 0 sampai 255. Nilai tersebut kemudian dikonversi menjadi kode *heksadesimal*.

8. Alur Integrasi YOLOv8 dan *K-Means Clustering*

Integrasi YOLOv8 dan K-Means Clustering dimulai ketika pengguna memasukkan citra batik ke dalam sistem. Citra terlebih dahulu diperiksa untuk memastikan formatnya dapat dibaca, kemudian disesuaikan menjadi ukuran 640×640 piksel dan diubah menjadi bentuk tensor sebagai masukan model YOLOv8.

YOLOv8 memproses citra melalui tiga bagian utama, yaitu *backbone*, *neck*, dan *detection head*. Bagian *backbone* mengekstraksi fitur visual berupa garis, tepi, tekstur, bentuk ornamen, dan pola pengulangan motif. Fitur tersebut kemudian

digabungkan pada bagian *neck* dalam beberapa *skala*. Bagian *detection head* menghasilkan *prediksi* berupa kelas motif, nilai *confidence*, dan *koordinat bounding box*.

Hasil *prediksi* disaring berdasarkan nilai *confidence* dan diproses menggunakan *Non-Maximum Suppression* untuk menghapus beberapa kotak yang *mendeteksi objek* yang sama. Apabila motif tidak terdeteksi, sistem menampilkan informasi bahwa citra belum berhasil dikenali. Apabila motif terdeteksi, *koordinat bounding box* digunakan untuk mengambil *Region of Interest* dari citra asli.

Area Region of Interest kemudian *dikonversi* ke dalam format RGB. Setiap *piksel* direpresentasikan oleh nilai merah, hijau, dan biru, kemudian disusun menjadi data berukuran $N \times 3$. Nilai N menunjukkan jumlah seluruh piksel pada area motif, sedangkan tiga kolom menunjukkan komponen warna merah, hijau, dan biru. *K-Means Clustering* mengelompokkan *piksel* tersebut ke dalam tiga *cluster*. Proses dimulai dengan menentukan pusat awal atau *centroid*. Setiap piksel ditempatkan ke dalam *cluster* yang memiliki jarak paling dekat berdasarkan jarak *Euclidean*. Setelah seluruh *piksel* memperoleh label *cluster*, nilai *centroid* diperbarui menggunakan rata-rata nilai RGB anggota masing-masing *cluster*. Proses pengelompokan dan pembaruan *centroid* dilakukan secara berulang sampai perubahan pusat *cluster* berada di bawah nilai toleransi atau jumlah iterasi maksimum telah tercapai. Setelah proses *konvergen*, sistem menghitung jumlah *piksel* pada setiap *cluster*. *Cluster* dengan jumlah *piksel* terbanyak dipilih sebagai kelompok warna dominan.

Nilai *centroid* dari *cluster* dominan ditampilkan dalam bentuk nilai RGB dan kode heksadesimal. Hasil akhir sistem berupa nama motif batik, nilai *confidence*, lokasi motif dalam bentuk *bounding box*, serta informasi warna dominan pada area motif yang terdeteksi. *Mekanisme integrasi* tersebut disusun berdasarkan *arsitektur*

YOLOv8 dan proses pengelompokan *K-Means Clustering* (Terven et al., 2023; Vijayakumar & Vairavasundaram, 2024; Sinaga & Yang, 2020).

g. Analisis Warna Dominan

Warna dominan merupakan warna yang mempunyai jumlah *piksel* terbesar pada suatu *area* citra. Dalam penelitian ini, analisis warna dilakukan pada *Region of Interest* yang diperoleh dari *bounding box* hasil deteksi YOLOv8.

Piksel dalam ROI dikelompokkan menjadi tiga *cluster* berdasarkan kedekatan nilai RGB. Cluster dengan jumlah anggota terbesar dipilih sebagai *cluster* dominan. Nilai centroid dari cluster tersebut ditetapkan sebagai warna dominan.

Pembatasan analisis pada ROI mengurangi pengaruh warna latar belakang dan bagian citra di luar motif. Hasil warna dominan ditampilkan dalam bentuk nilai RGB, kode *heksadesimal*, dan visualisasi warna (Zain & Sumijan, 2024).

h. Evaluasi Model Deteksi Objek

Evaluasi dilakukan untuk mengukur kemampuan YOLOv8 dalam mendeteksi motif batik. Metrik yang digunakan meliputi *precision*, *recall*, *mAP50*, dan *mAP50–95*.

Precision mengukur proporsi hasil *prediksi positif* yang sesuai dengan objek sebenarnya. Nilai *precision* yang tinggi menunjukkan bahwa model menghasilkan lebih sedikit *false positive*.

Recall mengukur proporsi objek sebenarnya yang berhasil ditemukan oleh model. Nilai *recall* yang tinggi menunjukkan bahwa model dapat mendeteksi sebagian besar objek yang tersedia.

mAP50 merupakan rata-rata *Average Precision* pada *ambang Intersection over Union sebesar 0,50*. *mAP50–95* merupakan rata-rata *Average Precision* pada beberapa *ambang IoU* dari 0,50 sampai 0,95. Metrik tersebut digunakan untuk menggambarkan ketepatan klasifikasi, kemampuan menemukan objek, dan kesesuaian lokasi

bounding box hasil *prediksi* terhadap *anotasi* sebenarnya (Vijayakumar & Vairavasundaram, 2024; Mao & Hong, 2025).

2. Kajian Pustaka

Kajian pustaka dilakukan untuk membandingkan metode, jenis data, hasil, kelebihan, dan keterbatasan penelitian terdahulu yang berkaitan dengan pengenalan motif serta analisis warna pada citra batik. Perbandingan tersebut digunakan untuk menentukan posisi dan kontribusi penelitian yang mengintegrasikan YOLOv8 dengan *K-Means Clustering*.

- a. Dani dan Handayani (2024) memadukan Gray Level Co-occurrence Matrix (GLCM) dengan CNN berbasis MobileNetV2 untuk mengklasifikasikan lima motif batik Yogyakarta. Dataset yang digunakan berjumlah 3.223 citra yang mencakup motif Ceplok, Kawung, Truntum, Parang, dan Ciptoning. Kombinasi fitur tekstur dan pembelajaran fitur visual menghasilkan akurasi sebesar 99%. Keunggulan penelitian tersebut terletak pada pemanfaatan GLCM untuk menangkap tekstur halus dan MobileNetV2 untuk menjaga efisiensi komputasi. Meskipun menghasilkan akurasi tinggi, keluaran model masih berupa kelas citra secara keseluruhan. Sistem belum menunjukkan lokasi motif melalui *bounding box* dan belum melakukan ekstraksi warna dominan pada area tertentu.
- b. Prayoga . (2023) mengevaluasi tujuh rancangan *arsitektur* CNN untuk mengklasifikasikan 600 citra Batik Yogyakarta yang terdiri atas motif Ceplok, Kawung, dan Parang. Model keenam menghasilkan *performa* terbaik dengan akurasi 87,83%, rata-rata *precision* 88,46%, dan rata-rata *recall* 87,66%. Penelitian tersebut menunjukkan bahwa susunan lapisan *ekstraksi fitur* berpengaruh terhadap kemampuan CNN dalam membedakan pola batik. Keterbatasannya terletak pada jumlah kelas yang hanya mencakup tiga motif dan penggunaan pendekatan klasifikasi *global*. Model tidak dirancang untuk menentukan posisi motif maupun menganalisis warna pada objek yang dikenali.
- c. Maulana . (2023) menerapkan CNN berbasis *transfer learning* pada aplikasi *Android* untuk mengklasifikasikan empat kelas motif, yaitu Corak Insang, Dayak, Ikat Celup, dan Megamendung. Data penelitian

terdiri atas 1.320 citra latih, 80 citra validasi, dan 120 citra uji. Penggunaan aplikasi *Android* memperlihatkan bahwa model klasifikasi motif dapat diterapkan pada perangkat yang berinteraksi langsung dengan pengguna. Sistem juga menggunakan ambang keyakinan sebesar 70% untuk membedakan kelas motif dan kategori objek lainnya. Walaupun memiliki nilai praktis dalam bentuk aplikasi, penelitian tersebut belum menyediakan pelokalan objek dan belum menghubungkan hasil klasifikasi dengan proses ekstraksi warna dominan.

- d. Berbeda dari pendekatan CNN, Azizah dan Agustin (2025) menggunakan fitur *Histogram of Oriented Gradients* (HOG) dan *texture moments* yang diklasifikasikan melalui *Random Forest*. HOG menghasilkan akurasi 78,99%, sedangkan *texture moments* menghasilkan 81,88%. Penggabungan kedua fitur meningkatkan akurasi menjadi 86,23%. Hasil tersebut membuktikan bahwa kombinasi fitur bentuk dan tekstur dapat memperbaiki performa klasifikasi dibandingkan penggunaan satu jenis fitur. Kelemahannya adalah fitur masih dirancang secara manual sehingga kemampuannya bergantung pada jenis karakteristik yang dipilih. Pendekatan ini juga belum mendeteksi lokasi motif dan belum mengolah warna pada area hasil deteksi.
- e. Fokus pada fitur warna ditunjukkan oleh Zain dan Sumijan (2024) melalui penggunaan Canny, GLCM, LBP, HSV color moment, dan *K-Means Clustering*. Canny, GLCM, serta LBP digunakan untuk memperoleh karakteristik bentuk dan tekstur, sedangkan HSV *color moment* dan *K-Means* digunakan untuk menganalisis serta mengelompokkan warna citra batik. Penelitian tersebut relevan karena menunjukkan bahwa *K-Means* dapat mengelompokkan piksel berdasarkan kedekatan warna. Meskipun demikian, proses *K-Means* masih diterapkan dalam konteks citra batik secara umum dan tidak didahului oleh deteksi lokasi motif. Hasil penelitian juga belum

- menyajikan evaluasi kuantitatif deteksi objek seperti *precision*, *recall*, *mAP50*, dan *mAP50–95*.
- f. Whardana dan Aziz (2025) menempatkan proses pra-pemrosesan sebagai bagian penting dalam klasifikasi motif. Citra yang diperoleh dari *internet* dan *Museum Batik TMII* dikonversi menjadi citra keabuan, *disegmentasi* menggunakan *Otsu*, dan diperbaiki dengan median *filter* sebelum diklasifikasikan menggunakan CNN. Model dilatih selama 3.600 iterasi dengan *batch size* 20 dan menghasilkan akurasi 85,36%, rata-rata *precision* 86,76%, serta *recall* 87,36%. Pendekatan ini mampu mengurangi derau dan memisahkan motif dari latar sebelum klasifikasi. Namun, segmentasi *Otsu* bekerja berdasarkan pemisahan intensitas dan belum menghasilkan kelas serta lokasi objek secara bersamaan seperti model deteksi objek.

Berdasarkan enam penelitian tersebut, metode CNN dan *ekstraksi fitur tekstur* telah memberikan hasil yang baik dalam klasifikasi motif batik. Sebagian besar penelitian masih memproses citra secara keseluruhan sehingga hanya menghasilkan label kelas tanpa menunjukkan lokasi motif. Analisis warna juga umumnya dilakukan secara terpisah dan belum memanfaatkan area objek yang telah dideteksi.

Penelitian ini menempati celah tersebut dengan menggunakan YOLOv8 untuk menghasilkan label kelas, nilai *confidence*, dan *koordinat bounding box*. *Koordinat* hasil deteksi digunakan untuk mengambil *Region of Interest*, kemudian *K-Means Clustering* diterapkan secara khusus pada *area* tersebut untuk memperoleh warna dominan. Perbedaan utama penelitian ini terletak pada *integrasi* deteksi lokasi motif dan *eksi warna* karna dominan dalam satu alur pemrosesan citra.

Tabel 2.1. Matriks Perbandingan Penelitian Terdahulu

No	Peneliti dan Tahun	Metode	Data atau Objek	Hasil dan Kelebihan	Keterbatasan
1	Dani dan Handayani (2024)	GLCM dan CNN berbasis MobileNet V2	3.223 citra, lima motif Batik Yogyakarta	Akurasi 99%; mampu menggabungkan fitur tekstur dan fitur visual dengan komputasi relatif ringan	Hanya melakukan klasifikasi global; tidak menghasilkan bounding box dan warna dominan
2	Prayoga . (2023)	Tujuh rancangan arsitektur CNN	600 citra motif Ceplok, Kawung , dan Parang	Model terbaik memperoleh akurasi 87,83%, <i>precision</i> 88,46%, dan <i>recall</i> 87,66%	Jumlah kelas terbatas dan tidak melakukan pelokalan motif
3	Maulana (2023)	CNN, transfer learning, dan aplikasi Android	1.320 data latih, 80 validasi, 120 uji; empat kelas motif	Berhasil diterapkan pada aplikasi Android dan menggunakan ambang keyakinan untuk kategori objek lainnya	Keluaran hanya berupa kelas; tidak menghasilkan lokasi motif dan warna dominan

4	Azizah dan Agustin (2025)	HOG, texture moments, dan <i>Random Forest</i>	Dataset citra pola batik	Kombinasi fitur menghasilkan akurasi 86,23%, lebih tinggi daripada HOG 78,99% dan <i>texture moments</i> 81,88%	Menggunakan fitur buatan tangan dan belum mendeteksi lokasi objek
5	Zain dan Sumijan (2024)	Canny, GLCM, LBP, HSV color moment, dan K-Means	Citra batik dengan karakteristik tekstur dan warna	Mengintegrasikan fitur tekstur dan pengelompokan warna menggunakan <i>K-Means</i>	<i>K-Means</i> belum diterapkan pada ROI hasil deteksi dan belum dievaluasi dengan metrik deteksi objek
6	Whardana dan Aziz (2025)	Otsu, median filter, dan CNN	Citra dari internet dan <i>Museum Batik TMII</i>	Akurasi 85,36%, <i>precision</i> 86,76%, dan <i>recall</i> 87,36%; <i>pra-pemrosesan</i>	Segmentasi dan klasifikasi masih terpisah serta belum menghasilkan <i>bounding box</i>

				membantu mengurangi derau	
7	Penelitian yang dilakukan	YOLOv8 dan <i>K-Means Clustering</i>	586 citra, lima kelas motif batik	Menghasilkan kelas, <i>confidence</i> , <i>bounding box</i> , nilai RGB, dan kode heksadesimal dalam satu sistem	<i>Performa</i> bergantung pada jumlah data, kualitas <i>anotasi</i> , pencahayaan, dan <i>area</i> hasil deteksi

Berdasarkan Tabel 2.1, penelitian terdahulu didominasi pendekatan klasifikasi citra yang menghasilkan label kelas tanpa menentukan lokasi motif. Penelitian ini menggunakan YOLOv8 untuk mendeteksi kelas dan lokasi motif, kemudian menerapkan *K-Means Clustering pada Region of Interest*. Integrasi tersebut membuat analisis warna lebih terfokus pada *area* motif dan mengurangi pengaruh bagian citra di luar objek.

B. Kerangka Berpikir

Kerangka berpikir dalam penelitian ini menjelaskan hubungan antara permasalahan, data masukan, proses komputasi, keluaran sistem, dan ukuran keberhasilan penelitian. Permasalahan utama yang dikaji adalah belum tersedianya sistem yang dapat mendeteksi jenis dan lokasi motif batik sekaligus mengekstraksi warna dominan secara terfokus pada area motif yang terdeteksi.

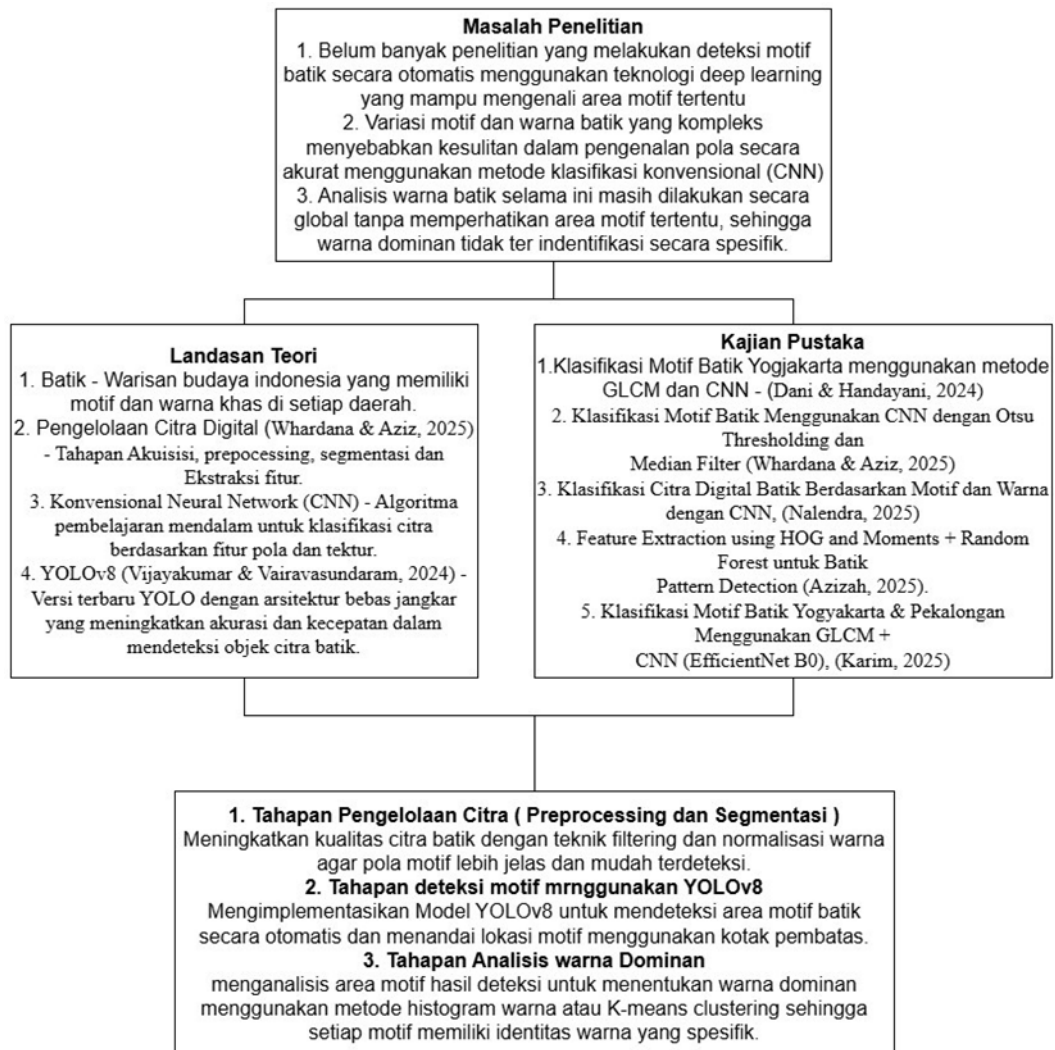
Data yang digunakan berupa 586 citra batik yang terdiri atas lima kelas, yaitu Batik Bali, Batik Betawi, Batik Garutan, Batik Parang, dan Batik Sekar. Dataset dibagi menjadi data latih, data validasi, dan data uji. Setiap citra diberi *anotasi kelas* dan *koordinat bounding box* agar dapat digunakan dalam proses pelatihan YOLOv8.

Tahap pemrosesan dimulai dengan pra-pemrosesan citra yang meliputi pemeriksaan data, perubahan ukuran menjadi 640×640 piksel, *normalisasi*, *augmentasi*, dan *anotasi*. Dataset yang telah diproses digunakan untuk melatih model YOLOv8n. Model mempelajari fitur visual motif melalui bagian *backbone*, menggabungkan fitur multiskala melalui *neck*, dan menghasilkan prediksi kelas serta lokasi objek melalui *detection head*.

Hasil deteksi YOLOv8 berupa label kelas, nilai *confidence*, dan koordinat *bounding box*. Koordinat tersebut digunakan untuk mengambil *Region of Interest* dari citra asli. Piksel pada area hasil deteksi kemudian dikelompokkan menjadi tiga cluster menggunakan *K-Means Clustering*. Cluster yang mempunyai jumlah *Piksel* terbanyak dipilih sebagai warna dominan, kemudian nilai pusat *cluster* ditampilkan dalam bentuk RGB dan kode heksadesimal.

Ukuran keberhasilan penelitian ditentukan melalui pengujian *kuantitatif* dan pengujian *fungsional*. Performa deteksi motif diukur menggunakan *precision*, *recall*, *mAP50*, dan *mAP50–95*. Proses ekstraksi warna dievaluasi berdasarkan kemampuan sistem menghasilkan kelompok warna dominan pada area motif dan menampilkannya dalam bentuk nilai RGB serta kode heksadesimal. Pengujian *fungsional* dilakukan untuk memastikan aplikasi dapat menerima citra, menjalankan deteksi, mengambil area motif, mengekstraksi warna dominan, dan menampilkan seluruh hasil melalui antarmuka *Streamlit*.

Dengan kerangka tersebut, penelitian tidak hanya diarahkan untuk menghasilkan aplikasi, tetapi juga memperoleh hasil yang dapat diukur secara objektif melalui metrik performa model dan keluaran sistem.



Gambar 2. 2 Kerangka Berpikir

34

BAB III

METODE PENELITIAN

A. Desain Penelitian

49

Penelitian berjudul “Deteksi Motif dan Warna Dominan Batik Menggunakan YOLOv8” menggunakan pendekatan kuantitatif dengan metode *eksperimen* dan rekayasa perangkat lunak. Pendekatan kuantitatif digunakan karena performa model diukur secara numerik melalui nilai *precision*, *recall*, *mAP50*, dan *mAP50–95*. Metode eksperimen digunakan dalam proses pengolahan *dataset*, pelatihan, validasi, dan pengujian model YOLOv8. Sementara itu, rekayasa perangkat lunak digunakan untuk mengembangkan aplikasi berbasis *Streamlit* yang mengintegrasikan model YOLOv8 dan *K-Means Clustering*.

Desain penelitian ini termasuk desain hibrida karena menggabungkan *eksperimen* data citra dengan tahapan pengembangan aplikasi. *Eksperimen* data citra bertujuan menghasilkan model deteksi terbaik dalam bentuk berkas *best.pt*, sedangkan pengembangan perangkat lunak bertujuan menghasilkan aplikasi yang dapat menerima citra, menjalankan deteksi, mengekstraksi warna dominan, dan menampilkan hasil kepada pengguna.

8

Pengembangan aplikasi dilakukan secara bertahap menggunakan pendekatan SDLC model *Waterfall*, yang meliputi analisis kebutuhan, perancangan sistem, implementasi, integrasi model, dan pengujian. Tahapan tersebut dihubungkan dengan proses *eksperimen* model YOLOv8 sebagaimana dijelaskan berikut.

84

1. Pengumpulan Data

2

Data yang digunakan dalam penelitian ini berupa 586 citra batik yang terdiri atas lima kelas, yaitu Batik Bali, Batik Betawi, Batik Garutan, Batik Parang, dan Batik Sekar. Citra diperoleh melalui dokumentasi langsung dan *sumber daring* yang relevan.

Setiap citra diperiksa untuk memastikan gambar dapat dibaca, motif terlihat jelas, tidak mengalami kerusakan berkas, dan sesuai dengan kelas yang digunakan dalam penelitian. Citra yang tidak memenuhi kriteria tersebut tidak dimasukkan ke dalam *dataset*.

Salah satu contoh citra yang digunakan sebagai data penelitian ditunjukkan pada Gambar 3.1.



Gambar 3. 1 Batik Motif Pesisiran

Gambar 3.1 menunjukkan contoh citra batik yang digunakan sebagai masukan dalam proses anotasi dan pelatihan model. Citra tersebut selanjutnya diberi label kelas dan *koordinat bounding box* pada area motif yang menjadi objek penelitian.

2. Pra-Pemrosesan Data

Data yang telah dikumpulkan kemudian melalui tahap pra-pemrosesan agar sesuai dengan format masukan YOLOv8. Tahapan pra-pemrosesan meliputi pemeriksaan format citra, perubahan ukuran menjadi 640×640 piksel, normalisasi nilai piksel, serta pemeriksaan kesesuaian label dan anotasi.

Setiap objek motif diberi anotasi berupa nama kelas dan *koordinat bounding box* menggunakan perangkat seperti *LabelImg* atau *Roboflow*. Hasil anotasi disimpan dalam format YOLO.

Dataset kemudian dibagi menjadi:

- 513 citra sebagai data latih;
- 49 citra sebagai data validasi; dan
- 24 citra sebagai data uji.

Augmentasi diterapkan pada data latih untuk meningkatkan variasi citra. *Transformasi* yang digunakan meliputi *rotasi*, pembalikan citra, dan perubahan tingkat kecerahan. *Transformasi* tersebut dilakukan tanpa mengubah *identitas* kelas motif. Contoh proses *pra-pemrosesan* dan *augmentasi* ditunjukkan pada Gambar 3.2.



Gambar 3. 2 Tahapan Pra-Pemrosesan dan
Augmentasi Citra Batik

Gambar 3.2 memperlihatkan perubahan citra melalui proses pembalikan, *rotasi*, dan penyesuaian kecerahan. *Augmentasi* digunakan agar model mempelajari motif dalam kondisi visual yang lebih beragam dan tidak terbatas pada satu posisi atau tingkat pencahayaan

3. Pelatihan dan Validasi Model YOLOv8

Model deteksi objek yang digunakan dalam penelitian ini adalah YOLOv8n dengan bobot awal pralatih. Varian YOLOv8n dipilih karena memiliki ukuran model yang relatif ringan dan kebutuhan komputasi yang lebih rendah sehingga sesuai dengan spesifikasi perangkat yang digunakan dalam penelitian.

Pelatihan model dilakukan menggunakan 513 citra data latih yang telah diberi label kelas dan anotasi bounding box. Konfigurasi pelatihan yang digunakan meliputi 50 epoch, batch size 16, ukuran masukan 640×640 piksel, dan optimizer AdamW.

Data latih digunakan untuk memperbarui bobot model selama proses

pelatihan. Pada setiap epoch, YOLOv8n mempelajari karakteristik visual motif melalui bagian backbone, menggabungkan fitur dari beberapa skala melalui bagian neck, serta menghasilkan prediksi kelas dan koordinat bounding box melalui detection head.

Data validasi yang terdiri atas 49 citra dengan 57 objek digunakan untuk memantau performa model selama proses pelatihan. Proses validasi tidak digunakan untuk memperbarui bobot model, tetapi digunakan untuk mengukur kemampuan model dalam mengenali kelas dan menentukan lokasi motif pada citra yang tidak digunakan dalam pembaruan bobot.

Metrik yang diamati dalam proses validasi meliputi:

- precision*, untuk mengukur ketepatan prediksi positif;
- recall*, untuk mengukur kemampuan model menemukan objek;
- mAP50*, untuk mengukur performa pada ambang *IoU* 0,50; dan
- mAP50–95*, untuk mengukur performa pada beberapa ambang *IoU* dari 0,50 sampai 0,95.

Selain metrik tersebut, proses pelatihan juga menghasilkan nilai box loss, classification loss, dan distribution focal loss. Nilai loss digunakan untuk memantau perkembangan pembelajaran model, sedangkan *precision*, *recall*, *mAP50*, dan *mAP50–95* digunakan untuk mengevaluasi performa deteksi.

Model dengan hasil validasi terbaik disimpan secara otomatis dalam berkas best.pt. Berkas tersebut dipilih sebagai model final dan digunakan pada aplikasi Streamlit untuk menjalankan proses deteksi motif batik.

Data uji sebanyak 24 citra digunakan secara terpisah setelah model final diperoleh. Data tersebut digunakan untuk menjalankan inferensi dan memeriksa kemampuan implementasi model pada citra yang tidak digunakan dalam proses pembaruan bobot. Keluaran inferensi berupa nama kelas motif, nilai confidence, dan koordinat bounding box.

Nilai *precision*, *recall*, *mAP50*, dan *mAP50–95* yang disajikan pada BAB IV berasal dari hasil validasi model terbaik terhadap 49 citra dengan 57 objek. Data uji sebanyak 24 citra tidak digunakan sebagai sumber nilai

pada Tabel 4.2 dan Tabel 4.3, tetapi digunakan untuk pengujian inferensi dan implementasi aplikasi.

Apabila performa model belum memenuhi kebutuhan penelitian, dilakukan pemeriksaan ulang terhadap dataset, pembagian data, kualitas anotasi bounding box, proses pra-pemrosesan, dan konfigurasi pelatihan. Setelah perbaikan dilakukan, model dapat dilatih kembali untuk memperoleh hasil yang lebih baik.

4. Ekstraksi Warna Dominan

Model terbaik hasil pelatihan yang tersimpan dalam berkas best.pt digunakan dalam aplikasi berbasis Streamlit. Setelah citra diproses menggunakan YOLOv8n, koordinat bounding box digunakan untuk mengambil bagian citra yang memuat objek motif. Bagian citra tersebut disebut Region of Interest atau ROI.

ROI kemudian dikonversi ke dalam format warna RGB. Setiap piksel direpresentasikan oleh tiga komponen warna, yaitu merah (red), hijau (green), dan biru (blue). Data piksel disusun menjadi matriks dua dimensi berukuran $N \times 3$, dengan N menunjukkan jumlah piksel pada ROI, sedangkan tiga kolom menunjukkan nilai R, G, dan B.

Piksel pada ROI dikelompokkan menggunakan K-Means Clustering. Konfigurasi yang digunakan dalam proses tersebut adalah:

- $n_clusters = 3$, untuk membagi piksel menjadi tiga kelompok warna;
- $random_state = 0$, untuk menjaga konsistensi hasil inisialisasi; dan
- $n_init = 10$, untuk menjalankan inisialisasi pusat cluster sebanyak sepuluh kali dan memilih hasil terbaik.

Inisialisasi pusat cluster dilakukan menggunakan metode `k-means++`. Setiap piksel ditempatkan ke dalam cluster yang memiliki jarak terdekat berdasarkan jarak Euclidean. Setelah seluruh piksel memperoleh label cluster, pusat setiap cluster diperbarui berdasarkan rata-rata nilai RGB seluruh anggotanya.

Proses pengelompokan dan pembaruan pusat cluster dilakukan secara berulang sampai mencapai kondisi konvergen. Selanjutnya, jumlah

piksel pada setiap cluster dihitung. Cluster dengan jumlah piksel terbanyak ditetapkan sebagai kelompok warna dominan.

Nilai pusat cluster dominan kemudian dibulatkan menjadi bilangan bulat dalam rentang 0 sampai 255. Nilai tersebut disajikan dalam bentuk RGB, kode heksadesimal, dan kotak visual warna pada antarmuka aplikasi.

Ekstraksi warna dilakukan pada ROI hasil deteksi, bukan pada keseluruhan citra. Pendekatan tersebut digunakan untuk mengurangi pengaruh warna latar belakang dan bagian citra di luar objek motif. Hasil ekstraksi warna tetap dapat dipengaruhi oleh pencahayaan, kualitas citra, bayangan, dan ketepatan posisi bounding box.

5. Evaluasi dan Kesimpulan

Tahap evaluasi dilakukan untuk mengetahui performa model YOLOv8n dan kemampuan aplikasi dalam menjalankan fungsi yang telah dirancang. Evaluasi penelitian meliputi evaluasi model, pengujian fungsional aplikasi, pengujian non-fungsional, dan analisis hasil ekstraksi warna dominan.

Evaluasi model dilakukan menggunakan hasil validasi model terbaik yang tersimpan dalam berkas best.pt. Data validasi terdiri atas 49 citra dengan 57 objek dari lima kelas motif batik. Metrik yang digunakan meliputi precision, recall, mAP50, dan mAP50–95. Selain itu, grafik hasil pelatihan dan confusion matrix digunakan untuk melihat perkembangan pembelajaran serta distribusi prediksi pada setiap kelas.

Pengujian fungsional aplikasi dilakukan menggunakan metode black-box testing. Pengujian bertujuan mengetahui kesesuaian antara fungsi yang dirancang dan hasil aktual yang ditampilkan aplikasi. Skenario pengujian mencakup kondisi aplikasi tanpa citra, unggah citra dengan format yang didukung, deteksi citra batik, pengujian citra bukan motif batik, pengujian berkas citra rusak, dan pemrosesan beberapa citra secara bergantian.

Pengujian non-fungsional dilakukan untuk mengukur waktu respons dan stabilitas proses inti sistem. Pengujian menggunakan 45 citra khusus benchmark. Setiap citra diproses sebanyak 10 kali sehingga diperoleh 450 proses pengujian. Sebelum pengukuran utama, model dijalankan sebanyak tiga kali sebagai tahap pemanasan.

Parameter pengujian non-fungsional meliputi waktu pemuatan model, waktu inferensi YOLOv8, waktu K-Means Clustering, waktu total pemrosesan, waktu tercepat, waktu terlama, throughput, dan jumlah proses yang berhasil menghasilkan catatan waktu.

Hasil ekstraksi warna dominan dianalisis berdasarkan kemampuan sistem dalam mengambil ROI, mengelompokkan piksel menjadi tiga cluster, memilih cluster dengan jumlah piksel terbanyak, serta

menampilkan warna dominan dalam bentuk nilai RGB, kode heksadesimal, dan kotak visual warna. Penelitian ini tidak menghitung tingkat akurasi warna karena belum menggunakan warna acuan atau ground truth sebagai pembanding.

Seluruh hasil evaluasi selanjutnya dianalisis untuk mengetahui performa model secara keseluruhan, perbedaan performa setiap kelas, keberhasilan fungsi aplikasi, waktu pemrosesan, dan keterbatasan sistem. Hasil analisis tersebut digunakan sebagai dasar dalam penyusunan kesimpulan dan saran pengembangan penelitian.

B. Instrumen Penelitian

1. Instrumen Perangkat Keras (*Hardware*)

Perangkat keras digunakan untuk menjalankan proses pra-pemrosesan citra, pelatihan dan pengujian model YOLOv8, ekstraksi warna dominan menggunakan *K-Means Clustering*, serta pengoperasian aplikasi berbasis *Streamlit*. Spesifikasi perangkat keras yang digunakan dalam penelitian ini adalah sebagai berikut:

- a. Nama perangkat: LAPTOP-0V5EC325.
- b. Prosesor: 11th Gen Intel® Core™ i5-11400H dengan kecepatan 2,70 GHz.
- c. Memori utama: RAM 8 GB dengan kapasitas yang dapat digunakan sebesar 7,74 GB.
- d. Kartu grafis diskrit: NVIDIA GeForce RTX 3050 Laptop GPU dengan memori 4 GB.
- e. Kartu grafis terintegrasi: Intel® UHD Graphics.
- f. Penyimpanan: SSD berkapasitas 477 GB.
- g. Sistem operasi: Windows 11 Home Single Language versi 25H2.
- h. Arsitektur sistem: sistem operasi 64-bit dengan prosesor berbasis x64.

Spesifikasi tersebut merupakan perangkat yang digunakan selama proses pengembangan, pengujian model, dan pengoperasian aplikasi.

2. Perangkat Lunak (*Software*)

Perangkat lunak yang digunakan dalam proses pengembangan sistem adalah sebagai berikut:

- a. Python versi 3.12.0 sebagai bahasa pemrograman utama yang digunakan untuk membangun dan menjalankan sistem.
- b. *Visual Studio Code* versi 1.124.2 sebagai editor kode yang digunakan untuk menulis, menjalankan, dan mengelola program.
- c. Ultralytics versi 8.4.53 digunakan untuk melakukan pelatihan, validasi, dan inferensi model YOLOv8n.
- d. PyTorch versi 2.11.0 sebagai pustaka pendukung komputasi tensor dan pembelajaran mendalam.
- e. OpenCV versi 4.13.0.92 sebagai pustaka yang digunakan untuk membaca, mengubah ukuran, memotong, dan mengolah citra digital.
- f. NumPy versi 2.4.4 sebagai pustaka yang digunakan untuk mengolah data numerik, matriks citra, dan nilai piksel.
- g. *Scikit-Learn* versi 1.8.0 sebagai pustaka yang digunakan untuk menerapkan *K-Means*
- h. *Streamlit* versi 1.57.0 sebagai *framework* yang digunakan untuk membangun antarmuka aplikasi berbasis web.
- i. Matplotlib versi 3.10.9 sebagai pustaka yang digunakan untuk menampilkan grafik, visualisasi hasil pelatihan, dan *confusion matrix*.
- j. *Pillow* versi 12.2.0 sebagai pustaka pendukung untuk membaca, mengonversi, dan menampilkan citra.
- k. *Roboflow* berbasis web sebagai platform yang digunakan untuk memberikan label kelas dan membuat anotasi *bounding box* pada citra dataset.

Seluruh kode program dikembangkan menggunakan *Visual Studio Code* dan dijalankan pada lingkungan virtual *Python* yang berada di dalam folder proyek. Proses anotasi data dilakukan melalui website *Roboflow*, sedangkan pengembangan dan pengujian aplikasi dilakukan secara lokal tanpa menggunakan *Jupyter Notebook* maupun *Google Colab*.

3. Dataset Citra Batik

Dataset merupakan instrumen utama yang digunakan dalam proses pelatihan, validasi, dan pengujian model. Dataset penelitian berjumlah 586 citra batik yang terdiri atas lima kelas, yaitu Batik Bali, Batik Betawi,

Batik Garutan, Batik Parang, dan Batik Sekar.

Dataset dibagi menjadi:

- a. 513 citra sebagai data latih;
- b. 49 citra sebagai data validasi; dan
- c. 24 citra sebagai data uji.

Setiap citra diberikan anotasi berupa label kelas dan *koordinat bounding box* menggunakan *Roboflow* berbasis *web*. Hasil anotasi disimpan dalam format YOLO. Data latih digunakan untuk membentuk model, data validasi digunakan untuk memantau performa selama pelatihan, sedangkan data uji digunakan untuk menilai kemampuan model terhadap citra yang tidak digunakan dalam pembaruan bobot.

4. Evaluasi Model

Evaluasi model dilakukan untuk mengukur kemampuan sistem dalam mendeteksi motif batik. Instrumen evaluasi yang digunakan meliputi:

- a. *precision*, untuk mengukur ketepatan hasil prediksi positif;
- b. *recall*, untuk mengukur kemampuan model menemukan objek;
- c. *mAP50*, untuk mengukur performa deteksi pada ambang *IoU* sebesar 0,50;
- d. *mAP50–95*, untuk mengukur performa pada beberapa ambang *IoU* mulai dari 0,50 sampai 0,95; dan
- e. *confusion matrix*, untuk menunjukkan distribusi prediksi benar dan salah pada setiap kelas.

Hasil evaluasi disajikan dalam bentuk tabel, grafik, dan *confusion matrix*. Pengolahan hasil evaluasi dilakukan menggunakan *Scikit-Learn* dan *Matplotlib*

5. Instrumen Dokumentasi Penelitian

Instrumen dokumentasi digunakan untuk mencatat seluruh proses dan hasil penelitian. Dokumen yang dikumpulkan meliputi:

- a. citra dataset dan berkas anotasi;
- b. konfigurasi pelatihan model;
- c. model hasil pelatihan;

- d. grafik hasil pelatihan dan validasi;
- e. hasil *confusion matrix*;
- f. tangkapan layar antarmuka aplikasi;
- g. hasil deteksi berupa label, nilai *confidence*, dan *bounding box*;
- h. hasil warna dominan berupa nilai RGB dan kode heksadesimal; dan
- i. catatan hasil pengujian fungsional dan non-fungsional.

6. Instrumen Pengujian Fungsional

Pengujian fungsional dilakukan menggunakan metode *black-box testing*. Pengujian bertujuan memastikan setiap fungsi aplikasi berjalan sesuai kebutuhan. Pengujian tidak hanya dilakukan pada kondisi masukan yang benar, tetapi juga mencakup masukan kosong, format yang tidak didukung, berkas rusak, motif tidak terdeteksi, dan kegagalan pemrosesan.

7. Instrumen Pengujian Non-Fungsional

Pengujian non-fungsional digunakan untuk mengukur waktu respons dan stabilitas proses inti sistem. Pengujian dilakukan menggunakan 45 citra khusus benchmark yang dibedakan dari 24 citra data uji model.

Setiap citra diproses sebanyak 10 kali sehingga diperoleh 450 proses pengujian. Sebelum pengukuran utama, model dijalankan sebanyak tiga kali sebagai tahap pemanasan.

Parameter yang dicatat dalam pengujian non-fungsional meliputi:

- a. waktu pemuatan model;
- b. waktu inferensi YOLOv8;
- c. waktu K-Means Clustering;
- d. waktu total pemrosesan;
- e. waktu pemrosesan tercepat;
- f. waktu pemrosesan terlama;
- g. throughput;
- h. jumlah proses yang menghasilkan catatan waktu; dan
- i. pesan kesalahan yang muncul selama pelaksanaan benchmark.

Waktu total pemrosesan merupakan gabungan waktu inferensi YOLOv8 dan waktu K-Means Clustering. Pengukuran tidak mencakup waktu pengguna memilih berkas, waktu unggah citra, dan waktu rendering halaman Streamlit.

C. Tempat Dan Jadwal Penelitian

Penelitian ini dilaksanakan di Toko Batik Serbah Indah Kota Kediri sebagai lokasi utama untuk proses pengambilan data motif batik.

Tabel 3. 1 Jadwal Penelitian

No	Kegiatan	Bulan Ke-																							
		1	2	3	4	5	6																		
1.	Studi Pustaka																								
2.	Observasi																								
3.	Pengumpulan data																								
4.	Perancangan sistem																								
5.	Implementasi sistem																								
6.	Evaluasi sistem																								
7.	Penyusunan laporan																								

Penelitian dilakukan selama 6 bulan dengan beberapa tahapan yang dijabarkan dalam tabel 3.1 Jadwal penelitian.

D. Kebutuhan Sistem

1. Kebutuhan Fungsional

Kebutuhan *fungsional* menjelaskan kemampuan atau layanan yang harus dapat dilakukan oleh sistem deteksi motif dan warna dominan batik. Adapun kebutuhan *fungsional* sistem yang dikembangkan adalah sebagai

berikut:

- a. Sistem mampu menerima input berupa citra batik dengan format JPG, JPEG, atau PNG.
- b. Sistem mampu melakukan pra-pemrosesan citra, meliputi resize dan normalisasi sebelum proses deteksi.
- c. Sistem mampu mendeteksi motif batik pada citra menggunakan model YOLOv8.
- d. Sistem mampu menampilkan *bounding box (bbox)* pada area motif batik yang terdeteksi.
- e. Sistem mampu menampilkan label motif batik hasil deteksi pada citra.
- f. Sistem mampu mengekstraksi warna dominan dari area motif batik yang telah terdeteksi menggunakan metode *K-Means Clustering*.
- g. Sistem mampu menampilkan hasil warna dominan dalam bentuk nilai RGB atau kode warna (*Hex*).
- h. Sistem mampu menyimpan dan menampilkan hasil analisis deteksi motif dan warna dominan kepada pengguna.

2. Kebutuhan Non-Fungsional

Kebutuhan *non-fungsional* menjelaskan batasan dan karakteristik kualitas sistem yang dikembangkan agar dapat berjalan secara optimal. Adapun kebutuhan *non-fungsional* sistem adalah sebagai berikut:

- a. Kinerja (*Performance*)
Sistem mampu melakukan proses deteksi motif dan ekstraksi warna dominan dalam waktu yang *relatif* cepat, dengan waktu *inferensi* yang sesuai untuk penggunaan *eksperimental*.
- b. Akurasi (*Accuracy*)
Sistem diharapkan memiliki tingkat akurasi deteksi motif yang tinggi, yang diukur menggunakan *metrik precision, recall, dan mAP*.
- c. Kemudahan Penggunaan (*Usability*)
Sistem dirancang dengan antarmuka sederhana sehingga pengguna dapat dengan mudah mengunggah citra dan melihat hasil analisis tanpa keahlian teknis khusus.

d. Kompatibilitas (*Compatibility*)

Sistem dapat dijalankan pada perangkat komputer/laptop dengan sistem operasi umum serta mendukung penggunaan Python dan *framework* YOLOv8.

e. Keandalan (*Reliability*)

Sistem mampu berjalan secara stabil tanpa *error* saat memproses citra batik dengan variasi motif dan warna yang berbeda.

Dengan adanya perincian kebutuhan *fungsi*ional dan *non-fungsi*ional ini, pengembangan sistem deteksi motif dan warna dominan batik diharapkan dapat dilakukan secara lebih terstruktur serta mendukung proses pengembangan, *implementasi*, dan pengujian eksperimental sesuai dengan tujuan penelitian.

E. Objek / subjek penelitian

1. Objek Penelitian

Objek dalam penelitian ini adalah citra digital motif batik yang digunakan untuk melatih, memvalidasi, dan menguji sistem deteksi motif serta ekstraksi warna dominan. Motif batik yang digunakan dibatasi menjadi lima kelas, yaitu:

- a. Batik Bali;
- b. Batik Betawi;
- c. Batik Garutan;
- d. Batik Parang; dan
- e. Batik Sekar.

Citra diperoleh melalui dokumentasi langsung di Toko Batik Serbah Indah Kota Kediri, Jl Pattimura No.123, Jagala, kec. Kota. Kota Kediri, Jawa Timur 64125 serta dari *sumber daring* yang *relevan*. Setiap citra diperlakukan sebagai satu unit data penelitian.

2. Populasi Data

Populasi terjangkau dalam penelitian ini adalah seluruh citra batik yang berhasil dikumpulkan dan memenuhi kriteria penelitian. Setelah proses

pemeriksaan dan seleksi, diperoleh *populasi* sebanyak 586 citra.

Secara *matematis*, jumlah populasi dituliskan sebagai:

$$N = 586 \text{ citra}$$

Populasi tersebut mencakup citra dari lima kelas motif batik yang digunakan dalam penelitian. Penelitian ini tidak menggunakan responden manusia sebagai sumber data utama karena unit analisisnya berupa citra digital. Oleh sebab itu, rumus penentuan sampel *responden*, seperti rumus *Slovin*, tidak digunakan.

3. Teknik Pengambilan Sampe

Teknik pengambilan sampel yang digunakan adalah total *sampling*. Seluruh citra yang telah memenuhi kriteria seleksi digunakan sebagai dataset penelitian. Dengan demikian, jumlah sampel sama dengan jumlah *populasi* terjangkau.

Secara matematis, penentuan sampel dituliskan sebagai:

$$n = N$$

$$n = 586 \text{ citra}$$

Keterangan:

N = jumlah *populasi* citra yang memenuhi kriteria;

n = jumlah citra yang digunakan sebagai sampel penelitian.

Penggunaan total *sampling* dipilih karena seluruh citra yang telah dikumpulkan diperlukan untuk meningkatkan jumlah dan variasi data pembelajaran model.

4. Kriteria Inklusi Data

Citra dimasukkan ke dalam dataset apabila memenuhi kriteria berikut:

- citra termasuk dalam salah satu dari lima kelas motif penelitian;
- citra dapat dibaca oleh sistem dan tidak mengalami kerusakan berkas;
- motif batik terlihat cukup jelas untuk dilakukan anotasi;
- citra menggunakan format JPG, JPEG, atau PNG;
- citra tidak mengalami gangguan visual berat, seperti buram berlebihan, pencahayaan terlalu gelap, atau objek tertutup;
- citra memiliki ukuran yang memungkinkan objek motif diberi *anotasi*

bounding box; dan

g. citra tidak merupakan duplikasi identik dari data lain.

5. Kriteria Eksklusi Data

Citra tidak digunakan dalam penelitian apabila:

- a. tidak termasuk dalam lima kelas motif penelitian;
- b. berkas tidak dapat dibuka atau mengalami kerusakan;
- c. motif tidak terlihat secara jelas;
- d. citra memiliki tingkat keburaman atau gangguan pencahayaan yang terlalu tinggi;
- e. sebagian besar motif tertutup oleh objek lain;
- f. anotasi kelas dan *bounding box* tidak dapat ditentukan secara konsisten; atau
- g. citra teridentifikasi sebagai duplikasi data.

6. Batas Ukuran dan Dimensi Citra

Citra yang digunakan memiliki ukuran asli yang beragam sesuai sumber pengambilan data. Dimensi asli terkecil dalam dataset adalah [isi hasil pemeriksaan dimensi minimum] *piksel*, sedangkan dimensi terbesar adalah [isi hasil pemeriksaan dimensi maksimum] *piksel*.

Sebelum digunakan dalam proses pelatihan dan inferensi, seluruh citra diseragamkan menjadi ukuran:

640×640 pikse

Penyeragaman ukuran dilakukan agar seluruh citra memiliki dimensi masukan yang sesuai dengan konfigurasi model. Perubahan ukuran mempertahankan objek motif dan menyesuaikan posisi anotasi *bounding box*.

7. Pembagian Dataset

Sebanyak 586 citra dibagi menjadi data latih, data validasi, dan data uji sebagai berikut:

- a. data latih sebanyak 513 citra;
- b. data validasi sebanyak 49 citra; dan
- c. data uji sebanyak 24 citra.

Persentase pembagian *dataset* dihitung menggunakan rumus:

Persentase data = jumlah data pada bagian ÷ jumlah seluruh data × 100%

Berdasarkan perhitungan tersebut, diperoleh:

Data latih = $513 \div 586 \times 100\% = 87,54\%$

Data validasi = $49 \div 586 \times 100\% = 8,36\%$

Data uji = $24 \div 586 \times 100\% = 4,10\%$

Data latih digunakan untuk memperbarui bobot model, data validasi digunakan untuk memantau performa selama proses pelatihan, sedangkan data uji digunakan untuk menilai kemampuan model terhadap citra yang tidak digunakan dalam pembaruan bobot.

8. Subjek Penelitian

Penelitian ini tidak melibatkan responden manusia. *Subjek* yang dianalisis adalah model *komputasi* dan sistem aplikasi yang terdiri atas:

- a. model deteksi objek untuk mengenali kelas dan lokasi motif batik;
- b. *K-Means Clustering* untuk mengelompokkan nilai piksel RGB pada area motif; dan
- c. aplikasi berbasis Streamlit untuk menerima citra dan menampilkan hasil analisis.

Model dievaluasi berdasarkan nilai *precision*, *recall*, *mAP50*, dan *mAP50–95*. Sistem aplikasi dievaluasi melalui pengujian fungsional dan *non-fungsional*.

F. Prosedur penelitian

Prosedur penelitian ini disusun untuk memastikan seluruh tahapan penelitian berlangsung secara sistematis dan sesuai dengan tujuan penelitian. Secara umum, prosedur penelitian terdiri dari beberapa tahap utama, yaitu pengumpulan data, pengolahan data, perancangan model, pelatihan model, pengujian, serta analisis hasil. Tahapan tersebut dijelaskan sebagai berikut:

1. Studi Pustaka

Tahap ini dilakukan dengan mempelajari berbagai referensi terkait batik, motif batik, pengolahan citra digital, serta algoritma *object detection*

khususnya YOLOv8. Studi *literatur* berfungsi sebagai landasan teori dalam menyusun metode dan kerangka penelitian.

2. Observasi

Observasi dilakukan secara langsung di Tokoh Serbah Indah, Kota Kediri untuk memahami proses pembuatan batik, karakteristik motif, pola pewarnaan, serta kondisi lapangan terkait pengambilan data. Observasi juga digunakan untuk menentukan kebutuhan data, teknik pengambilan gambar, dan kategori motif yang akan dijadikan *dataset*.

3. Pengumpulan Data

Data yang dikumpulkan berupa citra motif batik dari Tokoh Serbah Indah. Tahapan pengumpulan data meliputi:

- a. Pengambilan gambarmotif batik menggunakan kamera atau *smartphone*.
- b. Mencatat informasi pendukung seperti nama motif, pola dasar, dan variasinya.
- c. Menyimpan gambar dalam format yang sesuai (JPG/PNG).
- d. Melakukan seleksi gambar agar *dataset* yang digunakan memiliki kualitas yang baik.

4. Perancangan Sistem

Pada tahap ini dilakukan perancangan sistem deteksi motif batik berbasis YOLOv8, meliputi:

- a. Perancangan alur proses sistem (*flowchart* atau *diagram*).
- b. Perancangan struktur *dataset* (*train*, *validation*, dan *test*).
- c. Menentukan parameter model YOLOv8 yang akan digunakan.
- d. Menyusun konfigurasi *dataset* dan label.

5. Implementasi Sistem

Tahap ini mencakup proses pembangunan model dan sistem berdasarkan perancangan sebelumnya, meliputi:

- a. Melakukan labeling objek menggunakan aplikasi seperti *LabelImg* atau *Roboflow*.
- b. Melakukan *training* model YOLOv8 sesuai parameter yang ditentukan.

c. Mengintegrasikan model hasil *training* ke aplikasi atau sistem yang digunakan.

d. Melakukan pengujian awal untuk melihat apakah sistem berjalan dengan baik.

6. Evaluasi Sistem

Evaluasi dilakukan untuk menguji performa model dalam mendeteksi motif batik. Evaluasi dilakukan menggunakan metrik seperti:

a. *Mean Average Precision* (mAP)

b. *Precision* dan *recall*

c. Tingkat keberhasilan deteksi pada gambar baru

d. Analisis kesalahan deteksi atau *miss detection*

Hasil evaluasi digunakan untuk menentukan apakah model telah memenuhi tujuan penelitian.

7. Penyusunan Laporan

Tahap terakhir adalah menyusun laporan penelitian secara sistematis berdasarkan seluruh rangkaian kegiatan, mulai dari pengumpulan data hingga evaluasi sistem. Hasil laporan ini kemudian disusun menjadi skripsi sesuai format akademik yang berlaku.

8. Flowchart Alur Penelitian

Diagram alir penelitian digunakan untuk menggambarkan urutan pelaksanaan penelitian secara menyeluruh, mulai dari identifikasi masalah hingga penyusunan kesimpulan. Diagram tersebut menghubungkan tahapan eksperimen data citra dengan proses pengembangan aplikasi.

Tahapan penelitian dimulai dengan identifikasi masalah dan studi pustaka. Selanjutnya, dilakukan pengumpulan citra, seleksi data, pemberian label kelas, serta anotasi *bounding box* menggunakan *Roboflow berbasis web*. *Dataset* yang telah dianotasi dibagi menjadi data latih, data validasi, dan data uji.

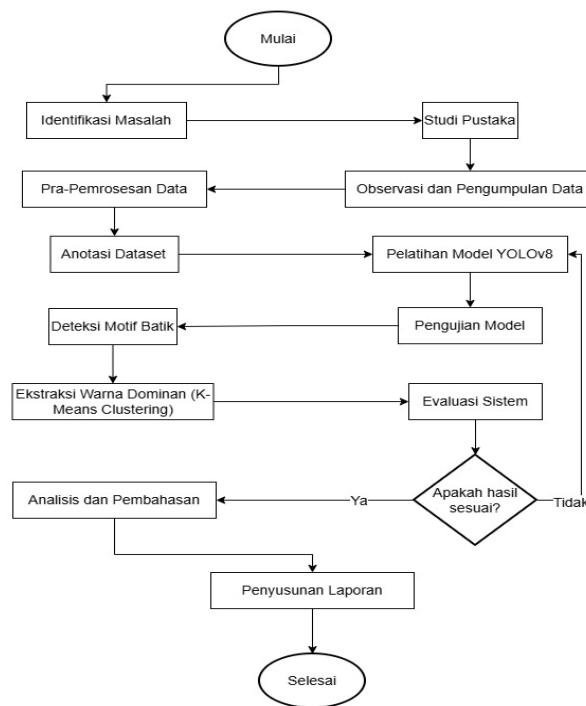
Data latih digunakan dalam proses pelatihan model, sedangkan data validasi digunakan untuk memantau performa model selama pelatihan.

Model terbaik kemudian diuji menggunakan data uji. Apabila performa

model belum memadai, proses dikembalikan ke tahap pemeriksaan data, anotasi, pra-pemrosesan, atau pelatihan ulang.

Model yang telah memenuhi hasil pengujian diintegrasikan ke dalam aplikasi Streamlit. Aplikasi menerima citra dari pengguna, menjalankan deteksi motif, mengambil *Region of Interest*, mengekstraksi warna dominan menggunakan *K-Means Clustering*, dan menampilkan hasil analisis. Setelah seluruh fungsi aplikasi berhasil diuji, hasil penelitian dianalisis dan digunakan sebagai dasar penyusunan kesimpulan.

Keseluruhan tahapan penelitian tersebut disajikan pada Gambar 3.3.



Gambar 3.3 tahapan penelitian

Gambar 3.3 memperlihatkan bahwa eksperimen model dan pengembangan aplikasi merupakan dua proses yang saling berhubungan. Hasil eksperimen berupa model terbaik digunakan sebagai komponen utama dalam aplikasi, sedangkan aplikasi digunakan untuk menjalankan *inferensi* dan menampilkan hasil deteksi motif serta warna dominan.

G. Teknik Analisis Data

Teknik analisis data pada penelitian ini dilakukan untuk mengolah, menguji, dan mengevaluasi data citra motif batik sehingga dapat diperoleh model deteksi yang akurat dan sesuai dengan tujuan penelitian.

1. Desain sistem

Perancangan sistem menggunakan pendekatan *Object-Oriented Analysis and Design* (OOAD). Pemodelan dilakukan menggunakan *Unified Modeling Language* (UML) yang terdiri atas use case diagram, activity diagram, sequence diagram, dan class diagram.

Seluruh diagram dirancang secara sinkron. Setiap aktivitas pada use case diagram dijelaskan melalui activity diagram, urutan interaksi antarkomponen ditunjukkan melalui sequence diagram, sedangkan struktur kelas yang menjalankan proses tersebut digambarkan melalui class diagram.

Sistem tidak menggunakan DBMS. Penyimpanan dilakukan berbasis berkas lokal berupa citra hasil deteksi dan file CSV. Karena itu, pemodelan UML menggunakan kelas *File StorageHandler* untuk menangani proses pembacaan dan penulisan berkas.

a. Arsitektur Sistem

1) Use Case Diagram

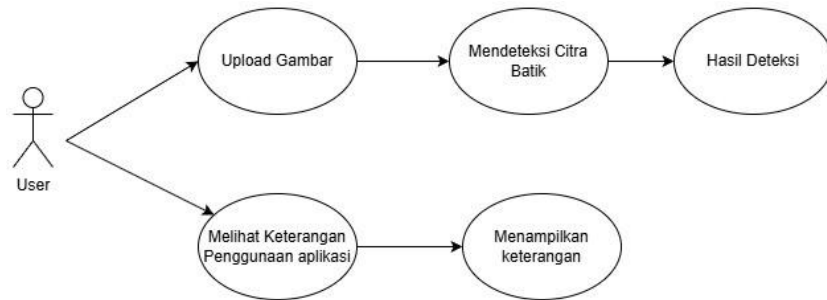
Aktor utama pada sistem adalah pengguna. Pengguna berinteraksi dengan aplikasi melalui beberapa fungsi berikut:

- mengunggah citra;
- menjalankan analisis citra;
- melihat hasil deteksi motif;
- melihat hasil warna dominan; dan
- menyimpan hasil analisis.

Use case “Menjalankan Analisis Citra” mencakup beberapa proses internal, yaitu validasi citra, pra-pemrosesan, deteksi motif, pengambilan *Region of Interest*, dan ekstraksi warna dominan.

Hubungan antara pengguna dan fungsi sistem ditunjukkan pada

Gambar 3.4.

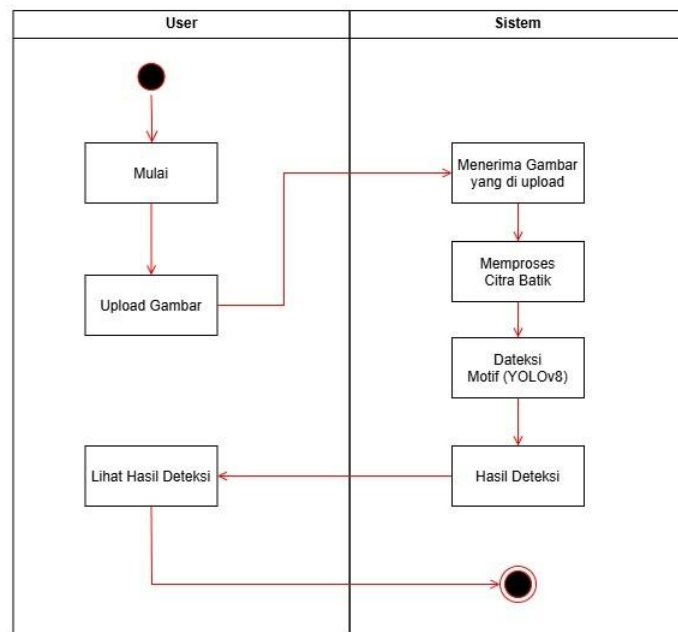


Gambar 3. 4 Use Case Diagram

Gambar 3.4 menunjukkan bahwa pengguna memberikan masukan berupa citra batik. Sistem selanjutnya menjalankan proses analisis secara otomatis dan menampilkan hasil berupa label motif, nilai *confidence*, *bounding box*, nilai RGB, serta kode *heksadesimal*.

2) Activiy Diagram

Diagram ini menunjukkan bahwa *user* hanya melakukan dua aktivitas utama, yaitu memilih/unggah gambar dan melihat hasil deteksi, sedangkan seluruh proses pemrosesan citra dan deteksi motif dilakukan otomatis oleh sistem.



Gambar 3. 5 Activity Diagram

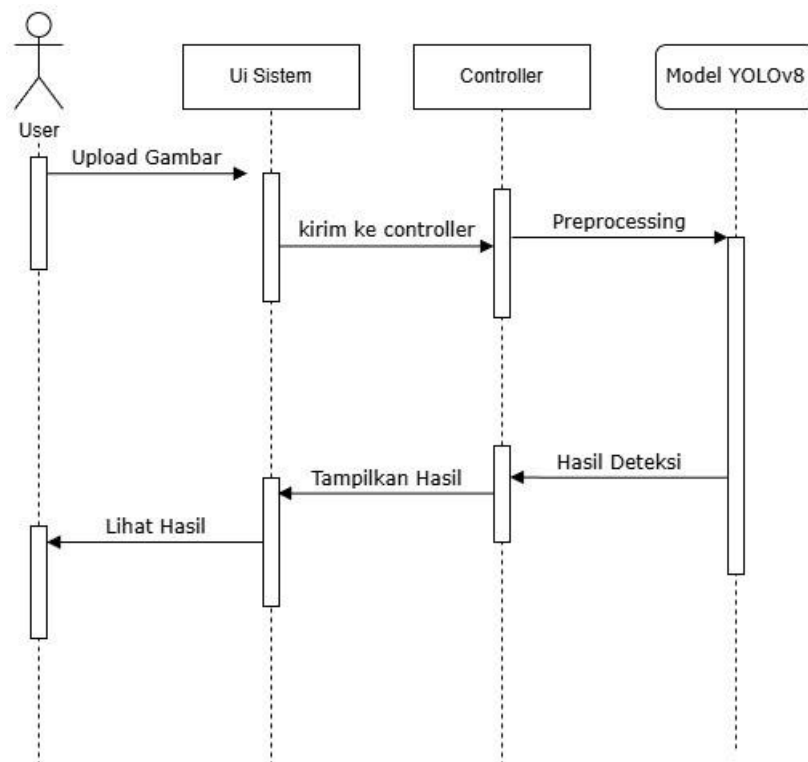
3) Sequence Diagram

Sequence diagram digunakan untuk menggambarkan urutan *interaksi* antara pengguna, antarmuka sistem, *controller*, dan model YOLOv8 dalam proses analisis citra batik. Diagram ini menunjukkan bagaimana data citra dikirim, diproses, hingga hasil deteksi ditampilkan kembali kepada pengguna.

Proses dimulai ketika User mengunggah gambar melalui UI Sistem. Setelah citra diterima, UI Sistem mengirimkan data gambar ke *Controller* untuk diproses lebih lanjut. Selanjutnya, *Controller* meneruskan citra ke Model YOLOv8 untuk menjalankan tahap preprocessing dan deteksi objek.

Setelah proses deteksi selesai, Model YOLOv8 mengembalikan hasil deteksi kepada *Controller*. Hasil tersebut kemudian diteruskan ke UI Sistem untuk ditampilkan. Setelah hasil ditampilkan, User dapat melihat hasil analisis yang berisi informasi motif batik yang berhasil dikenali.

Urutan interaksi tersebut ditunjukkan pada Gambar 3.8.



Gambar 3. 6 *Sequence Diagram*

Alur interaksi pada sequence diagram adalah sebagai berikut:

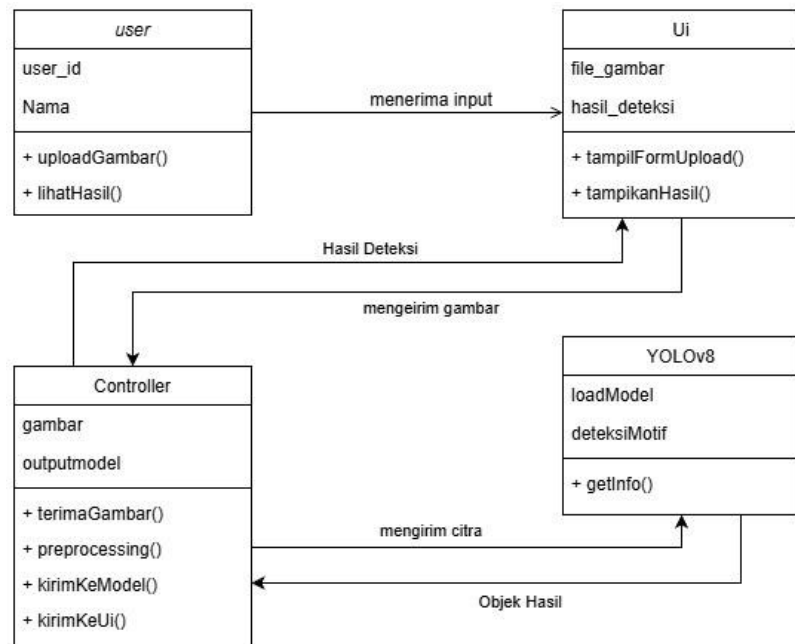
- a) *User* mengunggah gambar ke UI Sistem.
- b) UI Sistem mengirim gambar ke *Controller*.
- c) *Controller* meneruskan proses ke Model YOLOv8 untuk melakukan preprocessing.
- d) Model YOLOv8 melakukan deteksi dan menghasilkan keluaran berupa hasil deteksi.
- e) Hasil deteksi dikirim kembali ke *Controller*.
- f) *Controller* meneruskan hasil ke UI Sistem.
- g) UI Sistem menampilkan hasil kepada *User*.
- h) *User* melihat hasil deteksi yang ditampilkan oleh sistem.

Dengan demikian, sequence diagram ini menunjukkan bahwa proses deteksi motif batik berlangsung secara berurutan dari masukan pengguna, pemrosesan oleh sistem, sampai penyajian hasil akhir melalui antarmuka aplikasi.

4) *Class Diagram*

Class diagram digunakan untuk menggambarkan struktur kelas yang terdapat pada sistem deteksi motif batik, beserta *atribut*, *operasi*, dan hubungan antar kelas. Diagram ini menunjukkan komponen utama sistem yang berperan dalam proses unggah gambar, pengolahan citra, deteksi motif, dan penampilan hasil deteksi.

Pada sistem ini terdapat empat kelas utama, yaitu *User*, *Ui*, *Controller*, dan YOLOv8. Hubungan antar kelas tersebut ditunjukkan pada Gambar 3.9.



Gambar 3. 7 Class Diagram

Kelas *User* merepresentasikan pengguna yang berinteraksi langsung dengan sistem. Kelas ini memiliki atribut `user_id` dan nama, serta dua operasi utama, yaitu `uploadGambar()` untuk mengunggah citra batik ke sistem dan `lihatHasil()` untuk melihat hasil deteksi yang ditampilkan.

Kelas *Ui* merepresentasikan antarmuka sistem yang berfungsi menerima masukan dari pengguna dan menampilkan hasil proses. Kelas ini memiliki atribut `file_gambar` dan `hasil_deteksi`. Operasi yang dimiliki kelas ini adalah `tampil Form Upload()` untuk menampilkan formulir unggah gambar dan `tampilkan Hasil()` untuk menampilkan hasil deteksi motif batik kepada pengguna.

Kelas *Controller* berfungsi sebagai pengatur alur proses antara antar muka sistem dan model deteksi. Kelas ini memiliki atribut `gambar` dan `outputmodel`. Operasi yang dimiliki meliputi `terima Gambar()` untuk menerima data gambar dari antar muka, `preprocessing()` untuk melakukan persiapan data citra sebelum

diproses, kirim `KeModel()` untuk mengirim citra ke model YOLOv8, dan `kirimKeUi()` untuk mengirim hasil deteksi kembali ke antarmuka sistem.

Kelas YOLOv8 merepresentasikan model deteksi objek yang digunakan untuk mengenali motif batik pada citra. Kelas ini memiliki operasi `load Model`, `deteksi Motif`, dan `getInfo()`. Operasi `load Model` digunakan untuk memuat model yang telah dilatih, `deteksi Motif` digunakan untuk melakukan proses deteksi motif pada citra masukan, sedangkan `getInfo()` digunakan untuk mengambil informasi hasil deteksi.

Hubungan antar kelas pada diagram menunjukkan bahwa *User* memberikan masukan kepada *Ui* melalui proses unggah gambar. Selanjutnya, *Ui* mengirim gambar ke *Controller* untuk diproses. *Controller* meneruskan citra ke YOLOv8 untuk dilakukan deteksi motif. Setelah proses deteksi selesai, hasil dikembalikan ke *Controller*, kemudian diteruskan ke *Ui* agar dapat ditampilkan kepada *User*.

Dengan demikian, class diagram ini menunjukkan bahwa sistem bekerja melalui interaksi empat kelas utama yang saling terhubung, yaitu pengguna sebagai pemberi masukan, antarmuka sebagai penghubung interaksi, controller sebagai pengendali proses, dan YOLOv8 sebagai model inti untuk mendeteksi motif batik.

b. Mekanisme Penyimpanan Data

Penelitian ini tidak menggunakan database (seperti MySQL atau PostgreSQL), melainkan menerapkan mekanisme penyimpanan berbasis file (*file-based storage*). Pemilihan mekanisme ini disesuaikan dengan ruang lingkup penelitian yang bersifat eksperimental serta fokus pada evaluasi kinerja metode YOLOv8 dan *K-Means Clustering*, bukan pada pengelolaan data berskala besar.

1) Mekanisme Penyimpanan Data Berbasis File

Mekanisme penyimpanan data dalam sistem deteksi motif dan warna dominan batik dilakukan dengan cara sebagai berikut:

a) Penyimpanan Citra Input

Citra batik yang diunggah oleh pengguna disimpan dalam *folder* khusus, misalnya *dataset/input/*, dengan format file JPG atau PNG. Setiap file diberi nama unik berdasarkan waktu unggah untuk menghindari duplikasi data.

b) Penyimpanan Hasil Deteksi

Hasil deteksi motif batik berupa citra yang telah dilengkapi *bounding box* (*bbox*) dan label motif disimpan dalam *folder output/detection/*. Citra ini berfungsi sebagai dokumentasi visual hasil kerja model YOLOv8.

c) Penyimpanan Informasi Warna Dominan

Data warna dominan hasil *ekstraksi K-Means Clustering* disimpan dalam bentuk *file teks* atau CSV pada *folder output/color/*. Informasi yang disimpan meliputi:

- Nama file citra
- Nilai warna dominan (RGB dan/atau *Hex*)
- Waktu proses analisis

d) Penyimpanan Riwayat Deteksi (*History*)

Riwayat proses deteksi disimpan dalam file log (misalnya *history_detection.csv*) yang berisi:

- Nama citra
- Motif batik terdeteksi
- Nilai confidence
- Warna dominan
- Tanggal dan waktu pemrosesan

File ini berfungsi sebagai pengganti database untuk merekam hasil analisis tanpa memerlukan sistem manajemen basis data.

2) Alasan Tidak Menggunakan *Database*

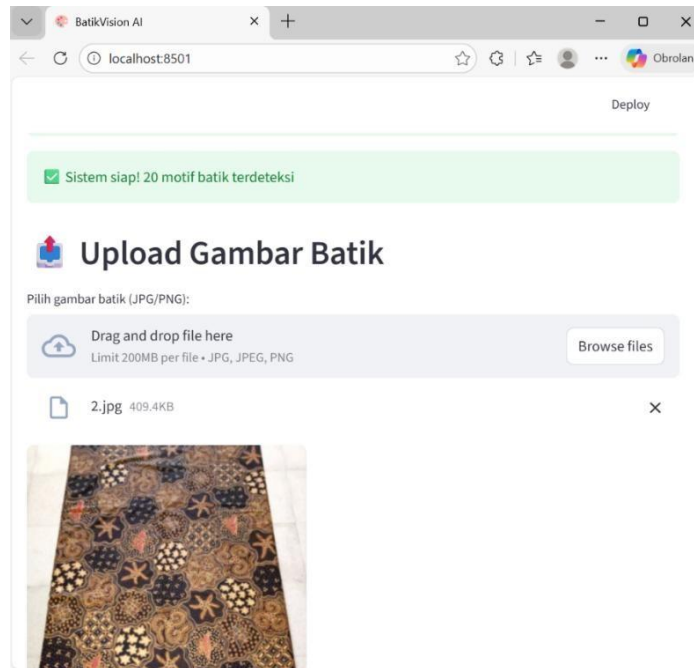
Tidak digunakannya database dalam penelitian ini didasarkan pada beberapa pertimbangan, yaitu:

- a) Fokus penelitian adalah pengujian dan evaluasi metode deteksi objek dan ekstraksi warna, bukan pengelolaan data jangka panjang.
- b) Jumlah data yang digunakan masih berada dalam skala kecil hingga menengah, sehingga penyimpanan berbasis *file* dinilai lebih *efisien* dan sederhana.
- c) Penyimpanan berbasis *file* sudah memadai untuk kebutuhan dokumentasi hasil eksperimen dan pelaporan penelitian.

Namun demikian, sistem yang dikembangkan tetap memungkinkan untuk dikembangkan lebih lanjut dengan *integrasi database* apabila diterapkan pada sistem produksi atau aplikasi skala besar.

c. Desain Antar Muka Pengguna

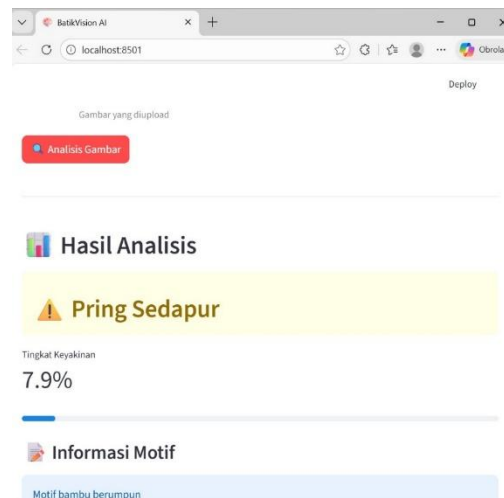
1) Halaman Utama



Gambar 3. 8 Halaman Utama

Halaman utama di sajikan pada gambar 3.7. pengguna pertama kali masuk ke halaman dan melakukan upload gambar melalu *browse* file setelah itu pilih gambar yang akan diupload.

2) Halaman hasil Analisis



Gambar 3. 9 halaman hasil analisis

Setelah gambar diupload pengguna klik tombol analisis gambar yang dapat di lihat pada gambar 3.8. setelah itu hasil analisis akan muncul dengan informasi motif yang di deteksi.

d. Metode Perhitungan dan Evaluasi

Metode perhitungan digunakan untuk menganalisis performa model YOLOv8, hasil ekstraksi warna dominan, pengujian fungsional, serta pengujian non-fungsional aplikasi. Pada tahap metode, persamaan disajikan tanpa menggunakan data simulasi. Substitusi angka dan interpretasi hasil pengujian aktual disajikan pada BAB IV.

1) Precision

Precision digunakan untuk mengukur proporsi prediksi positif yang sesuai dengan objek sebenarnya. Rumus precision dituliskan sebagai berikut:

$$Precision = \frac{TP}{TP + FP} \quad (3.1)$$

Keterangan:

- a) (TP) (True Positive) adalah objek yang berhasil dideteksi dengan benar;
- b) (FP) (False Positive) adalah prediksi objek yang tidak sesuai dengan objek sebenarnya.

Nilai precision yang semakin tinggi menunjukkan bahwa model menghasilkan lebih sedikit false positive.

2) Recall

Recall digunakan untuk mengukur kemampuan model dalam menemukan seluruh objek yang sebenarnya terdapat pada citra. Rumus recall dituliskan sebagai berikut:

$$Recall = \frac{TP}{TP + FN} \quad (3.2)$$

Keterangan:

- a) (TP) (True Positive) adalah objek yang berhasil dideteksi dengan benar;

b) (FN) (False Negative) adalah objek yang sebenarnya ada, tetapi tidak berhasil dideteksi model.

Nilai recall yang semakin tinggi menunjukkan bahwa model mampu menemukan lebih banyak objek yang tersedia.

3) Intersection over Union

Intersection over Union digunakan untuk mengukur tingkat kesesuaian antara bounding box hasil prediksi dan bounding box ground truth. Rumus IoU dituliskan sebagai berikut:

$$IoU = \frac{\text{Luas Area Irisan}}{\text{Luas Area Gabungan}} \quad (3.3)$$

Nilai IoU berada pada rentang 0 sampai 1. Nilai yang semakin mendekati 1 menunjukkan bahwa posisi bounding box prediksi semakin sesuai dengan anotasi sebenarnya.

4) Average Precision

Average Precision merupakan luas area di bawah kurva precision-recall untuk satu kelas. Rumus AP dituliskan sebagai berikut:

$$AP = \int_0^1 P(R), dR \quad (3.4)$$

Keterangan:

- a) (P(R)) adalah nilai precision pada tingkat recall tertentu;
- b) (AP) adalah Average Precision untuk satu kelas.

5) Mean Average Precision

Mean Average Precision diperoleh dari rata-rata nilai AP seluruh kelas. Rumus mAP dituliskan sebagai berikut:

$$mAP = \frac{1}{C} \sum_{i=1}^C AP_i \quad (3.5)$$

Keterangan:

- a) (AP_i) adalah Average Precision kelas ke-(i);
- b) (C) adalah jumlah kelas.

Dalam penelitian ini digunakan mAP50 dan mAP50–95.

23

mAP50 dihitung pada ambang IoU sebesar 0,50, sedangkan mAP50–95 dihitung berdasarkan rata-rata AP pada ambang IoU 0,50 sampai 0,95.

6) Perhitungan Warna Dominan

K-Means Clustering mengelompokkan piksel pada Region of Interest berdasarkan kedekatan nilai RGB. Jarak antara piksel dan pusat cluster dihitung menggunakan jarak Euclidean:

$$d(x_i, \mu_j) = \sqrt{(R_i - \mu_{Rj})^2 + (G_i - \mu_{Gj})^2 + (B_i - \mu_{Bj})^2} \quad (3.6)$$

Keterangan:

- a) (x_i) adalah piksel ke-(i);
- b) (μ_j) adalah pusat cluster ke-(j);
- c) (R_i) , (G_i) , dan (B_i) adalah nilai RGB piksel;
- d) (μ_{Rj}) , (μ_{Gj}) , dan (μ_{Bj}) adalah nilai RGB pusat cluster.

Setelah proses pengelompokan selesai, sistem menghitung jumlah anggota setiap cluster. Cluster dengan jumlah piksel terbanyak dipilih sebagai warna dominan. Nilai pusat cluster tersebut ditampilkan dalam bentuk RGB dan kode heksadesimal.

7) Pengujian Fungsional

Persentase keberhasilan pengujian fungsional dihitung dengan membandingkan jumlah skenario yang berhasil dengan jumlah seluruh skenario pengujian.

$$\begin{aligned} \text{Keberhasilan Fungsional} & \quad (3.7) \\ &= \frac{\text{Jumlah Skenario Berhasil}}{\text{Jumlah Seluruh Skenario}} \times 100 \end{aligned}$$

Status berhasil diberikan apabila hasil aktual sesuai dengan hasil yang diharapkan.

112

8) Rata-Rata Waktu Respons

Rata-rata waktu respons dihitung dari jumlah seluruh waktu pemrosesan dibagi dengan jumlah proses pengujian.

140

62

$$\bar{t} = \frac{\sum_{i=1}^n t_i}{n} \quad (3.8)$$

Keterangan:

- a) $(\bar{t})$ adalah rata-rata waktu pemrosesan;
- b) $(\bar{t})$ adalah rata-rata waktu pemrosesan;
- c) (t_i) adalah waktu pemrosesan pada pengujian ke-(i);
- d) (n) adalah jumlah proses pengujian.

Waktu total pemrosesan terdiri atas waktu inferensi YOLOv8 dan waktu K-Means Clustering.

$$t_{total} = t_{YOLOv8} + t_{K-Means} \quad (3.9)$$

9) Throughput

Throughput digunakan untuk mengetahui jumlah citra yang dapat diproses sistem dalam satu detik.

$$Throughput = \frac{Jumlah\ Citra\ yang\ Diproses}{Jumlah\ Seluruh\ Waktu\ Pemrosesan} \quad (3.10)$$

Nilai throughput yang semakin tinggi menunjukkan bahwa sistem mampu memproses lebih banyak citra dalam waktu yang sama.

10) Pengujian Confidence Threshold

Pengujian confidence threshold dilakukan untuk mengetahui pengaruh nilai ambang keyakinan terhadap jumlah prediksi benar dan prediksi salah yang ditampilkan pada aplikasi. Pengujian menggunakan 24 citra data uji yang sama pada nilai ambang 0,05, 0,25, 0,40, dan 0,50. Sebuah kandidat deteksi dipertahankan apabila nilai confidence memenuhi kondisi:

$$c_i \geq \tau$$

Keterangan:

- c_i = nilai confidence kandidat deteksi ke-i;
- τ = nilai confidence threshold.

60

Setiap hasil deteksi dibandingkan dengan anotasi ground truth menggunakan ambang Intersection over Union sebesar 0,50. Hasil pengujian dicatat dalam bentuk jumlah true positive, false positive, dan false negative. Nilai precision dan recall pada setiap ambang dihitung menggunakan persamaan yang telah dijelaskan sebelumnya.

Pada deteksi objek, proporsi prediksi positif yang salah dihitung menggunakan false discovery rate sebagai berikut:

$$FDR(\tau) = FP(\tau) / [TP(\tau) + FP(\tau)]$$

Nilai tersebut juga dapat dituliskan sebagai:

$$FDR(\tau) = 1 - Precision(\tau)$$

Penggunaan false discovery rate dipilih karena deteksi objek tidak mempunyai jumlah true negative yang dapat ditentukan secara langsung seperti pada klasifikasi biasa. Perubahan jumlah false positive antara ambang 0,05 dan ambang pembanding dihitung sebagai berikut:

$$\Delta FP = FP_{0,05} - FP_{\tau}$$

Apabila nilai false positive pada ambang pembanding lebih besar dari nol, persentase peningkatannya dihitung dengan persamaan:

$$Peningkatan FP = [(FP_{0,05} - FP_{\tau}) / FP_{\tau}] \times 100$$

Pengujian ini digunakan untuk menentukan keseimbangan antara kemampuan aplikasi menemukan objek dan risiko menampilkan prediksi yang tidak sesuai.

BAB IV

HASIL DAN PEMBAHASAN

A. Hasil Sistem

Hasil penelitian ini berupa aplikasi deteksi motif dan warna dominan batik yang dijalankan secara lokal menggunakan Streamlit. Aplikasi dikembangkan menggunakan bahasa pemrograman Python melalui Visual Studio Code. Sistem menerima masukan berupa citra dengan format JPG, JPEG, atau PNG.

Citra yang diunggah diproses menggunakan model YOLOv8 untuk menghasilkan label kelas, nilai confidence, dan koordinat bounding box. Koordinat tersebut digunakan untuk mengambil Region of Interest (ROI) dari citra asli. Pixel pada area ROI kemudian diproses menggunakan K-Means Clustering untuk memperoleh warna dominan dalam bentuk nilai RGB, kode heksadesimal, dan visualisasi warna.

Model terbaik hasil pelatihan disimpan dalam berkas best.pt dan digunakan sebagai model utama pada aplikasi. Sistem dikembangkan untuk mendeteksi lima kelas motif batik, yaitu Batik Bali, Batik Betawi, Batik Garutan, Batik Parang, dan Batik Sekar. Contoh citra dari setiap kelas disajikan pada gambar 4.1.



Gambar 4.1 Citra Batik

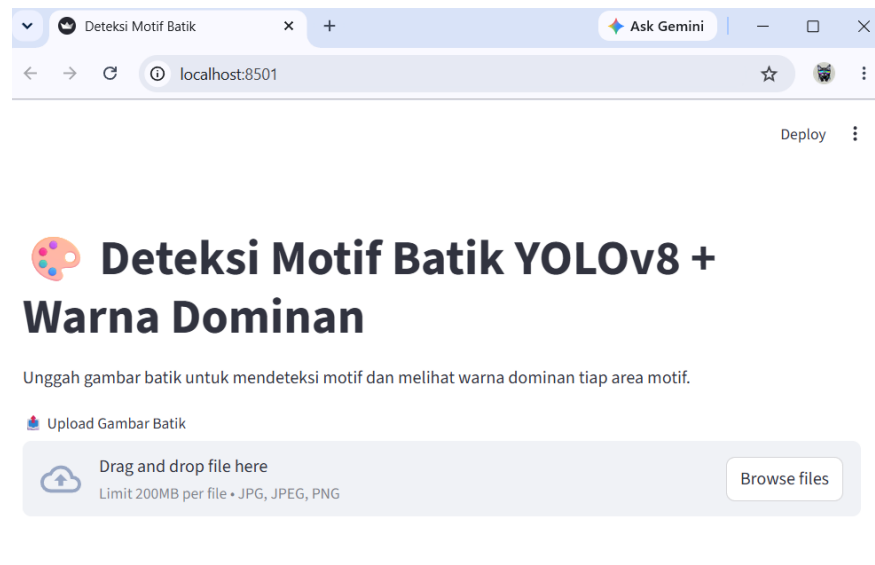
Berdasarkan Gambar 4.1, setiap kelas mempunyai karakteristik visual yang berbeda dalam bentuk susunan ornamen, arah pola, tekstur, dan pengulangan motif. Perbedaan tersebut digunakan oleh YOLOv8 sebagai dasar dalam mempelajari dan membedakan kelas motif batik.

Citra yang ditampilkan pada tabel merupakan contoh data dari setiap kelas dan tidak mewakili seluruh variasi citra yang terdapat dalam dataset.

1. Antarmuka Sistem

a. Halaman Utama

Pengguna mengunggah gambar batik melalui *fitur file uploader*.

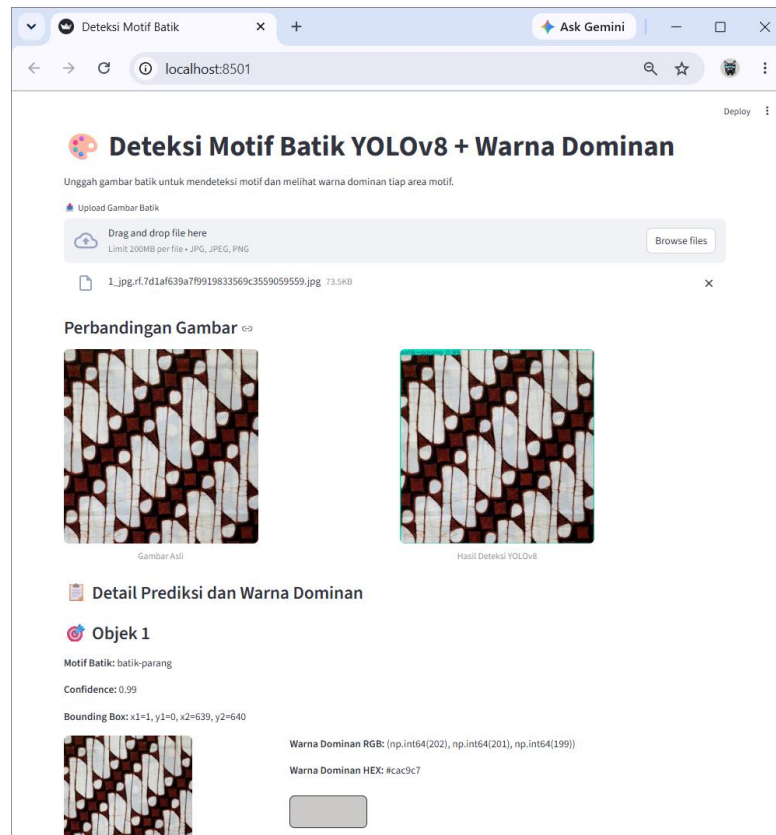


Gambar 4. 2 Tampilan Halaman Utama

Gambar 4.1 menunjukkan fitur unggah citra pada aplikasi berbasis *Streamlit*. Pengguna dapat memilih citra *berformat* JPG, JPEG, atau PNG. Citra yang dipilih ditampilkan pada halaman sebelum proses analisis dijalankan.

b. Halaman Hasil analisis

Setelah citra berhasil diunggah, aplikasi menjalankan proses deteksi motif dan ekstraksi warna dominan secara otomatis. Aplikasi kemudian menampilkan citra asli dan citra yang telah dilengkapi bounding box.



Gambar 4. 3 Hasil Deteksi Batik

Gambar 4.3 menunjukkan hasil analisis yang ditampilkan oleh aplikasi. Informasi yang dihasilkan meliputi:

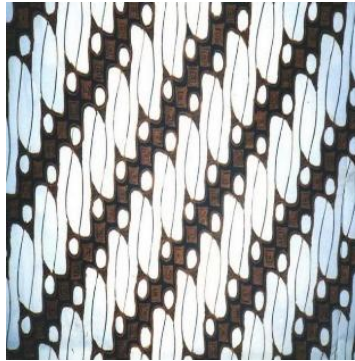
- 1) citra asli yang diunggah pengguna;
- 2) citra hasil deteksi dengan bounding box;
- 3) nama kelas motif;
- 4) nilai confidence;
- 5) koordinat bounding box;
- 6) hasil pemotongan ROI;
- 7) nilai RGB warna dominan;
- 8) kode heksadesimal; dan
- 9) kotak visual warna dominan.

Tampilan tersebut memungkinkan pengguna melihat perbandingan antara citra masukan dan hasil pemrosesan pada satu halaman aplikasi.

2. Hasil Implementasi Deteksi

a. Citra Masukan

Salah satu citra yang digunakan untuk menguji fungsi aplikasi adalah citra Batik Parang. Citra masukan tersebut ditunjukkan pada Gambar 4.4.

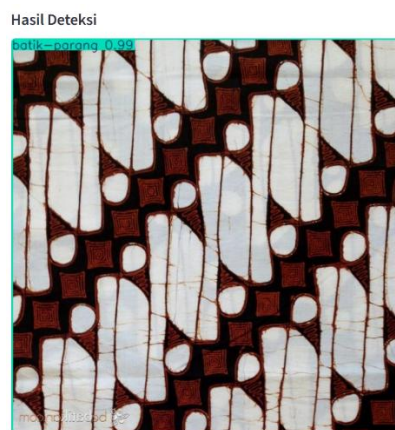


Gambar 4. 4 Citra Masukan Batik Parang

Gambar 4.4 merupakan citra Batik Parang yang diunggah ke dalam aplikasi. Citra tersebut selanjutnya diproses oleh model YOLOv8n untuk menentukan kelas dan lokasi motif.

b. Hasil deteksi YOLOv8

Hasil deteksi terhadap citra Batik Parang ditunjukkan pada Gambar 4.5.



Gambar 4. 5 Hasil Deteksi Batik Parang

Berdasarkan Gambar 4.5, model berhasil mengenali Batik Parang dan menampilkan bounding box pada area motif. Hasil deteksi juga dilengkapi dengan label kelas dan nilai confidence.

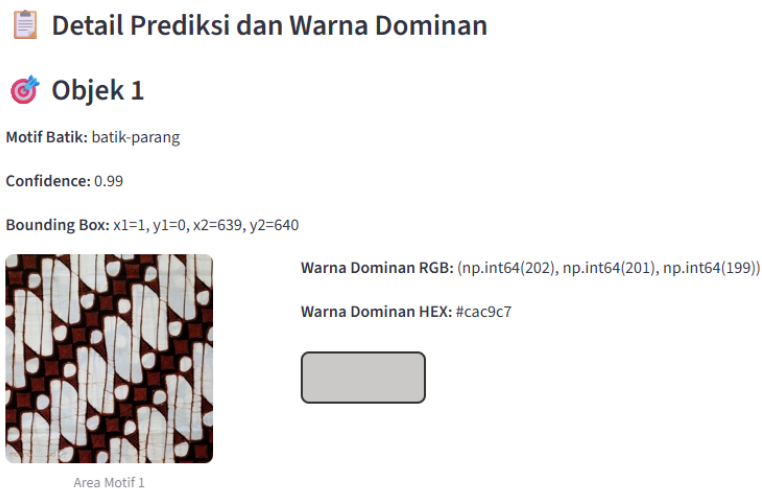
148

Nilai confidence sebesar 0,99 menunjukkan tingkat keyakinan model terhadap prediksi pada citra tersebut, tetapi tidak menunjukkan akurasi keseluruhan model.

96

c. Hasil Ekstraksi Warna Dominan

Detail hasil prediksi dan warna dominan ditunjukkan pada Gambar 4.6.



Gambar 4. 6 Detail Prediksi dan Warna Dominan

Gambar 4.6 menunjukkan ROI hasil pemotongan berdasarkan koordinat bounding box. Pixel pada ROI dikelompokkan menjadi tiga cluster menggunakan K-Means Clustering. Cluster dengan jumlah pixel terbanyak dipilih sebagai warna dominan. Aplikasi kemudian menampilkan nilai RGB, kode heksadesimal, dan kotak visual warna.

Nilai RGB dan kode heksadesimal yang digunakan dalam pembahasan harus mengikuti hasil yang benar-benar ditampilkan pada aplikasi. Nilai contoh atau nilai simulasi tidak digunakan sebagai hasil penelitian.

76

3. Hasil Pelatihan dan Validasi Model YOLOv8

100

Pelatihan model dilakukan menggunakan YOLOv8n dengan bobot awal pralatih. Konfigurasi pelatihan disajikan pada Tabel 4.1.

Tabel 4. 1 Konfigurasi Pelatihan Model YOLOv8

Parameter	Konfigurasi
Varian model	YOLOv8n
Bobot awal	Pralatih YOLOv8n
Jumlah epoch	50
Batch size	16
Ukuran masukan	640 × 640 piksel
Optimizer	AdamW
Data latih	513 citra
Data validasi	49 citra
Jumlah objek validasi	57 objek
Jumlah kelas	5 kelas

Model terbaik berdasarkan hasil validasi disimpan dalam berkas best.pt.

Hasil validasi setiap kelas disajikan pada Tabel 4.2.

Tabel 4. 2 Hasil Validasi Model per Kelas

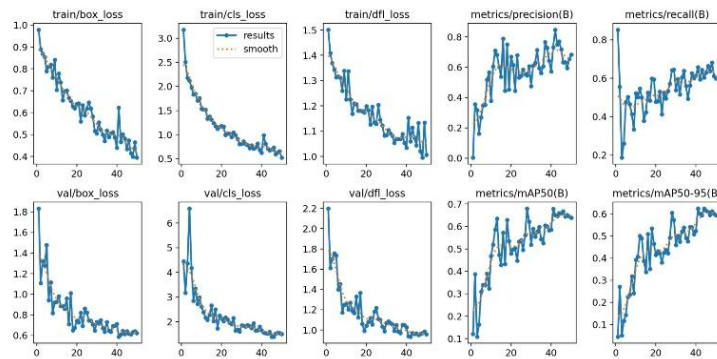
Kelas	Citra	Objek	Precision	Recall	mAP50	mAP50–95
Batik Parang	9	9	0,990	1,000	0,995	0,973
Batik Bali	14	15	0,956	0,667	0,846	0,792
Batik Betawi	9	9	0,596	0,778	0,671	0,670
Batik Garutan	8	15	0,536	0,200	0,199	0,157
Batik Sekar	9	9	0,769	0,377	0,588	0,528
Keseluruhan kelas	49	57	0,769	0,604	0,660	0,624

Berdasarkan Tabel 4.2, Batik Parang memperoleh performa tertinggi dengan mAP50–95 sebesar 0,973. Performa terendah diperoleh Batik Garutan dengan mAP50–95 sebesar 0,157. Perbedaan tersebut

menunjukkan bahwa kemampuan model belum merata pada seluruh kelas.

a. Grafik Hasil Pelatihan

Perubahan nilai loss dan metrik validasi selama 50 epoch ditunjukkan pada Gambar 4.7.



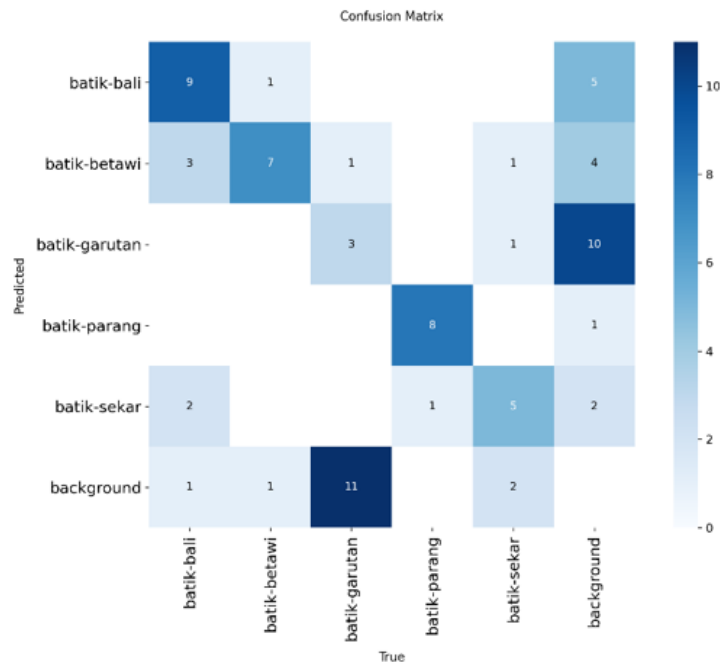
Gambar 4. 7 Grafik Hasil Pelatihan Model YOLOv8

Gambar 4.7 menunjukkan kecenderungan penurunan box loss, classification loss, dan distribution focal loss selama pelatihan. Penurunan tersebut menunjukkan bahwa model secara bertahap mempelajari kelas dan lokasi objek.

Nilai precision, recall, mAP50, dan mAP50–95 cenderung meningkat, meskipun masih mengalami fluktuasi pada beberapa epoch. Fluktuasi tersebut menunjukkan bahwa proses generalisasi belum sepenuhnya stabil. Kondisi ini dapat dipengaruhi oleh jumlah data validasi yang hanya terdiri atas 49 citra dengan 57 objek serta perbedaan tingkat kesulitan setiap kelas.

b. Confusion Matrix

Distribusi prediksi model pada setiap kelas ditunjukkan pada Gambar 4.8.



Gambar 4. 8 Confusion Matrix Hasil Validasi

Gambar 4.8 memperlihatkan kelas yang berhasil dikenali dan kelas yang masih sering diprediksi sebagai kelas lain atau sebagai latar belakang. Batik Parang menunjukkan hasil pengenalan yang lebih konsisten, sedangkan Batik Garutan masih banyak tidak ditemukan. Kondisi tersebut sejalan dengan nilai recall Batik Garutan yang hanya sebesar 0,200.

4. Evaluasi Model YOLOv8

Evaluasi model dilakukan menggunakan hasil validasi model terbaik yang tersimpan dalam berkas best.pt. Data validasi terdiri atas 49 citra dengan 57 objek dari lima kelas motif, yaitu Batik Bali, Batik Betawi, Batik Garutan, Batik Parang, dan Batik Sekar. Seluruh nilai evaluasi pada bagian ini berasal dari keluaran validasi aktual model YOLOv8n dan bukan dari data simulasi.

a. Metrik Evaluasi Model

Metrik yang digunakan dalam evaluasi meliputi precision, recall, mAP50, dan mAP50–95. Hasil evaluasi keseluruhan model disajikan pada Tabel 4.3.

Tabel 4. 3 Hasil Evaluasi Model YOLOv8

Metrik	Nilai	Keterangan
Precision	0,769	Ketepatan prediksi positif model
Recall	0,604	Kemampuan model menemukan objek sebenarnya
mAP50	0,660	Rata-rata AP pada ambang IoU 0,50
mAP50–95	0,624	Rata-rata AP pada ambang IoU 0,50 sampai 0,95

Nilai precision sebesar 0,769 menunjukkan bahwa sebagian besar prediksi positif yang dihasilkan model sesuai dengan objek motif sebenarnya. Namun, model masih menghasilkan prediksi yang tidak sesuai pada beberapa citra.

Nilai recall sebesar 0,604 menunjukkan bahwa model mampu menemukan sekitar 60,4% objek yang terdapat pada data validasi. Nilai recall yang lebih rendah daripada precision menunjukkan bahwa model lebih baik dalam menjaga ketepatan prediksi daripada menemukan seluruh objek motif.

Nilai mAP50 sebesar 0,660 menunjukkan kemampuan model dalam menentukan kelas dan lokasi objek pada ambang IoU 0,50. Nilai mAP50–95 sebesar 0,624 menunjukkan performa model ketika posisi bounding box dievaluasi menggunakan ambang IoU yang lebih ketat.

Selisih antara nilai mAP50 dan mAP50–95 dihitung sebagai berikut:

$$\text{Selisih } mAP = 0,660 - 0,624$$

$$\text{Selisih } mAP = 0,036$$

Persentase penurunannya adalah:

$$\begin{aligned} \text{Penurunan} &= \frac{0,660 - 0,624}{0,660} \times 100 \\ &= 5,45 \end{aligned}$$

Penurunan sebesar 5,45% menunjukkan bahwa performa keseluruhan model relatif tidak mengalami perubahan yang sangat besar ketika persyaratan kesesuaian posisi bounding box diperketat. Akan tetapi, hasil tersebut merupakan rata-rata seluruh kelas sehingga belum menunjukkan perbedaan performa yang terjadi pada kelas tertentu.

b. Perbandingan Performa Antar Kelas

Hasil validasi menunjukkan bahwa kemampuan model berbeda pada setiap kelas. Batik Parang memperoleh nilai mAP50–95 tertinggi sebesar 0,973, sedangkan Batik Garutan memperoleh nilai mAP50–95 terendah sebesar 0,157.

Untuk mengetahui besarnya penurunan performa Batik Garutan dibandingkan nilai keseluruhan model, terlebih dahulu dihitung selisih absolut sebagai berikut:

$$0,624 - 0,157 = 0,467$$

Persentase penurunan mAP50–95 Batik Garutan terhadap mAP50–95 keseluruhan model dihitung menggunakan persamaan berikut:

$$\frac{\text{mAP50--95 keseluruhan} - \text{mAP50--95 Batik Garutan}}{\text{mAP50--95 keseluruhan}} \times 100$$

$$\frac{0,624 - 0,157}{0,624} \times 100$$

$$\text{Penurunan} = 74,84$$

Hasil tersebut menunjukkan bahwa nilai mAP50–95 Batik Garutan lebih rendah sebesar 0,467 poin atau 74,84% dibandingkan nilai mAP50–95 keseluruhan model. Perbedaan tersebut menunjukkan bahwa kemampuan model dalam mengenali dan melokalisasi Batik Garutan masih jauh lebih rendah dibandingkan performa rata-rata keseluruhan kelas.

5. Hasil Pengujian Fungsional Aplikasi

Pengujian fungsional dilakukan menggunakan metode black-box

testing. Pengujian bertujuan memastikan bahwa fungsi aplikasi menghasilkan keluaran yang sesuai dengan kebutuhan.

Tabel 4. 4 Hasil Pengujian Fungsional Aplikasi

No.	Skenario Pengujian	Hasil yang Diharapkan	Hasil Aktual	Status
1	Aplikasi dibuka tanpa mengunggah citra	Halaman utama dan fitur unggah ditampilkan, sedangkan deteksi tidak dijalankan	Halaman utama, petunjuk, dan fitur unggah berhasil ditampilkan. Proses deteksi tidak dijalankan	Berhasil
2	Mengunggah citra JPG, JPEG, atau PNG	Citra dapat dibaca dan ditampilkan	Citra berhasil diunggah, dibaca, dan ditampilkan	Berhasil
3	Mengunggah citra batik yang termasuk kelas penelitian	Sistem menampilkan label, confidence, bounding box, ROI, RGB, dan kode Hex	Sistem menampilkan seluruh informasi hasil deteksi dan warna dominan	Berhasil
4	Mengunggah citra bukan motif batik	Sistem menampilkan pesan bahwa motif tidak terdeteksi	Sistem menampilkan pesan "Tidak ada motif yang terdeteksi pada gambar ini"	Berhasil
5	Mengunggah berkas citra rusak	Sistem menampilkan pesan kesalahan yang terkontrol dan aplikasi tetap	Aplikasi menampilkan galat Streamlit dan proses analisis berhenti	Tidak Berhasil

No.	Skenario Pengujian	Hasil yang Diharapkan	Hasil Aktual	Status
		berjalan		
6	Menganalisis beberapa citra secara bergantian	Aplikasi tetap berjalan dan memperbarui hasil sesuai citra terakhir	Aplikasi berhasil memperbarui hasil sesuai citra terakhir	Berhasil
No.	Skenario Pengujian	Hasil yang Diharapkan	Hasil Aktual	Status

Berdasarkan enam *skenario* pengujian, sebanyak lima *skenario* memperoleh status berhasil dan satu skenario memperoleh status tidak berhasil. Persentase keberhasilan fungsional dihitung sebagai berikut:

$$Keberhasilanfungsional = 5 \div 6 \times 100$$

$$Keberhasilanfungsional = 83,33\%$$

Hasil tersebut menunjukkan bahwa sebagian besar fungsi utama aplikasi telah berjalan sesuai kebutuhan. Kegagalan terdapat pada pengujian berkas citra rusak karena kode program belum memiliki penanganan kesalahan yang memadai pada proses pembacaan citra.

6. Pengujian Non-Fungsional

Pengujian non-fungsional dilakukan untuk mengetahui waktu respons dan stabilitas proses inti sistem. Pengujian menggunakan 45 citra yang disiapkan untuk pengukuran performa. Kumpulan citra ini digunakan khusus untuk benchmark dan dibedakan dari 24 citra uji model.

Setiap citra diproses sebanyak 10 kali sehingga diperoleh:

$$45 \times 10 = 450 \text{ proses}$$

Sebelum pengukuran utama dilakukan, model dijalankan sebanyak tiga kali sebagai tahap pemanasan.

a. Hasil Pengujian Waktu Respons

Waktu respons dalam pengujian ini merupakan waktu proses inti

yang terdiri atas waktu inferensi YOLOv8 dan waktu K-Means Clustering. Pengukuran tidak mencakup waktu pengguna memilih berkas, waktu unggah, dan waktu rendering halaman Streamlit.

Tabel 4. 5 Hasil Pengujian Waktu Respons Sistem

Parameter Pengujian	Hasil
Waktu pemuatan model	0,0375 detik
Rata-rata waktu inferensi YOLOv8	0,0777 detik
Rata-rata waktu K-Means Clustering	0,7757 detik
Rata-rata waktu total pemrosesan	0,8534 detik
Waktu total tercepat	0,1798 detik
Waktu total terlama	1,8923 detik
Throughput	1,1718 citra/detik

Rata-rata waktu total dihitung menggunakan rumus:

$$\frac{\sum_{i=1}^n t_i}{n} \quad (4.1)$$

Berdasarkan total waktu sebesar 384,0326 detik dan 450 proses, diperoleh:

$$\frac{384,0326}{450} 0,8534 \text{ detik}$$

Nilai throughput dihitung sebagai berikut:

$$\frac{450}{384,0326} 1,1718 \text{ citra/detik}$$

Hasil tersebut menunjukkan bahwa proses deteksi motif dan ekstraksi warna dominan membutuhkan rata-rata 0,8534 detik per citra. Waktu tercepat sebesar 0,1798 detik dan waktu terlama sebesar 1,8923 detik menunjukkan adanya fluktuasi waktu pemrosesan.

Fluktuasi terjadi karena jumlah objek dan luas ROI pada setiap citra berbeda. ROI yang lebih luas memiliki jumlah piksel lebih banyak sehingga membutuhkan waktu K-Means

yang lebih panjang.

b. Hasil Pengujian Stabilitas Pemrosesan

Pengujian stabilitas dilakukan untuk mengetahui kemampuan proses inti sistem ketika dijalankan secara berulang. Sebanyak 45 citra diproses masing-masing sebanyak 10 kali sehingga diperoleh 450 proses pengujian.

Hasil pengujian stabilitas disajikan pada Tabel 4.6.

Tabel 4. 6 Hasil Pengujian Stabilitas Pemrosesan

Parameter Pengujian	Hasil
Jumlah citra berbeda	45 citra
Pengulangan setiap citra	10 kali
Total proses pengujian	450 proses
Proses yang menghasilkan catatan waktu	450 proses
Pesan kegagalan pada log	Tidak ditemukan

Berdasarkan Tabel 4.6, Seluruh proses menghasilkan catatan waktu inferensi YOLOv8, waktu K-Means, dan waktu total. Selama pelaksanaan benchmark, tidak ditemukan pesan kesalahan yang menyebabkan proses berhenti sebelum hasil dicatat.

Hasil tersebut hanya menunjukkan bahwa proses inti dapat dijalankan secara berulang pada kondisi dan perangkat pengujian yang digunakan. Hasil ini tidak berarti bahwa aplikasi mempunyai tingkat keandalan 100%, karena pengujian belum mencakup berkas rusak, kehilangan berkas model, keterbatasan memori, dan penggunaan oleh beberapa pengguna secara bersamaan.

7. Hasil Pengujian Confidence Threshold

Pengujian confidence threshold dilakukan untuk mengetahui pengaruh nilai ambang keyakinan terhadap jumlah prediksi benar dan prediksi salah yang dihasilkan model. Pengujian menggunakan 24

citra data uji pada empat nilai ambang, yaitu 0,05, 0,25, 0,40, dan 0,50. Setiap prediksi dibandingkan dengan anotasi ground truth menggunakan ambang Intersection over Union sebesar 0,50. Hasil pengujian disajikan pada Tabel 4.7.

Tabel 4. 7 Hasil Pengujian Confidence Threshold

Confidence Threshold	TP	FP	FN	Precision	Recall	FDR	Jumlah Deteksi
0,05	25	18	9	0,5814	0,7353	0,4186	43
0,25	25	9	9	0,7353	0,7353	0,2647	34
0,40	24	9	10	0,7273	0,7059	0,2727	33
0,50	23	9	11	0,7188	0,6765	0,2812	32

Sumber: Hasil pengujian peneliti, 2026.

Berdasarkan Tabel 4.7, nilai confidence threshold sebesar 0,05 menghasilkan jumlah deteksi paling banyak, yaitu 43 deteksi. Namun, dari jumlah tersebut terdapat 18 false positive, sehingga nilai precision hanya sebesar 0,5814 dan false discovery rate mencapai 0,4186. Nilai tersebut menunjukkan bahwa sekitar 41,86% prediksi yang ditampilkan pada ambang 0,05 merupakan prediksi yang tidak sesuai dengan objek sebenarnya.

Pada ambang 0,25, jumlah true positive tetap sebanyak 25 dan jumlah false negative tetap sebanyak 9. Namun, jumlah false positive menurun dari 18 menjadi 9. Kondisi tersebut meningkatkan nilai precision dari 0,5814 menjadi 0,7353 tanpa menurunkan nilai recall, yang tetap sebesar 0,7353.

Peningkatan jumlah false positive pada ambang 0,05 dibandingkan ambang 0,25 dihitung sebagai berikut:

$$\frac{FP_{0,05} - FP_{0,25}}{FP_{0,25}} \times 100\%$$

$$\frac{18 - 9}{9} \times 100\%$$

Hasil tersebut menunjukkan bahwa jumlah false positive pada ambang 0,05 lebih tinggi sebesar 100% atau dua kali lipat

dibandingkan ambang 0,25.

B. Pembahasan

1. Analisis Kinerja YOLOv8

Berdasarkan hasil validasi model terbaik, YOLOv8n memperoleh nilai precision keseluruhan sebesar 0,769, recall sebesar 0,604, mAP50 sebesar 0,660, dan mAP50–95 sebesar 0,624. Nilai precision yang lebih tinggi daripada recall menunjukkan bahwa model lebih baik dalam menjaga ketepatan prediksi dibandingkan menemukan seluruh objek motif yang terdapat pada citra.

Nilai precision sebesar 0,769 menunjukkan bahwa sebagian besar objek yang diprediksi sebagai motif batik sesuai dengan objek sebenarnya. Meskipun demikian, model masih menghasilkan beberapa prediksi yang tidak sesuai. Nilai recall sebesar 0,604 menunjukkan bahwa model mampu menemukan sekitar 60,4% objek motif yang terdapat pada data validasi. Dengan demikian, masih terdapat objek motif yang tidak berhasil dideteksi oleh model.

Nilai mAP50 sebesar 0,660 menunjukkan kemampuan model dalam menentukan kelas dan lokasi objek pada ambang Intersection over Union sebesar 0,50. Sementara itu, nilai mAP50–95 sebesar 0,624 menunjukkan performa model ketika kesesuaian posisi bounding box dievaluasi menggunakan beberapa ambang IoU yang lebih ketat, yaitu mulai dari 0,50 sampai 0,95.

Selisih antara mAP50 dan mAP50–95 sebesar 0,036 atau 5,45% menunjukkan bahwa performa keseluruhan model tidak mengalami penurunan yang terlalu besar ketika kriteria kesesuaian posisi bounding box diperketat. Namun, nilai tersebut merupakan rata-rata seluruh kelas sehingga belum menggambarkan perbedaan performa pada setiap jenis motif.

Batik Parang memperoleh performa tertinggi dengan precision sebesar 0,990, recall sebesar 1,000, mAP50 sebesar 0,995, dan mAP50–95 sebesar 0,973. Nilai tersebut menunjukkan bahwa model mampu

menemukan seluruh objek Batik Parang pada data validasi dan menghasilkan sedikit prediksi yang tidak sesuai.

Karakteristik Batik Parang yang memiliki pola diagonal, pengulangan bentuk, dan susunan ornamen yang relatif konsisten dapat membantu model mempelajari fitur visual kelas tersebut. Karakteristik yang lebih seragam membuat fitur Batik Parang lebih mudah dibedakan dari kelas motif lainnya.

Batik Bali memperoleh precision sebesar 0,956, recall sebesar 0,667, mAP50 sebesar 0,846, dan mAP50–95 sebesar 0,792. Nilai precision yang tinggi menunjukkan bahwa sebagian besar prediksi Batik Bali yang dihasilkan model sesuai dengan objek sebenarnya. Namun, nilai recall sebesar 0,667 menunjukkan bahwa masih terdapat beberapa objek Batik Bali yang belum berhasil ditemukan.

Perbedaan antara precision dan recall tersebut menunjukkan bahwa model cukup tepat ketika memberikan prediksi Batik Bali, tetapi belum mampu mendeteksi seluruh variasi objek yang tersedia. Kondisi ini dapat berkaitan dengan perbedaan bentuk ornamen, komposisi warna, ukuran motif, dan kondisi pencahayaan pada citra.

Pada Batik Betawi, model memperoleh precision sebesar 0,596, recall sebesar 0,778, mAP50 sebesar 0,671, dan mAP50–95 sebesar 0,670. Nilai recall yang lebih tinggi daripada precision menunjukkan bahwa model cukup mampu menemukan objek Batik Betawi, tetapi masih menghasilkan beberapa prediksi yang tidak tepat.

Kondisi tersebut menunjukkan kemungkinan adanya bagian citra atau motif lain yang mempunyai karakteristik visual menyerupai Batik Betawi. Kemiripan bentuk, tekstur, atau warna antarkelas dapat menyebabkan model menghasilkan false positive sehingga menurunkan nilai precision.

Batik Sekar memperoleh precision sebesar 0,769, recall sebesar 0,377, mAP50 sebesar 0,588, dan mAP50–95 sebesar 0,528. Nilai precision tersebut menunjukkan bahwa prediksi Batik Sekar yang

dihasilkan model relatif tepat. Namun, nilai recall yang rendah menunjukkan bahwa sebagian besar objek Batik Sekar pada data validasi belum berhasil ditemukan.

Rendahnya recall Batik Sekar dapat dipengaruhi oleh variasi bentuk ornamen, ukuran motif, komposisi warna, dan kondisi pengambilan citra. Motif yang berukuran kecil atau memiliki detail visual yang rapat juga dapat menjadi lebih sulit dikenali setelah citra disesuaikan menjadi ukuran masukan 640×640 piksel.

Batik Garutan memperoleh performa terendah dengan nilai precision sebesar 0,536, recall sebesar 0,200, mAP50 sebesar 0,199, dan mAP50–95 sebesar 0,157. Nilai mAP50–95 tersebut lebih rendah sebesar 74,84% dibandingkan nilai mAP50–95 keseluruhan model. Nilai recall yang hanya mencapai 0,200 menunjukkan bahwa sebagian besar objek Batik Garutan pada data validasi belum berhasil ditemukan. Hasil pada confusion matrix juga menunjukkan bahwa objek Batik Garutan lebih sering tidak terdeteksi atau diprediksi sebagai background. Kondisi tersebut mengindikasikan bahwa model belum mampu membentuk representasi fitur Batik Garutan secara konsisten.

Dari sudut pandang ekstraksi fitur, Citra Batik Garutan yang digunakan dalam dataset penelitian menunjukkan detail ornamen yang relatif kecil, rapat, dan kompleks. Lapisan awal backbone YOLOv8 mengekstraksi karakteristik visual berupa garis, tepi, perubahan intensitas, dan tekstur lokal. Apabila garis ornamen berukuran tipis, saling berdekatan, atau mempunyai kontras yang rendah terhadap warna dasar kain, respons fitur yang dihasilkan dapat menjadi lemah. Setelah citra disesuaikan menjadi ukuran 640×640 piksel dan melewati proses downsampling, sebagian detail lokal dapat kehilangan ketajamannya. Akibatnya, pola Batik Garutan dapat terlihat menyerupai tekstur umum kain atau motif dari kelas lain pada peta fitur model.

Variasi pola, warna, skala ornamen, pencahayaan, dan sudut

pengambilan gambar dalam kelas Batik Garutan juga dapat menghasilkan variasi fitur yang tinggi dalam satu kelas. Kondisi tersebut menyebabkan model lebih sulit mempelajari karakteristik yang tetap sebagai pembeda Batik Garutan. Pada saat yang sama, kemiripan tekstur dengan motif lain dapat menyebabkan skor kelas Batik Garutan menjadi rendah. Apabila respons klasifikasi model terhadap Batik Garutan rendah, prediksi dapat tidak dipertahankan sebagai deteksi yang benar. Pada confusion matrix, kondisi tersebut ditunjukkan sebagai objek ground truth Batik Garutan yang diprediksi sebagai background. Kesalahan tersebut meningkatkan jumlah false negative sehingga menurunkan nilai recall dan Average Precision.

Kualitas anotasi bounding box juga dapat memengaruhi hasil pelokalan objek. Motif batik memiliki pola berulang dan tidak selalu mempunyai batas objek yang tegas. Apabila sebagian anotasi mencakup area kain yang luas, sedangkan anotasi lainnya hanya mencakup bagian ornamen tertentu, model menerima target lokasi yang tidak konsisten. Bounding box yang terlalu luas dapat memasukkan bagian latar belakang atau pola lain, sedangkan bounding box yang terlalu sempit dapat menghilangkan sebagian karakteristik penting motif. Ketidakkonsistenan tersebut menyebabkan cabang regresi YOLOv8 kesulitan mempelajari ukuran dan posisi objek secara stabil sehingga nilai Intersection over Union terhadap ground truth menjadi rendah.

Selisih antara mAP50 Batik Garutan sebesar 0,199 dan mAP50-95 sebesar 0,157 adalah 0,042. Selisih tersebut menunjukkan bahwa penggunaan ambang IoU yang lebih ketat memang menurunkan performa pelokalan, tetapi bukan menjadi satu-satunya penyebab rendahnya hasil. Penyebab yang lebih dominan terlihat dari nilai recall sebesar 0,200 dan banyaknya objek yang tidak berhasil dikenali. Dengan demikian, rendahnya performa Batik Garutan mengindikasikan kombinasi antara kompleksitas tekstur lokal, hilangnya detail setelah perubahan ukuran, tingginya variasi visual dalam kelas, kemiripan fitur

dengan kelas lain, dan kemungkinan ketidakkonsistenan anotasi bounding box.

Data validasi Batik Garutan hanya terdiri atas delapan citra dengan 15 objek. Jumlah tersebut menyebabkan hasil evaluasi menjadi sensitif terhadap setiap kesalahan deteksi. Satu objek yang tidak berhasil ditemukan dapat memberikan pengaruh relatif besar terhadap nilai recall dan Average Precision. Oleh karena itu, peningkatan performa Batik Garutan perlu dilakukan melalui penambahan data yang lebih representatif, penyeimbangan variasi objek, pemeriksaan ulang anotasi bounding box, dan pelatihan ulang model menggunakan data dengan kualitas serta cakupan anotasi yang lebih konsisten.

Secara keseluruhan, hasil validasi menunjukkan bahwa YOLOv8n telah mampu mendeteksi lima kelas motif batik, tetapi performanya belum merata. Model memberikan hasil terbaik pada Batik Parang dan Batik Bali, sedangkan Batik Garutan dan Batik Sekar masih memerlukan peningkatan.

Peningkatan performa dapat dilakukan melalui penambahan jumlah dan variasi citra, penyeimbangan jumlah objek setiap kelas, pemeriksaan ulang anotasi bounding box, serta pelatihan ulang menggunakan data yang lebih representatif. Hasil pengujian confidence threshold menunjukkan bahwa ambang 0,25 memberikan keseimbangan terbaik karena mempertahankan 25 true positive dan nilai recall sebesar 0,7353, tetapi mengurangi jumlah false positive dari 18 menjadi 9 dibandingkan ambang 0,05.

2. Analisis Hasil Deteksi pada Aplikasi

Model terbaik hasil pelatihan disimpan dalam file *best.pt* dan diintegrasikan ke dalam aplikasi berbasis *Streamlit*. Pengguna dapat mengunggah citra berformat JPG, JPEG, atau PNG. Sistem kemudian memproses citra menggunakan model YOLOv8 dan menampilkan gambar hasil deteksi berupa *bounding box*, label motif, dan nilai *confidence*.

163

4

Hasil implementasi menunjukkan bahwa sistem dapat menjalankan proses deteksi secara otomatis setelah pengguna mengunggah gambar. Nilai *confidence* yang ditampilkan menunjukkan tingkat keyakinan model terhadap kelas motif yang diprediksi. Semakin mendekati nilai 1, semakin tinggi keyakinan model terhadap hasil prediksi tersebut. Meskipun demikian, nilai *confidence* tidak dapat dianggap sebagai akurasi keseluruhan model karena *confidence* hanya menunjukkan keyakinan terhadap satu hasil prediksi pada citra tertentu.

9

2

Pada contoh pengujian Batik Parang, sistem berhasil mendeteksi motif dengan nilai *confidence* sebesar 0,99. Hasil tersebut sesuai dengan evaluasi per kelas, di mana Batik Parang memperoleh nilai *precision*, *recall*, dan *mAP* tertinggi. Kesesuaian antara hasil validasi dan pengujian aplikasi menunjukkan bahwa model mampu mengenali pola Batik Parang secara konsisten.

3. Analisis Pengaruh Confidence Threshold terhadap Fungsionalitas Aplikasi

Hasil pengujian menunjukkan bahwa nilai *confidence threshold* berpengaruh langsung terhadap jumlah prediksi yang diteruskan dan ditampilkan pada antarmuka Streamlit. Pada ambang 0,05, model menghasilkan 43 deteksi yang terdiri atas 25 true positive dan 18 false positive. Sementara itu, pada ambang 0,25, model menghasilkan 34 deteksi yang terdiri atas 25 true positive dan 9 false positive. Jumlah true positive dan false negative pada kedua ambang tersebut tidak berubah sehingga nilai *recall* tetap sebesar 0,7353.

Secara matematis, nilai *precision* pada ambang 0,05 diperoleh dari pembagian 25 true positive dengan 43 seluruh prediksi positif sehingga menghasilkan nilai 0,5814. Pada ambang 0,25, nilai *precision* diperoleh dari pembagian 25 true positive dengan 34 seluruh prediksi positif sehingga menghasilkan nilai 0,7353. Penurunan ambang dari 0,25 menjadi 0,05 menambah sembilan kotak deteksi, tetapi tidak menambah objek yang berhasil dikenali dengan benar. Dengan demikian, seluruh

deteksi tambahan tersebut merupakan false positive.

Jumlah false positive pada ambang 0,05 meningkat dari 9 menjadi 18 atau sebesar 100% dibandingkan ambang 0,25. Peningkatan tersebut menyebabkan false discovery rate naik dari 0,2647 menjadi 0,4186. Artinya, pada ambang 0,05, sekitar 41,86% prediksi yang ditampilkan aplikasi tidak sesuai dengan objek sebenarnya. Hasil tersebut menunjukkan bahwa penggunaan ambang yang terlalu rendah memang mempertahankan lebih banyak kandidat deteksi, tetapi kandidat tambahan tidak selalu memberikan manfaat terhadap kemampuan sistem menemukan objek.

Dampak false positive tidak hanya terjadi pada hasil deteksi YOLOv8. Setiap bounding box yang lolos dari penyaringan ditampilkan pada antarmuka Streamlit sebagai objek motif beserta label kelas dan nilai confidence. Apabila kotak tersebut merupakan false positive, pengguna akan menerima informasi kelas motif yang tidak sesuai dengan objek sebenarnya. Kondisi ini menunjukkan bahwa keberhasilan aplikasi dalam menampilkan hasil tidak selalu menjamin kebenaran informasi yang disajikan.

Kesalahan pada tahap deteksi juga diteruskan ke proses K-Means Clustering. Setiap bounding box digunakan untuk mengambil Region of Interest. Apabila bounding box merupakan false positive, ROI yang dipotong tidak mewakili motif yang sebenarnya. Namun, sistem tetap mengelompokkan piksel pada ROI tersebut dan menampilkan warna dominan dalam bentuk nilai RGB dan kode heksadesimal. Dengan demikian, kesalahan deteksi pada lapisan YOLOv8 dapat merambat menjadi kesalahan informasi warna dominan pada lapisan K-Means.

Jumlah kotak yang lebih banyak juga berpotensi meningkatkan beban pemrosesan karena setiap ROI harus diproses menggunakan K-Means Clustering. Hasil pengujian waktu menunjukkan bahwa rata-rata proses K-Means sebesar 0,7757 detik, sedangkan rata-rata inferensi YOLOv8 sebesar 0,0777 detik. Oleh karena itu, penambahan ROI yang

berasal dari false positive berpotensi menambah pekerjaan komputasi tanpa menghasilkan informasi motif yang benar.

Pada ambang 0,40 dan 0,50, jumlah false positive tetap sebanyak sembilan, tetapi jumlah true positive menurun menjadi 24 dan 23. Nilai recall juga menurun menjadi 0,7059 dan 0,6765. Berdasarkan hasil tersebut, ambang 0,25 memberikan keseimbangan terbaik pada data uji penelitian karena mempertahankan jumlah true positive dan recall yang sama dengan ambang 0,05, tetapi menurunkan jumlah false positive sebesar 50%. Oleh karena itu, nilai confidence threshold sebesar 0,25 lebih layak digunakan pada aplikasi dibandingkan nilai 0,05.

4. Analisis Ekstraksi Warna Dominan Menggunakan *K-Means* Setelah YOLOv8 mendeteksi area motif, *koordinat bounding box* digunakan untuk mengambil *Region of Interest* atau *area* motif yang terdeteksi. *Area* tersebut selanjutnya diproses menggunakan metode *K-Means Clustering*. *Piksel* dalam *area* motif dikelompokkan ke dalam tiga *cluster* warna berdasarkan kemiripan nilai RGB. Cluster dengan jumlah *piksel* terbanyak dipilih sebagai warna dominan.

Penggunaan *K-Means* pada *area bounding box* membuat proses ekstraksi warna lebih terfokus dibandingkan apabila analisis dilakukan terhadap keseluruhan citra. Dengan cara ini, warna latar belakang di luar area motif dapat dikurangi pengaruhnya terhadap hasil ekstraksi. Warna dominan kemudian ditampilkan dalam format RGB, *kode Hex*, dan kotak visual warna agar lebih mudah dipahami oleh pengguna.

Hasil ekstraksi warna dominan dapat berbeda meskipun motif yang dideteksi sama. Perbedaan tersebut dipengaruhi oleh pencahayaan, kualitas kamera, sudut pengambilan gambar, komposisi warna kain, dan luas *area bounding box*. Oleh karena itu, hasil warna dominan menunjukkan warna yang paling banyak terdapat pada *area* terdeteksi, bukan warna baku atau identitas mutlak dari suatu jenis motif batik.

5. Kesesuaian Hasil dengan Tujuan Penelitian

Tujuan pertama penelitian adalah menerapkan YOLOv8 untuk

mendeteksi dan mengidentifikasi motif batik. Tujuan tersebut telah tercapai karena model mampu mengenali lima kelas motif, yaitu Batik Bali, Batik Betawi, Batik Garutan, Batik Parang, dan Batik Sekar, serta menampilkan lokasi motif dalam bentuk *bounding box*.

Tujuan kedua adalah menentukan warna dominan pada area motif batik. Tujuan tersebut juga telah tercapai melalui penerapan *K-Means Clustering* terhadap area motif hasil deteksi. Sistem mampu menampilkan warna dominan dalam format RGB dan Hex.

Tujuan ketiga adalah mengevaluasi kemampuan sistem. Berdasarkan hasil validasi, diperoleh *precision* sebesar 0,769, *recall* sebesar 0,604, *mAP50* sebesar 0,660, dan *mAP50-95* sebesar 0,624. Hasil tersebut menunjukkan bahwa sistem cukup mampu mendeteksi motif batik, meskipun masih terdapat perbedaan performa yang cukup besar antar kelas.

6. Perbandingan dengan Penelitian Terdahulu

Perbandingan kinerja dilakukan untuk mengetahui posisi hasil penelitian ini terhadap beberapa penelitian terdahulu mengenai pengenalan motif batik. Nilai kinerja seluruh penelitian diseragamkan ke dalam skala persentase agar dapat ditampilkan dalam satu tabel. Nilai *mAP50* penelitian ini sebesar 0,660 dikonversi menjadi 66,00%. Meskipun demikian, jenis metrik dan tugas komputasi setiap penelitian berbeda sehingga hasil perbandingan tidak digunakan untuk menyatakan keunggulan secara mutlak. Matriks komparasi kinerja disajikan pada Tabel 4.8.

Tabel 4. 8 Matriks Komparasi Kinerja Saintifik

No	Peneliti (Tahun)	Metode Utama	Dataset (Ringkas)	Tugas	Metrik	Nilai (%)	Selisih	Deviasi
1	Dani & Handayani (2024)	GLCM + CNN MobileNetV2	3.223 citra, 5 motif	Klasifikasi	Akurasi	99,00	33,00	33,33%

No	Peneliti (Tahun)	Metode Utama	Dataset (Ringkas)	Tugas	Metrik	Nilai (%)	Selisih	Deviasi
2	Prayoga (2023)	CNN (7 arsitektur)	600 citra, 3 kelas	Klasifikasi	Akurasi	87,83	21,83	24,85%
3	Whardana & Aziz (2025)	Otsu + Median + CNN	Internet & TMII	Klasifikasi	Akurasi	85,36	19,36	22,68%
4	Penelitian ini	YOLOv8n + K-Means	586 citra, 5 kelas	Deteksi	mAP50	66,00	0,00	0,00%

Sumber: Diolah peneliti berdasarkan Dani dan Handayani (2024), Prayoga (2023), Whardana dan Aziz (2025), serta hasil penelitian, 2026. Persentase deviasi relatif dihitung dengan membandingkan selisih nilai kinerja penelitian terdahulu dan mAP50 penelitian ini terhadap nilai penelitian terdahulu. Persamaan yang digunakan adalah sebagai berikut:

$$\frac{|M_i - M_p|}{M_i} \times 100$$

Keterangan:

(M_i) = nilai metrik kinerja penelitian terdahulu;

(M_p) = nilai mAP50 penelitian ini, yaitu 66,00%;

$(Deviasi_i)$ = deviasi relatif terhadap penelitian terdahulu.

Deviasi terhadap penelitian Dani dan Handayani (2024) dihitung sebagai berikut:

$$\begin{aligned} & \frac{|99,00 - 66,00|}{99,00} \times 100\% \\ & = 33,33\% \end{aligned}$$

Deviasi terhadap penelitian Prayoga (2023) dihitung sebagai berikut:

$$\begin{aligned} & \frac{|87,83 - 66,00|}{87,83} \times 100\% \\ & = 24,85 \end{aligned}$$

Deviasi terhadap penelitian Whardana dan Aziz (2025) dihitung sebagai berikut:

$$\frac{|85,36 - 66,00|}{85,36} \times 100$$
$$= 22,68$$

Berdasarkan Tabel 4.8, nilai mAP50 penelitian ini secara numerik lebih rendah 33,00 poin dari akurasi Dani dan Handayani, 21,83 poin dari akurasi Prayoga, serta 19,36 poin dari akurasi Whardana dan Aziz. Deviasi relatif terbesar terdapat pada perbandingan dengan Dani dan Handayani sebesar 33,33%, sedangkan deviasi terkecil terdapat pada perbandingan dengan Whardana dan Aziz sebesar 22,68%.

Perbedaan tersebut perlu ditafsirkan secara hati-hati karena akurasi klasifikasi dan mAP50 deteksi objek bukan metrik yang setara. Penelitian terdahulu mengukur kemampuan model dalam menentukan satu kelas terhadap citra secara keseluruhan. Sementara itu, mAP50 dalam penelitian ini mengukur kemampuan model dalam menentukan kelas sekaligus kesesuaian lokasi bounding box pada ambang Intersection over Union sebesar 0,50. Deteksi objek mempunyai tugas yang lebih kompleks karena model harus melakukan klasifikasi dan pelokalan objek secara bersamaan.

Perbedaan jumlah data, kelas motif, kualitas citra, pembagian dataset, dan metode evaluasi juga memengaruhi nilai kinerja yang diperoleh. Dani dan Handayani menggunakan 3.223 citra, sedangkan penelitian ini menggunakan 586 citra. Jumlah data yang lebih terbatas dapat mengurangi variasi fitur yang dipelajari model, terutama pada kelas Batik Garutan dan Batik Sekar yang menunjukkan performa lebih rendah.

Walaupun nilai mAP50 penelitian ini secara numerik berada di bawah nilai akurasi ketiga penelitian terdahulu, sistem yang dikembangkan mempunyai keluaran yang lebih luas. Sistem tidak hanya menghasilkan label kelas, tetapi juga menentukan lokasi motif melalui bounding box dan mengekstraksi warna dominan dari Region of Interest menggunakan K-Means Clustering. Dengan demikian, kontribusi

penelitian ini terletak pada integrasi deteksi lokasi motif dan analisis warna dominan dalam satu alur aplikasi, bukan pada klaim bahwa nilai mAP50 secara langsung lebih tinggi daripada akurasi penelitian klasifikasi terdahulu.

7. Kelebihan Sistem

Sistem memiliki beberapa kelebihan. Pertama, sistem mampu melakukan deteksi motif dan ekstraksi warna dominan secara otomatis dalam satu aplikasi. Kedua, antarmuka Streamlit relatif sederhana sehingga pengguna hanya perlu mengunggah gambar untuk memperoleh hasil analisis. Ketiga, hasil deteksi ditampilkan secara visual melalui *bounding box*, label motif, confidence, nilai RGB, kode *Hex*, dan kotak warna. Keempat, penggunaan model YOLOv8 memungkinkan proses deteksi dilakukan dalam waktu yang relatif cepat.

8. Keterbatasan Sistem

Sistem yang dikembangkan masih memiliki beberapa keterbatasan. Pertama, performa model belum merata pada seluruh kelas motif. Hasil validasi menunjukkan bahwa Batik Garutan dan Batik Sekar memperoleh performa lebih rendah dibandingkan kelas lainnya. Batik Garutan memperoleh nilai recall sebesar 0,200 dan mAP50–95 sebesar 0,157, sedangkan Batik Sekar memperoleh nilai recall sebesar 0,377 dan mAP50–95 sebesar 0,528. Kondisi tersebut menunjukkan bahwa model masih mengalami kesulitan dalam mengenali beberapa variasi motif, terutama motif yang mempunyai detail lokal rapat, ukuran ornamen kecil, kemiripan tekstur dengan kelas lain, serta variasi visual yang tinggi. Selain itu, data validasi hanya terdiri atas 49 citra dengan 57 objek sehingga hasil evaluasi belum sepenuhnya mewakili seluruh variasi motif batik yang mungkin ditemukan pada kondisi penggunaan sebenarnya.

Kedua, implementasi awal aplikasi menggunakan nilai confidence threshold sebesar 0,05. Berdasarkan hasil pengujian terhadap 24 citra data uji, ambang 0,05 menghasilkan 25 true positive, 18 false positive, 9 false negative, nilai precision sebesar 0,5814, recall sebesar

0,7353, dan false discovery rate sebesar 0,4186. Pada ambang 0,25, jumlah true positive dan false negative tetap sama, tetapi jumlah false positive menurun menjadi sembilan dan nilai precision meningkat menjadi 0,7353. Hasil tersebut menunjukkan bahwa penggunaan ambang 0,05 meningkatkan jumlah false positive sebesar 100% dibandingkan ambang 0,25. Dengan demikian, penggunaan ambang yang terlalu rendah menjadi keterbatasan implementasi awal karena meningkatkan risiko ditampilkannya bounding box dan label motif yang tidak sesuai dengan objek sebenarnya.

Ketiga, kesalahan pada tahap deteksi YOLOv8 dapat diteruskan ke tahap ekstraksi warna dominan. Setiap bounding box yang lolos dari penyaringan digunakan untuk mengambil Region of Interest dan diproses menggunakan K-Means Clustering. Apabila bounding box merupakan false positive atau tidak sesuai dengan batas motif sebenarnya, piksel yang dikelompokkan tidak mewakili objek motif yang benar. Meskipun demikian, aplikasi tetap menghasilkan nilai RGB, kode heksadesimal, dan visualisasi warna dari area tersebut. Hal ini menyebabkan kesalahan pada lapisan deteksi dapat merambat menjadi kesalahan informasi warna dominan pada lapisan K-Means.

Keempat, hasil ekstraksi warna dominan masih dipengaruhi oleh kondisi pencahayaan, bayangan, kualitas kamera, sudut pengambilan gambar, komposisi warna kain, dan ketepatan posisi bounding box. Penelitian ini belum menggunakan warna acuan atau ground truth warna sehingga belum dapat menghitung tingkat ketepatan hasil warna dominan secara kuantitatif. Evaluasi warna saat ini masih terbatas pada kemampuan sistem mengelompokkan piksel, memilih kelompok dengan jumlah anggota terbanyak, dan menampilkan hasilnya dalam bentuk nilai RGB, kode heksadesimal, serta kotak visual warna.

Kelima, pengujian fungsional menunjukkan bahwa aplikasi belum mampu menangani berkas citra rusak secara terkontrol. Ketika berkas rusak diunggah, aplikasi menampilkan galat Streamlit dan proses

analisis berhenti. Kondisi tersebut menunjukkan bahwa mekanisme validasi berkas dan penanganan kesalahan pada proses pembacaan citra masih perlu diperbaiki agar aplikasi tetap berjalan dan menampilkan pesan yang mudah dipahami pengguna.

Keenam, aplikasi masih dijalankan secara lokal pada satu perangkat dan hanya melayani satu pengguna dalam satu waktu. Pengujian waktu respons dan stabilitas juga dilakukan pada perangkat dengan spesifikasi tertentu sehingga hasil waktu pemrosesan belum dapat digeneralisasikan untuk perangkat dengan kemampuan komputasi yang berbeda. Sistem belum diuji pada lingkungan server, penggunaan secara bersamaan, perangkat bergerak, maupun pengolahan video secara waktu nyata.

Berdasarkan keterbatasan tersebut, sistem telah mampu menjalankan fungsi utama berupa deteksi motif dan ekstraksi warna dominan, tetapi hasilnya masih perlu digunakan sebagai alat bantu awal. Untuk penggunaan yang lebih luas, diperlukan penambahan dan penyeimbangan data, perbaikan anotasi, penggunaan confidence threshold sebesar 0,25 berdasarkan hasil pengujian, evaluasi warna menggunakan ground truth, peningkatan penanganan kesalahan, serta pengujian pada lingkungan dan perangkat yang lebih beragam.

9. Implikasi Hasil Penelitian

Secara praktis, sistem dapat digunakan sebagai alat bantu awal untuk mengenali motif dan warna dominan batik serta mendukung dokumentasi dan edukasi digital. Namun, sistem belum dapat menggantikan penilaian ahli karena performa deteksi belum merata dan beberapa motif memiliki kemiripan visual. Secara akademik, penelitian ini menunjukkan bahwa integrasi YOLOv8 dan K-Means Clustering dapat digunakan untuk mendeteksi lokasi motif sekaligus mengekstraksi warna dominan pada area hasil deteksi, sehingga dapat menjadi dasar pengembangan penelitian pengenalan pola tekstil dan objek budaya.

BAB V

PENUTUP

A. Kesimpulan

Berdasarkan hasil penelitian, implementasi sistem, dan pembahasan yang telah dilakukan, diperoleh kesimpulan sebagai berikut:

1. Aplikasi deteksi motif dan warna dominan batik berhasil dibangun menggunakan Streamlit dengan mengintegrasikan model YOLOv8n dan K-Means Clustering. Aplikasi mampu menerima citra berformat JPG, JPEG, atau PNG, menjalankan deteksi motif, mengambil Region of Interest, mengekstraksi warna dominan, dan menampilkan hasil analisis. Berdasarkan pengujian fungsional, lima dari enam skenario berhasil dijalankan sehingga diperoleh tingkat keberhasilan sebesar 83,33%. Kegagalan masih ditemukan pada penanganan berkas citra rusak.
2. Metode YOLOv8n berhasil diterapkan untuk mendeteksi dan mengidentifikasi jenis serta lokasi lima kelas motif batik, yaitu Batik Bali, Batik Betawi, Batik Garutan, Batik Parang, dan Batik Sekar. Hasil deteksi ditampilkan dalam bentuk nama kelas motif, nilai confidence, dan koordinat bounding box. Performa antarkelas belum merata karena Batik Parang memperoleh hasil tertinggi, sedangkan Batik Garutan memperoleh hasil terendah.
3. K-Means Clustering berhasil diterapkan untuk menentukan warna dominan pada Region of Interest yang diperoleh dari bounding box hasil deteksi YOLOv8n. Piksel pada setiap ROI dikelompokkan menjadi tiga cluster berdasarkan kedekatan nilai RGB. Cluster dengan jumlah piksel terbanyak dipilih sebagai warna dominan, kemudian ditampilkan dalam bentuk nilai RGB, kode heksadesimal, dan visualisasi warna. Ketepatan warna dominan belum diukur secara kuantitatif karena penelitian belum menggunakan warna acuan atau ground truth warna.
4. Berdasarkan hasil validasi empiris model terbaik yang tersimpan dalam berkas `best.pt` terhadap 49 citra dengan 57 objek, YOLOv8n memperoleh

1 nilai precision sebesar 0,769, recall sebesar 0,604, mAP50 sebesar 0,660, dan mAP50–95 sebesar 0,624. Hasil tersebut menunjukkan bahwa model lebih baik dalam menjaga ketepatan prediksi positif daripada menemukan seluruh objek motif. Seluruh nilai tersebut berasal dari keluaran validasi aktual model dan tidak menggunakan data simulasi. Pengujian confidence threshold terhadap 24 citra uji menunjukkan bahwa ambang 0,25 memberikan keseimbangan terbaik dengan menghasilkan 25 true positive, 9 false positive, 9 false negative, precision sebesar 0,7353, dan recall sebesar 0,7353. Dibandingkan ambang 0,05, penggunaan ambang 0,25 mengurangi jumlah false positive sebesar 50% tanpa mengurangi jumlah objek yang berhasil dideteksi.

B. Saran

Berdasarkan hasil penelitian dan keterbatasan sistem, rekomendasi teknis untuk penelitian selanjutnya adalah sebagai berikut:

1. Performa deteksi Batik Garutan masih rendah, yaitu memperoleh recall sebesar 0,200, mAP50 sebesar 0,199, dan mAP50–95 sebesar 0,157. Penelitian selanjutnya disarankan menambah serta menyeimbangkan jumlah citra Batik Garutan, memeriksa konsistensi anotasi bounding box, dan menguji penambahan modul attention mechanism, seperti Convolutional Block Attention Module atau Squeeze-and-Excitation, pada bagian backbone atau neck YOLOv8. Pengujian harus membandingkan model dasar YOLOv8n dengan model yang telah ditambahkan modul perhatian menggunakan pembagian dataset yang sama serta metrik precision, recall, mAP50, dan mAP50–95.
2. Ekstraksi warna dominan masih menggunakan ruang warna RGB sehingga hasil centroid K-Means dipengaruhi oleh pencahayaan, bayangan, dan kualitas kamera. Penelitian selanjutnya disarankan mengonversi piksel ROI ke ruang warna CIELAB sebelum proses K-Means serta menerapkan normalisasi pencahayaan atau color constancy. Ketepatan warna perlu diuji menggunakan warna acuan dengan

menghitung selisih warna, misalnya ΔE , sehingga kestabilan hasil warna dominan dapat dievaluasi secara kuantitatif.

3. Aplikasi belum mampu menangani berkas citra rusak secara terkontrol sehingga proses analisis berhenti dan menampilkan galat Streamlit. Pengembangan selanjutnya disarankan menambahkan validasi tipe dan keterbacaan berkas serta blok try-except pada proses pembacaan citra, inferensi YOLOv8, pemotongan ROI, dan K-Means Clustering. Sistem perlu diuji kembali menggunakan berkas kosong, ekstensi palsu, citra terpotong, dan berkas rusak untuk memastikan aplikasi tetap berjalan serta menampilkan pesan kesalahan yang mudah dipahami pengguna.
4. Nilai confidence threshold sebesar 0,25 memberikan hasil terbaik pada 24 citra uji, tetapi jumlah data tersebut masih terbatas untuk menetapkan ambang secara umum. Penelitian selanjutnya disarankan memvalidasi ambang 0,25 menggunakan jumlah citra uji yang lebih besar dan seimbang pada setiap kelas. Evaluasi dilakukan dengan membandingkan true positive, false positive, false negative, precision, recall, dan false discovery rate. Apabila distribusi nilai confidence berbeda antarkelas, pengujian dapat dilanjutkan dengan ambang khusus setiap kelas.