

**EVALUASI ALGORITMA *CNN* DAN *FASTER R-CNN* UNTUK
KLASIFIKASI OTOMATIS TINGKAT KEMATANGAN TANDAN BUAH
SEGAR KELAPA SAWIT**

SKRIPSI

Oleh

Dinda Armeylia Putri

NPM 2115031109



JURUSAN TEKNIK ELEKTRO

FAKULTAS TEKNIK

UNIVERSITAS LAMPUNG

BANDAR LAMPUNG

2025

**EVALUASI ALGORITMA *CNN* DAN *FASTER R-CNN* UNTUK
KLASIFIKASI OTOMATIS TINGKAT KEMATANGAN TANDAN BUAH
SEGAR KELAPA SAWIT**

Oleh

Dinda Armeylia Putri

2115031109

SKRIPSI

**Diajukan Sebagai Salah Satu Syarat Untuk Mencapai Gelar Sarjana Teknik
Pada Jurusan Teknik Elektro Fakultas Teknik Universitas Lampung**



JURUSAN TEKNIK ELEKTRO

FAKULTAS TEKNIK

UNIVERSITAS LAMPUNG

BANDAR LAMPUNG

2025

ABSTRAK

EVALUASI ALGORITMA *CNN* DAN *FASTER R-CNN* UNTUK KLASIFIKASI OTOMATIS TINGKAT KEMATANGAN TANDAN BUAH SEGAR KELAPA SAWIT

Oleh

Dinda Armeylia Putri

Derajat kematangan tandan buah segar (TBS) kelapa sawit menentukan tingkat rendemen, kadar asam lemak bebas, dan stabilitas mutu minyak. Ketidaktepatan panen dapat memicu kerugian nilai di sepanjang rantai pasok. Penilaian manual terhadap tandan sawit yang akan dipanen bersifat subjektif dan tidak konsisten karena dipengaruhi variasi pencahayaan, oklusi pelepah, jarak pengambilan, serta heterogenitas varietas. Otomasi berbasis *computer vision* diperlukan agar keputusan panen terstandar dan dapat diskalakan. Studi ini membandingkan tiga pendekatan *deep learning* untuk klasifikasi kematangan TBS yaitu CNN standar, CNN ber-backbone ResNet50, dan Faster R-CNN ResNet50. Tiga himpunan data digunakan: dataset citra TBS dari perkebunan, 8.400 citra TBS beranotasi rapi dari Roboflow, dan subset Roboflow 625 citra. Evaluasi mencakup akurasi, precision, recall, F1-score, mean Average Precision (mAP), dan kecepatan inferensi (FPS), sehingga trade-off presisi-kinerja dapat dinilai. Hasil pemodelan menegaskan adanya perbedaan kinerja lintas domain. Pada dataset citra lapangan, CNN standar memiliki keunggulan pada penerapan *edge* berdaya terbatas dengan nilai akurasi 83,87%, mAP 84,29%, dan 464 FPS. Spesifikasi ini cukup untuk klasifikasi tunggal *near real-time*. Pada Roboflow 8.400 citra, Faster R-CNN ResNet50 menghasilkan akurasi 92,72%, precision 100%, recall 91,73%, F1 95,63%, mAP 86,25%, dengan 396 FPS, dimana kinerja tinggi ini didorong anotasi bersih dan distribusi visual lebih terkontrol, meski biaya komputasinya lebih besar karena proses deteksi dua tahap dan penggunaan *Region Proposal Network* (RPN). Pada subset Roboflow 625 citra, model Faster R-CNN ResNet50 tetap terbaik (akurasi 92,42%, mAP 81,00%, 307 FPS). CNN ResNet50 konsisten berada di antara keduanya, merefleksikan kapasitas fitur yang lebih kuat dari CNN standar namun tanpa mekanisme proposal wilayah. Secara praktis, CNN standar cocok untuk inspeksi *on-device* berbiaya rendah. Sedangkan Faster R-CNN tepat untuk deteksi multiobjek pada data terstruktur. Pemilihan model sebaiknya mempertimbangkan karakteristik domain, target latensi, serta ketersediaan komputasi.

Kata kunci: Kelapa Sawit, Klasifikasi , *CNN*, *ResNet50*, *Faster R-CNN*, *Deep Learning*.

ABSTRACT

EVALUATION OF CNN AND FASTER R-CNN ALGORITHMS FOR AUTOMATIC CLASSIFICATION OF OIL PALM FRESH FRUIT BUNCH (FFB) RIPENESS LEVELS

BY

Dinda Armeylia Putri

Fresh fruit bunch (FFB) ripeness governs oil yield, free-fatty-acid formation, and product stability; misgrading at harvest propagates losses along the supply chain. Manual grading is subjective and inconsistent under varying illumination, frond occlusion, camera distance, and cultivar heterogeneity. Scalable, computer-vision automation is therefore needed to standardize harvest decisions. This study compares three deep-learning approaches for FFB ripeness classification: standard CNN, CNN with ResNet50 backbone, and Faster R-CNN ResNet50. Three datasets are used: on-farm field dataset imagery, a curated Roboflow set of 8,400 annotated images, and a Roboflow subset of 625 images. The models are evaluated in term of accuracy, precision, recall, F1-score, mean Average Precision (mAP), and inference speed (FPS) to quantify precision–throughput trade-offs. Results confirm domain-dependent performance. On field imagery, the standard CNN is preferable for resource-constrained edge deployment with 83.87% accuracy, 84.29% mAP, and 464 FPS. It is sufficient for single-object, near-real-time classification. In addition to Roboflow-8,400, Faster R-CNN ResNet50 attains 92.72% accuracy, 100% precision, 91.73% recall, 95.63% F1, and 86.25% mAP at 396 FPS. This robust performance benefits from clean annotations and controlled visual distributions, albeit with higher compute cost due to the two-stages detection and the used of Region Proposal Network (RPN). On the 625-image Roboflow subset, the Faster R-CNN ResNet50 model remains preeminent (92.42% accuracy, 81.00% mAP, 307 FPS). The ResNet50-based CNN consistently ranks between the two, reflecting stronger feature capacity than the standard CNN but lacking region-proposal mechanisms. Practically, the plain CNN suits low-cost on-device inspection, whereas Faster R-CNN is better matched to multi-object detection on curated datasets. Model selection should consider domain characteristics, target latency, available compute, and operational risk tolerance.

Keywords: Oil Palm, Ripeness Classification, CNN, ResNet50, Faster R-CNN, Deep Learning.

Judul Skripsi

: **EVALUASI ALGORITMA CNN DAN FASTER R-CNN UNTUK KLASIFIKASI OTOMATIS TINGKAT KEMATANGAN TANDAN BUAH SEGAR KELAPA SAWIT**

Nama Mahasiswa

: **Dinda Armevia Putri**

Nomor Pokok Mahasiswa : 2115031109

Jurusan

: Teknik Elektro

Fakultas

: Teknik



Ing. Melvi, S.T., M.T., Ph.D.
NIP. 197301182000032001

Aryanto, S.T., M.T.
NIP. 199006212019031011

2. Mengetahui

Ketua Jurusan Teknik Elektro

Ketua Program Studi Teknik Elektro

Herlinawati, S.T., M.T.
NIP. 197103141999032001

Sumadi, S.T., M.T.
NIP. 197311042000031001

MENGESAHKAN

1. Tim Penguji

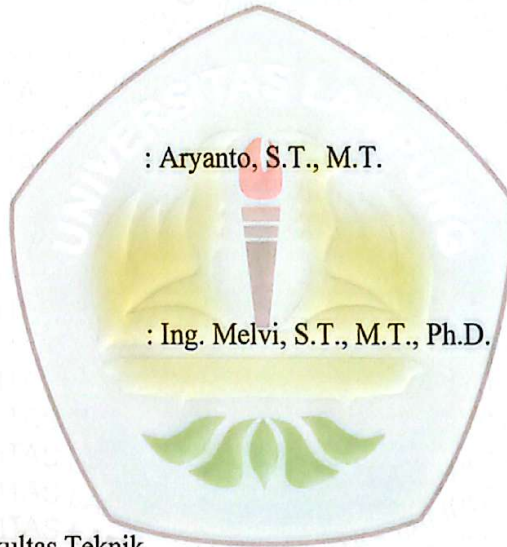
Ketua : Ing. Ardian Ulvan, S.T., M.Sc., Ph.D.


.....

Sekretaris : Aryanto, S.T., M.T.



.....

Penguji : Ing. Melvi, S.T., M.T., Ph.D.


.....

2. Dekan Fakultas Teknik




Dr. Eng. Ir. Helmy Fitriawan, S.T., M.Sc.)
NIP. 197509282001121002

Tanggal Lulus Ujian Skripsi: 30 Juli 2025

SURAT PERNYATAAN

Yang bertanda tangan dibawah ini:

Nama : Dinda Armeylia Putri

NPM : 2115031109

Dengan ini saya menyatakan bahwa skripsi saya yang berjudul “Evaluasi Algoritma *CNN* dan *Faster R-CNN* Untuk Klasifikasi Otomatis Tingkat Kematangan Tandan Buah Segar Kelapa Sawit” merupakan asli penelitian saya. Pada skripsi ini tidak terdapat karya yang pernah dilakukan orang lain dan sepengetahuan saya tidak terdapat atau diterbitkan oleh orang lain, kecuali secara tertulis diacu dalam naskah ini sebagaimana yang disebutkan dalam daftar pustaka. Selain itu, saya menyatakan bahwa skripsi ini dibuat oleh saya sendiri.

Apabila pernyataan saya tidak benar, maka saya bersedia dikenai sanksi sesuai dengan hukum yang berlaku.

Bandar Lampung, 25 Agustus 2025



Dinda Armeylia Putri

NPM. 2115031109

RIWAYAT HIDUP



Penulis lahir di Bandan Hurip, Kabupaten Lampung Selatan, pada tanggal 7 Mei 2002 sebagai anak tunggal dari Bapak Toni Andesta dan Ibu Ana Nopianah. Pendidikan dasar ditempuh di SD Negeri 2 Palas Jaya dan diselesaikan pada tahun 2014. Selanjutnya, penulis melanjutkan pendidikan menengah pertama di SMP Negeri 2 Sragi hingga lulus pada tahun 2017, kemudian menempuh pendidikan menengah atas di SMA Negeri 1 Kalianda yang diselesaikan pada tahun 2020. Pada tahun 2021, penulis diterima sebagai mahasiswa Jurusan Teknik Elektro, Fakultas Teknik, Universitas Lampung. Pada tahun 2022–2023, penulis menjadi anggota Departemen Sosial dan Kewirausahaan Himpunan Mahasiswa Teknik Elektro (HIMATRO), penulis juga aktif sebagai anggota PSDM di BEM Fakultas Teknik pada tahun 2023. Selain aktif di organisasi dan kemahasiswaan, penulis juga mengembangkan kemampuan akademik dan praktis melalui kegiatan laboratorium, penulis dipercaya sebagai asisten di Laboratorium Telekomunikasi pada tahun 2024. Pada tahun 2022, penulis berhasil meraih pendanaan Program Kreativitas Mahasiswa (PKM) dan menjadi Finalis PLN ICE. Tahun berikutnya, penulis terpilih sebagai finalis Lomba Inovasi Digital Mahasiswa (LIDM) diselenggarakan oleh Kementerian Pendidikan, Kebudayaan, Riset, dan Teknologi di Bandung. Pada tahun 2024, penulis terpilih sebagai Finalis sekaligus penerima pendanaan Lomba Riset Sawit Mahasiswa yang diselenggarakan oleh BPD-PKS Kementerian Keuangan. Di tahun yang sama, penulis juga berhasil lolos pendanaan Program Peningkatan Kapasitas Organisasi Kemahasiswaan (PPK Ormawa) bersama KOPMA Universitas Lampung dan melaksanakan kegiatan pengabdian di Desa Air Kubang, Tanggamus, Lampung. Selain itu, penulis turut berpartisipasi dalam program Merdeka Belajar Kampus Merdeka (MBKM) melalui Studi Independen di PT Ozamy Inti Sinergi yang berlangsung sejak 16 Februari hingga 30 Juni 2024.

PERSEMBAHAN

Alhamdulillahirabbil'alamin, ku ucapkan Puji Syukur kepada Allah Subhanahu
Wa Ta'ala atas segala rahmat dan hidayah-Nya serta
Sholawat kepada Nabi Muhammad Shallallahu Alaihi Wa Sallam yang selalu
menjadi teladan bagi kehidupanku.

Kupersembahkan dengan tulus Skripsi ini sebagai tanda terimakasih kepada:

Papa dan Mama Tersayang

TONI ANDESTA dan ANA NOPIANAH

Serta Kakek dan Nenek Tersayang

Alm. Muhammad Maulana dan Junainah

Alm. Endang Suhendar dan Almh. Martini

Keluarga besar, dosen, teman, serta almamater yang telah menjadi motivasi dan
inspirasi, terima kasih atas kekeluargaan, do'a, serta dukungan yang selalu
mengiringi hingga merasa lebih baik dari hari ke hari.

Terimakasih untuk segala tulus kasih, semangat, dan dukungan dalam
segala hal selama ini, sehingga penulis dapat menyelesaikan masa kuliah ini.

Dan Terimakasih untuk Diri Sendiri, **Dinda Armeylia Putri**

MOTTO

”Allah memang tidak menjanjikan hidupmu akan selalu mudah, tapi Allah dua kali berjanji bahwa :

فَإِنَّ مَعَ الْعُسْرِ يُسْرًا , إِنَّ مَعَ الْعُسْرِ يُسْرًا

Maka sesungguhnya bersama kesulitan ada kemudahan. Sesungguhnya bersama kesulitan ada kemudahan”

(QS. Al-Insyirah 94;5-6)

“Selalu ada harga dalam sebuah proses. Nikmati saja lelah-lelah itu. Lebarakan lagi rasa sabar itu. Semua yang kau investasikan untuk menjadikan dirimu serupa yang kau impikan, mungkin tidak akan selalu berjalan lancar. Tapi, gelombang-gelombang itu yang nanti bisa kau ceritakan”

(Boy Candra)

“God has perfect timing, never early, never late. It takes a little patience and it takes a lot of faith, but it’s a worth to wait”

“Hidup bukan saling mendahului, bermimpilah sendiri-sendiri”

(Hindia)

“Aku membahayakan nyawa ibuku untuk lahir ke dunia, jadi tidak mungkin aku tidak ada artinya”

SANWACANA

Puji Syukur penulis panjatkan kepada Tuhan Yang Maha Esa berkat Rahmat dan hidayah-Nya penulis dapat menyelesaikan skripsi yang berjudul “ Evaluasi Algoritma *CNN* dan *Faster R-CNN* Untuk Klasifikasi Otomatis Tingkat Kematangan Tandan Buah Segar Kelapa Sawit”. Penulis menyadari bahwa penulisan skripsi ini dapat diselesaikan atas bantuan dari berbagai pihak, oleh karena itu, penulis ingin mengucapkan terimakasih kepada:

1. Allah Subhanahu Wa Ta’ala yang telah memberikan rezeki serta kesehatan sehingga mampu terjalannya kerja praktik hingga menyelesaikan Proposal Tugas Akhir ini.
2. Ibu Prof. Dr. Ir Lusmeilia Afriani, D.E.A., I.P.M. Selaku Rektor Universitas Lampung.
3. Bapak Dr. Eng. Helmy Fitriawan, S.T., M.Sc. Selaku Dekan Fakultas Teknik Universitas Lampung.
4. Ibu Herlinawati, S.T.,M.T. Sebagai Ketua Jurusan Teknik Elektro Universitas Lampung.
5. Bapak Sumadi, S.T., M.T. Sebagai Ketua Program Studi Teknik Elektro Universitas Lampung.
6. Bapak Ing. Ardian Ulvan, S.T., M.Sc. Ph.D. selaku Kepala Laboratorium Telekomunikasi Informasi, sekaligus penguji utama tugas akhir dan telah banyak membantu memberikan kritik dan saran kepada penulis.
7. Ibu Ing. Melvi, S.T., M.T. Ph.D. Sebagai Dosen Pembimbing Utama tugas akhir dan telah banyak membantu dan memberi manfaat terutama dalam hal menulis tugas akhir.
8. Bapak Aryanto, S.T., M.T. Sebagai Dosen Pembimbing Pendamping yang telah banyak membimbing dan memberi dukungan kepada penulis.
9. Bapak Zulmiftah Huda, S.T., M.T. Selaku dosen Pembimbing Akademik.

10. Seluruh Dosen dan karyawan Jurusan Teknik Elektro Universitas Lampung, berkat ilmu yang telah diajarkan kepada penulis selama penulis menjalani masa studi di perkuliahan.
11. Kedua orang tuaku tersayang, Papa Toni Andesta dan Mama Ana Nopianah. Terima kasih penulis ucapkan atas segala pengorbanan dan ketulusan yang diberikan. Meskipun Papa dan Mama tidak sempat merasakan pendidikan dibangku perkuliahan, namun selalu senantiasa memberikan dukungan baik secara moral maupun finansial, serta memprioritaskan Pendidikan dan kebahagiaan anaknya. Perjalanan hidup kita sebagai keluarga memang tidak mudah, tetapi segala hal yang telah dilalui memberikan penulis Pelajaran yang sangat berharga tentang menjadi seorang wanita kuat, bertanggung jawab, selalu berjuang dan mandiri. Semoga dengan adanya skripsi ini dapat membuat Papa dan Mama lebih bangga karena telah berhasil menjadikan anak perempuannya menyandang gelar sarjana seperti yang diharapkan. Besar harapan penulis semoga Papa dan Mama selalu sehat, panjang umur, dan bisa menyaksikan keberhasilan lainnya yang akan penulis raih di masa yang akan depan.
12. Kepada Kakek dan Nenek penulis mengucapkan banyak terima kasih atas limpahan kasih sayang dan sosok yang selalu semangat untuk mendorong pendidikan penulis sejak kecil sampai saat ini.
13. Bella Amelia dan Siti Rahmawati yang senantiasa menemani serta memberikan dukungan kepada penulis selama perkuliahan dan proses penyusunan Tugas Akhir. Kehadiran kalian sebagai rekan sekaligus sahabat telah menjadi tempat berbagi ilmu, saling menguatkan, dan menumbuhkan semangat hingga skripsi ini dapat terselesaikan.
14. Donna Friska Sihombing dan Meily Amara Agieztya, sahabat sejak masa SMA yang selalu hadir menemani perjalanan penulis. Terima kasih telah setia mendengarkan setiap keluh kesah, memberi semangat disaat lelah, dan menjadi tempat berbagi cerita dalam suka maupun duka.
15. Kepada pemilik NPM 2115031001 yang tak kalah penting kehadirannya. Terimakasih telah menjadi bagian dari perjalanan hidup penulis, yang berkontribusi banyak dalam penulisan skripsi ini, baik tenaga maupun waktu

kepada penulis. Terima kasih telah mendukung, menghibur, mendengarkan keluh kesah dan memberikan semangat untuk pantang menyerah.

16. Semua pihak yang telah membantu penulis dalam menyelesaikan skripsi ini yang tidak dapat penulis sebutkan satu persatu.

Penulis menyadari bahwa masih banyak kekurangan dalam penulisan skripsi ini. Penulis mengharapkan adanya kritik dan saran yang membangun dari berbagai pihak demi kemajuan bersama. Terima kasih.

Bandar Lampung, 25 Agustus 2025

Penulis

Dinda Armeylia Putri

2115031109

DAFTAR ISI

Halaman

ABSTRAK	iii
ABSTRACT	iv
HALAMAN PERSETUJUAN	v
HALAMAN MENGESAHKAN	vi
SURAT PERNYATAAN	vii
RIWAYAT HIDUP.....	viii
PERSEMBAHAN.....	ix
MOTTO	x
SANWACANA.....	xi
DAFTAR ISI	xiv
DAFTAR GAMBAR	xviii
DAFTAR TABEL	xxi
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	3
1.3 Tujuan Penelitian.....	3
1.4 Batasan Masalah.....	4
1.5 Manfaat Penelitian.....	4
1.6 Sistematika Penulisan.....	5
BAB II TINJAUAN PUSTAKA	6
2.1 Penelitian Terdahulu	6

2.2 Kelapa Sawit.....	8
2.3 Kematangan Buah Kelapa Sawit.....	9
2.3.1 Fraksi Matang Panen	9
2.4 <i>Deep Learning</i>	10
2.5 <i>Computer Vision</i>	12
2.6 <i>Citra Digital</i>	13
2.7 Pengolaan Citra Digital	16
2.6.1 <i>Resizing</i>	16
2.6.2 <i>Augmentasi</i>	17
2.8 <i>Convolutional Neural Network (CNN)</i>	18
2.8.1 <i>Convolutional Layer</i>	19
2.8.2 <i>Pooling Layer</i>	22
2.8.3 <i>Fully Connected Layer</i>	23
2.8.4 <i>Activation Function</i>	23
2.9 <i>Region Convolutional Neural Network (R-CNN)</i>	25
2.10 <i>Fast Region Convolutional Neural Network (Fast R-CNN)</i>	25
2.11 <i>Faster Region Convolutional Neural Network (Faster R-CNN)</i>	26
2.12 <i>Resnet-50</i>	28
2.13 <i>Google Colaboratory</i>	30
2.14 <i>Python</i>	30
2.15 <i>TensorFlow</i>	31
2.17 <i>Optimizer Adam</i>	32
2.18 Siklus proyek <i>AI</i>	33
2.18.1 <i>Problem Scoping</i>	33
2.18.2 <i>Data Acquisition</i>	33
2.18.3 <i>Data Exploration</i>	34

2.18.5 <i>Evaluation</i>	34
2.19 Confussion Matrix	35
2.19.1 <i>Accuracy</i>	36
2.19.2 <i>Precision</i>	36
2.19.3 <i>Recall</i>	37
2.19.4 <i>F1-Score</i>	37
2.19.5 <i>Mean Average Precision</i>	37
BAB III METODOLOGI PENELITIAN	39
3.1 Waktu dan Tempat Penelitian	39
3.2 Alat dan Bahan Penelitian	39
3.3 Diagram Alir Penelitian	39
3.4 Prosedur Penelitian	41
3.4.1 Studi literatur	41
3.4.2 Pengumpulan data	41
3.5 Perencanaan Program	42
3.6 <i>Data Acquisition</i>	43
3.7 <i>Data Exploration</i>	45
3.6 <i>Modelling</i>	47
3.8 Evaluation	48
BAB IV. HASIL DAN PEMBAHASAN	49
4.1 <i>Data Acquisition</i>	49
4.2 <i>Data Exploration</i>	51
4.3 Proses <i>Modelling, Training</i> dan <i>Evaluation</i>	53
4.3.1 Model <i>CNN</i>	55
4.3.2 Model <i>Faster R-CNN RESNET50</i>	81
4.4 Hasil Evaluasi Kinerja Model	92

BAB V PENUTUP	96
5.1 KESIMPULAN	96
5.2 SARAN	97
DAFTAR PUSTAKA.....	98
LAMPIRAN.....	103

DAFTAR GAMBAR

Gambar 2. 1 Perbandingan Citra Sawit Matang, Mengkal, dan Mentah	10
Gambar 2. 2 <i>Machine Learning</i> dan <i>Deep Learning</i>	11
Gambar 2. 3 Warna RGB	15
Gambar 2. 4 Proses Perhitungan <i>Resizing</i>	17
Gambar 2. 5 Nilai matriks setelah <i>resizing</i>	17
Gambar 2. 6 Arsitektur <i>CNN</i>	19
Gambar 2. 7 Layer Input	20
Gambar 2. 8 <i>Convolution Layer</i>	21
Gambar 2. 9 Aktivasi <i>ReLU</i>	22
Gambar 2. 10 <i>Max Pooling</i>	23
Gambar 2. 11 <i>Fully Connected Layer</i>	24
Gambar 2. 12 Arsitektur <i>R-CNN</i>	25
Gambar 2. 13 Arsitektur <i>Fast R-CNN</i>	26
Gambar 2. 14 Arsitektur Umum <i>Faster R-CNN</i>	26
Gambar 2. 15 Arsitektur <i>Region Proposal Network (RPN)</i>	27
Gambar 2. 16 Arsitektur <i>ResNet50</i>	29
Gambar 2. 17 <i>Google Colaboratory</i>	30
Gambar 2. 18 Logo <i>Software Python</i>	31
Gambar 2. 19 Logo <i>TensorFlow</i>	31
Gambar 2. 20 Siklus Proyek <i>AI</i>	33
Gambar 2. 21 Tabel <i>Confusion Matrix</i>	35
Gambar 3. 1 Diagram Alir Penelitian.....	40
Gambar 3. 2 Data <i>Roboflow 8400</i>	43
Gambar 3. 3 Data <i>Roboflow 625</i>	44
Gambar 3. 4 Data Perkebunan	44
Gambar 3. 5 Diagram Alir Data <i>Exploration</i>	45
Gambar 3. 6 Diagram Alir <i>Modelling</i>	47

Gambar 4. 1 Rincian Dataset <i>Roboflow</i> 8400	49
Gambar 4. 2 Rincian Dataset <i>Roboflow</i> 625	50
Gambar 4. 3 Dataset Perkebunan	51
Gambar 4. 4 Proses Anotasi Buah Sawit di <i>Roboflow</i>	52
Gambar 4. 5 Rincian Dataset Perkebunan	53
Gambar 4. 6 Pengecekan Runtime GPU T4.....	54
Gambar 4. 7 Import Dataset Perkebunan dari Zip	55
Gambar 4. 8 Import Dataset <i>Roboflow</i> 8400 dari Zip.....	56
Gambar 4. 9 Import Dataset <i>Roboflow</i> 625 dari Zip.....	56
Gambar 4. 10 Kode Program Generator Data COCO untuk Membaca dan Menyiapkan Gambar serta Anotasi <i>CNN</i>	59
Gambar 4. 11 Visualisasi Data Batch <i>CNN</i>	61
Gambar 4. 12 Hasil <i>Bounding Box CNN</i> Dataset Perkebunan.....	61
Gambar 4. 13 Hasil <i>Bounding Box CNN</i> Dataset <i>Roboflow</i> 8400	62
Gambar 4. 14 Hasil <i>Bounding Box CNN</i> Dataset <i>Roboflow</i> 625	62
Gambar 4. 15 Kode <i>crop_and_save_images</i> untuk memotong objek dari gambar dan menyimpannya ke folder per kelas <i>CNN</i>	63
Gambar 4. 16 Kode Pembangunan dan Kompilasi Model <i>CNN</i> untuk Klasifikasi	65
Gambar 4. 17 Kode Evaluasi Model <i>CNN</i>	67
Gambar 4. 18 Kode Evaluasi Performa Model <i>CNN</i>	69
Gambar 4. 19 Kode Instalasi Pustaka	70
Gambar 4. 20 Kode Mengonversi Dataset Berformat COCO.....	71
Gambar 4. 21 Kode Pembuatan Custom Data Generator untuk Deteksi Objek ...	73
Gambar 4. 22 Kode Arsitektur <i>Backbone Resnet50 CNN</i>	74
Gambar 4. 23 Kode Pelatihan dan Penyimpanan Model <i>CNN ResNet50</i>	75
Gambar 4. 24 Visualisasi Hasil Prediksi Model Terhadap Citra Validas	77
Gambar 4. 25 Hasil <i>Bounding Box CNN Resnet50</i> Data Perkebunan.....	77
Gambar 4. 26 Hasil <i>Bounding Box CNN Resnet50 Roboflow</i> 8400.....	78
Gambar 4. 27 Hasil <i>Bounding Box CNN Resnet50 Roboflow</i> 625.....	78
Gambar 4. 28 Visualisasi Performa Model <i>CNN RESNET50</i>	81

Gambar 4. 29 Impor dan Inisialisasi Pustaka <i>Python</i> untuk Pelatihan dan Evaluasi Model <i>Faster R-CNN</i>	82
Gambar 4. 30 Kode Implementasi Kelas Dataset COCO untuk Tandan Buah Segar Kelapa Sawit	84
Gambar 4. 31 Kode Fungsi Evaluasi Kinerja Model Deteksi Objek <i>Faster-RCNN</i>	88
Gambar 4. 32 Kode Pelatihan, Evaluasi, dan Visualisasi Model <i>Faster R-CNN</i> ..	90
Gambar 4. 33 Hasil <i>Bounding Box Faster RCNN Resnet50</i> Data Perkebunan.....	91
Gambar 4. 34 Hasil <i>Bounding Box Faster RCNN Resnet50</i> Data <i>Roboflow</i> 8400.	91
Gambar 4. 35 Hasil <i>Bounding Box Faster RCNN Resnet50</i> Data <i>Roboflow</i> 625 .	91

DAFTAR TABEL

Tabel 3. 1 Data Citra Tandan Buah Segar Kelapa Sawit.....	42
Tabel 4. 1 Evaluasi Kinerja Model Menggunakan Dataset Perkebunan	92
Tabel 4. 2 Evaluasi Kinerja Model Menggunakan Dataset <i>Roboflow</i> 8400	93
Tabel 4. 3 Evaluasi Kinerja Model Menggunakan Dataset <i>Roboflow</i> 6s25	94

BAB I PENDAHULUAN

1.1 Latar Belakang

Indonesia merupakan salah satu produsen minyak kelapa sawit terbesar di dunia. CPO (Crude Palm Oil) merupakan salah satu bahan baku yang memiliki tingkat daya guna yang tinggi baik dalam kebutuhan sehari-hari maupun industri. Berdasarkan data dari *United States Department of Agriculture* (USDA), Indonesia adalah produsen minyak sawit terbesar di dunia. USDA memproyeksikan produksi CPO di Indonesia mencapai 45,5 juta metrik ton (MT) pada periode 2022/2023 [1]. Namun, permasalahan yang sering muncul pada industri kelapa sawit adalah menurunnya kualitas CPO pada minyak kelapa sawit. Penyebab utama penurunan kualitas CPO Indonesia adalah kandungan asam lemak bebas (ALB) yang tinggi. Hal ini disebabkan buah kelapa sawit harus dipanen tepat waktu, jika terlalu matang maka minyak yang dihasilkan mengandung ALB dalam jumlah tinggi (lebih dari 5%) sedangkan jika dipanen dalam keadaan belum matang maka tingkat ALB dan rendemen minyak yang dihasilkan akan rendah. Penentuan tingkat kematangan buah kelapa sawit secara umum ditentukan berdasarkan jumlah brondolan dan warna. Warna menjadi panduan penting untuk menyatakan kandungan minyak, sehingga penanganan panen buah kelapa sawit menjadi suatu kegiatan yang penting dalam meningkatkan mutu CPO [2].

Buah kelapa sawit yang matang menghasilkan rendemen CPO yang lebih tinggi dibandingkan dengan buah yang masih mentah atau setengah matang. Buah yang masih mentah umumnya berwarna hitam pada seluruh permukaannya, sementara buah yang cukup matang menunjukkan warna hitam kemerahan, dan buah yang matang sempurna akan tampak merah jingga. Di perkebunan kelapa sawit, proses panen masih banyak mengandalkan cara tradisional, yang sering dianggap kurang efisien karena bersifat subjektif, lambat, dan tidak selalu akurat. Proses identifikasi manual memerlukan waktu yang lama, dan ketelitian manusia sering kali

memungkinkan terjadinya kesalahan dalam menentukan kematangan buah secara tepat. Untuk menjaga kualitas CPO dan meningkatkan hasil perkebunan, diperlukan metode yang lebih efisien untuk mengidentifikasi buah kelapa sawit dengan tingkat kematangan optimal.

Beberapa penelitian sebelumnya telah menggunakan citra buah untuk identifikasi dan klasifikasi tingkat kematangan. Salah satunya adalah penelitian berjudul “Klasifikasi Tingkat Kematangan Tandan Buah Segar Kelapa Sawit Menggunakan Pendekatan *Deep Learning*” oleh [3], yang membahas penggunaan arsitektur *ResNet50* dalam mengklasifikasikan tingkat kematangan tandan buah segar (TBS) kelapa sawit. Penelitian ini bertujuan mengatasi kesalahan yang kerap terjadi dalam klasifikasi manual, yang rentan terhadap ketidakakuratan dan memakan waktu lama. Klasifikasi TBS dalam penelitian ini dibagi menjadi empat kategori kematangan: mentah, mengkal dan matang. Pengujian dilakukan menggunakan berbagai skenario, seperti variasi alokasi data, pilihan *optimizer* (Adam dan SGD), serta beberapa nilai *learning rate*. Hasil terbaik diperoleh dengan konfigurasi alokasi data 90:10, *optimizer* Adam, dan *learning rate* 0,0001, dengan akurasi mencapai 97%. Penelitian ini menunjukkan bahwa model *ResNet50* memiliki akurasi tinggi dalam klasifikasi kematangan TBS, tetapi membutuhkan data latihan yang besar dan teknik augmentasi data untuk menghindari *overfitting*.

Penelitian dengan menggunakan metode *Faster R-CNN* berjudul “Klasifikasi Mutu Buah Pala (*Myristica Fragrans Houtt*) Berbasis Pengolahan Citra Menggunakan Metode *Deep Learning* Arsitektur *Faster R-CNN*” yang dilakukan oleh [4] bertujuan untuk meningkatkan efisiensi dan akurasi dalam proses sortasi buah pala yang sebelumnya dilakukan secara manual. Penelitian ini berangkat dari tantangan yang dihadapi dalam memenuhi permintaan ekspor buah pala akibat kualitas yang tidak konsisten karena metode sortasi manual. Sortasi secara manual dinilai kurang efektif karena bergantung pada tenaga manusia yang sering kali menghasilkan ketidakkonsistenan, sehingga mempengaruhi nilai jual buah. Dengan menggunakan 1000 iterasi, metode *Faster R-CNN* dalam penelitian ini berhasil mencapai akurasi hingga 95%, lebih tinggi dibandingkan metode CNN konvensional yang hanya mencapai akurasi 87%. Selain itu, *Faster R-CNN* mampu mempercepat proses

pemrosesan secara signifikan, dengan waktu pemrosesan hanya 0,024 detik per gambar.

Untuk mengatasi tantangan ini, teknologi *deep learning* menawarkan solusi dalam mengotomatisasi klasifikasi kematangan buah dengan akurasi tinggi. Algoritma *Convolutional Neural Network (CNN)* dan *Faster R-CNN* merupakan dua metode populer yang mampu melakukan klasifikasi gambar secara efisien. Penelitian ini bertujuan untuk membandingkan kinerja kedua algoritma tersebut dalam mengklasifikasikan tingkat kematangan tandan buah segar kelapa sawit, guna menentukan algoritma yang paling efektif dalam memberikan akurasi pemrosesan terbaik.

1.2 Rumusan Masalah

Adapun rumusan masalah dari penelitian ini sebagai berikut:

1. Bagaimana cara mengklasifikasikan tingkat kematangan tandan buah segar kelapa sawit menjadi matang, mengkal dan mentah menggunakan algoritma *CNN* dan *Faster R-CNN*?
2. Bagaimana perbandingan kinerja algoritma *CNN* dan *Faster R-CNN* dalam mengklasifikasikan tingkat kematangan tandan buah segar kelapa sawit dengan menggunakan dua sumber data yang berbeda?
3. Bagaimana akurasi dan presisi dari kedua algoritma dalam mengklasifikasi kematangan tandan buah segar kelapa sawit?

1.3 Tujuan Penelitian

Adapun tujuan dari penelitian ini sebagai berikut:

1. Mengklasifikasikan tingkat kematangan tandan buah segar kelapa sawit menjadi matang, mengkal dan mentah menggunakan algoritma *CNN* dan *Faster R-CNN*.
2. Membandingkan kinerja algoritma *CNN* dan *Faster R-CNN* dalam mengklasifikasikan tingkat kematangan tandan buah segar kelapa sawit dari dua sumber data yang berbeda.
3. Menentukan algoritma dengan kinerja terbaik untuk mengklasifikasi tingkat kematangan tandan buah segar kelapa sawit.

1.4 Batasan Masalah

Adapun batasan masalah pada penelitian ini sebagai berikut:

1. Data yang digunakan berekstensi .jpg/.jpeg .
2. Klasifikasi dilakukan berdasarkan warna dari citra buah kelapa sawit.
3. Output klasifikasi berupa tingkat kematangan yaitu mentah, mengkal, dan matang.
4. Output penelitian berupa kedua model untuk didapatkan model terbaik.

1.5 Manfaat Penelitian

Adapun manfaat untuk petani dan industri kelapa sawit didalam penelitian ini adalah sebagai berikut:

1. Meningkatkan kualitas CPO yang dihasilkan dari buah kelapa sawit melalui identifikasi tingkat kematangan yang akurat.
2. Mengurangi kesalahan dalam penentuan tingkat kematangan, sehingga meningkatkan konsistensi dan nilai jual produk.
3. Memberikan informasi dan teknologi yang dapat digunakan oleh petani untuk meningkatkan hasil panen dan pendapatan.
4. Otomasi klasifikasi kematangan tandan buah segar kelapa sawit untuk meningkatkan efisiensi industri kelapa sawit.

Adapun manfaat untuk akademis dan perkembangan teknologi didalam penelitian ini adalah sebagai berikut:

1. Memberikan kontribusi pada pengembangan teknologi *deep learning*, khususnya pada penerapan algoritma *CNN* dan *Faster R-CNN* untuk klasifikasi tingkat kematangan tandan buah segar kelapa sawit.
2. Menyediakan referensi bagi penelitian selanjutnya dalam mengembangkan metode otomatisasi klasifikasi berbasis citra digital di sektor pertanian.
3. Menjadi bahan pembelajaran bagi akademisi dan praktisi dalam memahami kelebihan dan kekurangan algoritma *CNN* dan *Faster R-CNN* untuk pengolahan citra.

4. Membantu memperluas wawasan dalam penggunaan teknologi *AI* untuk menyelesaikan masalah nyata di industri, khususnya yang memerlukan akurasi tinggi dan efisiensi waktu.

1.6 Sistematika Penulisan

Untuk memudahkan penulisan dan pemahaman mengenai materi dari skripsi ini, maka skripsi ini dibagi menjadi 5 bab yang terdiri dari:

BAB I. PENDAHULUAN

Berisikan latar belakang, tujuan penelitian, manfaat penelitian, perumusan masalah, batasan masalah, dan sistematika penulisan.

BAB II. TINJAUAN PUSTAKA

Berisikan teori-teori yang digunakan dalam melakukan penelitian.

BAB III. METODE PENELITIAN

Bab ini berisikan tentang hal-hal yang dilakukan dalam penelitian, seperti; waktu dan tempat penelitian, metode dan proses pembuatan program.

BAB IV. HASIL DAN PEMBAHASAN

Berisikan hasil dari penelitian dan pembahasan dan perhitungan yang dilakukan pada penelitian ini.

BAB V. KESIMPULAN DAN SARAN

Berisikan kesimpulan dari penelitian dan saran untuk penelitian selanjutnya

DAFTAR PUSTAKA

Berisikan daftar pustaka yang digunakan pada penulisan skripsi

LAMPIRAN

Berisikan lampiran dan hasil pemantauan dan program.

BAB II TINJAUAN PUSTAKA

2.1 Penelitian Terdahulu

Penelitian [3] dengan judul “Klasifikasi Tingkat Kematangan Tandan Buah Segar Kelapa Sawit Menggunakan Pendekatan *Deep Learning*,” dimana mengeksplorasi penggunaan arsitektur *ResNet50* untuk mengklasifikasikan tingkat kematangan tandan buah segar (TBS) kelapa sawit. Studi ini bertujuan mengatasi kekeliruan dalam klasifikasi manual yang cenderung tidak akurat dan memakan waktu lama. Kategori kematangan TBS dalam penelitian ini terdiri dari empat tingkat: mentah, kurang matang, matang, dan terlalu matang. Pengujian dilakukan dengan berbagai skenario, termasuk variasi alokasi data, pilihan *optimizer* (Adam dan SGD), serta beberapa nilai *learning rate*. Hasil terbaik diperoleh dengan konfigurasi alokasi data 90:10, *optimizer* Adam, dan learning rate 0,0001, mencapai akurasi sebesar 97%. Penelitian ini membuktikan bahwa model *ResNet50* memiliki akurasi tinggi dalam klasifikasi kematangan TBS, tetapi memerlukan data latih yang besar dan teknik augmentasi untuk mencegah *overfitting*.

Penelitian [4] berjudul “Klasifikasi Mutu Buah Pala (*Myristica Fragrans Houtt*) Berbasis Pengolahan Citra Menggunakan Metode *Deep Learning* Arsitektur *Faster R-CNN*” bertujuan meningkatkan efisiensi dan akurasi dalam proses sortasi buah pala, yang sebelumnya dilakukan secara manual. Tantangan utama dalam memenuhi permintaan ekspor adalah kualitas buah yang tidak konsisten akibat proses sortasi manual yang bergantung pada tenaga manusia, sering menghasilkan ketidak konsistenan dan mempengaruhi nilai jual. Dalam penelitian ini, *Faster R CNN* diuji dengan 1000 iterasi dan berhasil mencapai akurasi hingga 95%, mengungguli metode *CNN* konvensional yang hanya mencapai 87%. *Faster R CNN* juga mampu mempercepat waktu pemrosesan menjadi hanya 0,024 detik per gambar. Untuk menjawab tantangan ini, deep learning menawarkan solusi otomatisasi dalam klasifikasi kematangan buah dengan akurasi tinggi. Algoritma *Convolutional*

Neural Network (CNN) dan *Faster R-CNN* merupakan metode populer yang efisien dalam klasifikasi gambar. Penelitian ini bertujuan membandingkan kinerja kedua algoritma dalam mengklasifikasikan tingkat kematangan TBS kelapa sawit, untuk menentukan algoritma yang paling efektif dalam hal akurasi dan kecepatan pemrosesan.

Penelitian [5] berjudul “*A Comparative Study of Different Architectural Models of CNN for Plant Leaf Disease Detection*” membahas perbandingan tiga model *Convolutional Neural Network (CNN)*, yaitu *AlexNet*, *VGG16Net*, dan *ResNet*, dalam mendeteksi penyakit pada daun tanaman. Penelitian ini menggunakan dataset *PlantVillage* yang berisi 54.306 gambar daun sehat dan berpenyakit dari 38 kelas. Setiap model diuji dengan 50 dan 100 epoch untuk menilai akurasi pelatihan, akurasi validasi, serta kehilangan pelatihan dan validasi. Hasil penelitian menunjukkan bahwa *ResNet* unggul dibandingkan *AlexNet* dan *VGG16Net*, dengan akurasi validasi tertinggi sebesar 96% pada 100 epoch. Selain itu, *ResNet* juga memiliki nilai kehilangan pelatihan dan validasi yang lebih rendah, yang menunjukkan bahwa model ini mampu mengenali pola penyakit dengan lebih akurat. *AlexNet* dan *VGG16Net* juga menunjukkan hasil yang cukup baik, tetapi tidak seoptimal *ResNet* dalam mendeteksi penyakit tanaman.

Penelitian [6] berjudul “*Klasifikasi Buah Segar Dan Busuk Menggunakan Algoritma Convolutional Neural Network (CNN)*” membahas tentang pengembangan sistem klasifikasi otomatis untuk buah segar dan busuk menggunakan algoritma *Convolutional Neural Network (CNN)*. Bertujuan untuk mengatasi keterbatasan metode inspeksi manual yang lambat, tidak efisien, dan rentan terhadap kesalahan. Dengan menggunakan dataset berisi 13.619 gambar buah dalam berbagai kondisi, peneliti melakukan *pre-processing* seperti normalisasi dan augmentasi data, serta merancang arsitektur *CNN* dengan lapisan *Conv2D* dan *MaxPooling2D* untuk ekstraksi fitur dan pengurangan dimensi data. Hasil penelitian menunjukkan bahwa model *CNN* mampu mencapai akurasi hingga 96,67% pada data uji, yang membuktikan bahwa metode ini efektif dalam mengidentifikasi kondisi buah secara konsisten. Akurasi yang tinggi ini menunjukkan potensi *CNN* dalam mengotomatisasi klasifikasi kesegaran buah,

yang diharapkan dapat meningkatkan kualitas produk dan efisiensi dalam industri pertanian.

Penelitian [7] berjudul "*Faster R-CNN for Multi-class Fruit Detection using a Robotic Vision System*", yang bertujuan untuk mengembangkan sistem deteksi buah secara otomatis menggunakan teknologi *deep learning*. Dalam penelitian ini, dikembangkan kerangka kerja berbasis *Faster R-CNN* untuk mendeteksi berbagai jenis buah dalam lingkungan perkebunan secara real-time. Penelitian ini mencakup beberapa tahap utama, yaitu pembuatan pustaka citra buah, augmentasi data, pembentukan model *Faster R-CNN* yang ditingkatkan, serta evaluasi kinerja model. Hasil penelitian menunjukkan bahwa model yang diusulkan mampu meningkatkan akurasi deteksi dan mengurangi waktu pemrosesan dibandingkan dengan metode deteksi konvensional. Model yang dikembangkan menggunakan optimalisasi pada lapisan konvolusi dan pooling untuk meningkatkan efisiensi dan presisi deteksi. Dengan menganalisis lebih dari 4.000 gambar buah dari lingkungan nyata, sistem ini berhasil mencapai akurasi tinggi dalam mendeteksi berbagai jenis buah dengan lebih cepat dan andal. Penggunaan *Faster R-CNN* dalam deteksi buah multi-kelas ini dapat diterapkan dalam sistem pertanian cerdas untuk meningkatkan efisiensi, efektivitas, serta keandalan dalam mendeteksi dan mengelompokkan buah berdasarkan jenisnya di lingkungan Perkebunan.

2.2 Kelapa Sawit

Kelapa sawit merupakan tanaman dari genus *Elaeis* yang terdiri atas dua spesies utama, yaitu *Elaeis melanococca* atau *Elaeis oleivera* dari Amerika Selatan dan *Elaeis guineensis* dari Afrika. Meskipun berasal dari luar Asia, tanaman ini mampu tumbuh subur di daerah tropis seperti Asia Tenggara, termasuk Indonesia, Malaysia, Papua Nugini, dan Thailand. Di negara-negara tropis ini, kelapa sawit bahkan menunjukkan produktivitas yang lebih tinggi dibandingkan daerah asalnya, sehingga menjadi komoditas unggulan perkebunan. Kelapa sawit di Indonesia pertama kali diperkenalkan pada tahun 1848 oleh pemerintah Kolonial Belanda. Bibit tanaman ini dibawa dari Mauritius dan Amsterdam, kemudian ditanam di Kebun Raya Bogor. Sejak itu, kelapa sawit berkembang pesat dan menjadi salah satu tanaman perkebunan yang penting bagi perekonomian Indonesia [8] [9].

Pohon kelapa sawit mulai menghasilkan buah saat berusia sekitar tiga tahun dan memiliki masa produktif rata-rata hingga 25 tahun. Buah kelapa sawit dapat dipanen sepanjang tahun, yaitu selama 12 bulan. Tanaman ini merupakan jenis pohon dengan tinggi mencapai 0-24 meter. Bunga dan buahnya tumbuh dalam bentuk tandan yang bercabang banyak. Buah kelapa sawit berukuran kecil dan akan berubah menjadi merah kehitaman saat matang. Bagian daging dan kulit buahnya mengandung minyak yang digunakan dalam berbagai produk seperti minyak goreng, pembersih, dan lilin [10]. Semakin matang buahnya, semakin banyak minyak yang dapat diperoleh dari proses ekstraksi daging buah. Namun, jika buah dibiarkan terlalu matang, kadar asam lemak bebas (ALB) akan meningkat, biasanya melebihi 5%. Oleh karena itu, panen kelapa sawit harus dilakukan tepat waktu untuk menghindari tingginya kandungan ALB dalam minyak yang dihasilkan.

2.3 Kematangan Buah Kelapa Sawit

Panen merupakan faktor krusial dalam budidaya kelapa sawit karena keberhasilannya berpengaruh langsung pada produktivitas tanaman. Buah kelapa sawit mencapai kematangan sekitar enam bulan setelah proses penyerbukan dan pembuahan. Penentuan kriteria kematangan panen biasanya didasarkan pada perubahan warna buah. Panen dilakukan pada tingkat kematangan optimal, yaitu ketika tandan buah segar (TBS) memiliki kandungan minyak dan kernel tertinggi [11]. Mengetahui tingkat kematangan buah kelapa sawit dapat dilihat melalui fraksi matang panen berikut.

2.3.1 Fraksi Matang Panen

Fraksi panen pada kelapa sawit sangat berpengaruh terhadap rendemen minyak dan kadar asam lemak bebas (ALB). Semakin tinggi fraksi panen, yang menunjukkan tingkat kematangan buah, maka rendemen minyak yang dihasilkan juga akan meningkat [12]. Hal ini disebabkan oleh peningkatan kandungan minyak dalam buah seiring dengan kematangan. Buah kelapa sawit yang matang sempurna memiliki potensi untuk menghasilkan minyak yang lebih banyak dibandingkan dengan buah yang belum matang [12].

Perbedaan tingkat kematangan buah kelapa sawit dapat diamati melalui citra yang ditampilkan



Gambar 2. 1 Perbandingan Citra Sawit Matang, Mengkal, dan Mentah

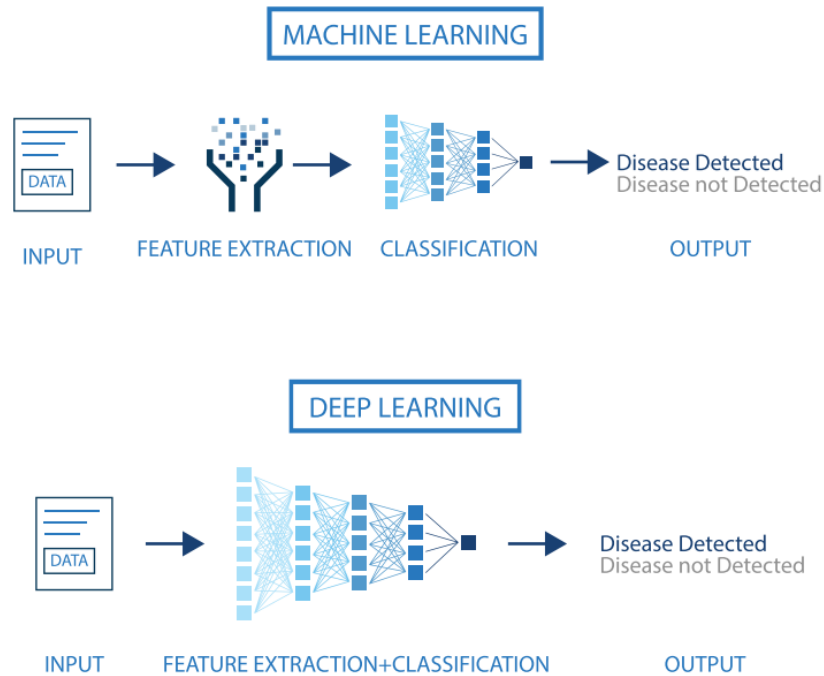
Gambar 2.1 menunjukkan perbandingan citra sawit matang, mengkal, dan mentah dimana terlihat perbedaan warna dari ketiga kategori yang ditetapkan.

2.4 Deep Learning

Deep learning adalah cabang dari machine learning dalam bidang *Artificial Intelligence (AI)* yang memanfaatkan jaringan saraf tiruan atau *Artificial Neural Networks (ANN)* untuk menganalisis dan mempelajari pola dari data. Algoritma ini terinspirasi dari struktur otak manusia dan memungkinkan mesin untuk mengenali pola dari data yang tidak terstruktur, seperti gambar, teks, audio, dan video. Dengan menggunakan beberapa lapisan pemrosesan nonlinier yang bekerja secara paralel, *deep learning* dapat secara otomatis mengekstraksi dan mempelajari fitur langsung dari data tanpa perlu ekstraksi fitur manual seperti pada metode machine learning klasik .

Beberapa arsitektur deep learning yang telah banyak diterapkan, antara lain *Convolutional Neural Network (CNN)*, *Deep Neural Network (DNN)*, *Recurrent Neural Network (RNN)*, *Deep Belief Network (DBN)*, dan *Fully Convolutional Network (FCN)*. Model-model ini telah menunjukkan performa unggul dalam berbagai bidang, termasuk klasifikasi objek, pengenalan wajah, pemrosesan bahasa alami, serta sektor pertanian [13]. Dalam bidang pertanian, *deep learning* memungkinkan otomatisasi klasifikasi dan deteksi objek dengan akurasi tinggi, menggantikan metode konvensional yang memerlukan ekstraksi fitur manual. Teknologi ini berpotensi meningkatkan efisiensi dan akurasi dalam berbagai aplikasi pertanian, seperti identifikasi tingkat kematangan buah dan deteksi penyakit tanaman, sehingga berkontribusi pada peningkatan produktivitas dan keberlanjutan pertanian [14]. Berikut merupakan gambar mengilustrasikan konsep

machine learning dan *deep learning*. Setelah model dianggap akurat, model tersebut akan digunakan untuk melakukan klasifikasi.



Gambar 2. 2 *Machine Learning* dan *Deep Learning* [15]

Gambar 2.2 menunjukkan perbedaan arsitektur dari *machine learning* dan *deep learning*. Perbedaan utama antara *Machine Learning* (ML) dan *Deep Learning* (DL) terletak pada proses ekstraksi fitur. Dalam ML, ekstraksi fitur harus dilakukan secara manual sebelum proses klasifikasi, yang membutuhkan waktu dan tenaga yang cukup besar. Sementara itu, DL dapat secara otomatis mengekstrak fitur langsung dari data menggunakan jaringan saraf tiruan yang dalam (*deep neural networks*), sehingga meningkatkan efisiensi dan akurasi, terutama dalam menangani data yang kompleks. Meskipun demikian, DL memerlukan jumlah data yang lebih besar serta daya komputasi yang lebih tinggi dibandingkan ML.

Dalam penelitian ini, algoritma *deep learning* yang digunakan adalah *Faster R-CNN* dengan *ResNet* sebagai *backbone* serta *Convolutional Neural Network* (CNN) untuk meningkatkan akurasi dalam klasifikasi tingkat kematangan buah kelapa sawit berdasarkan citra. *Faster R-CNN* memiliki keunggulan dalam deteksi objek dengan kecepatan dan ketepatan tinggi, sementara CNN mampu mengenali pola

serta mengekstrak fitur secara otomatis melalui lapisan konvolusi dan *pooling*. Kombinasi kedua metode ini memungkinkan sistem untuk tidak hanya mendeteksi buah kelapa sawit dalam citra tetapi juga mengklasifikasikan tingkat kematangannya dengan lebih akurat dan efisien dibandingkan metode konvensional.

2.5 Computer Vision

Computer Vision (CV) merupakan cabang dari *Artificial Intelligence (AI)* yang dirancang untuk memungkinkan komputer menganalisis citra dan video guna mengekstrak informasi serta memahami objek dengan cara yang mendekati atau bahkan melampaui kemampuan penglihatan manusia [16]. *CV* berfokus pada pemrosesan otomatis citra digital menggunakan kombinasi algoritma pengolahan citra dan Teknik kecerdasan buatan untuk mengidentifikasi pola, mendeteksi objek, serta mengekstraksi fitur penting. Melalui proses ini, sistem *computer vision* dapat menghasilkan informasi yang akurat dan bermanfaat untuk berbagai aplikasi, termasuk dalam bidang pertanian, medis, dan industry.

Pada *Computer Vision (CV)*, terdapat beberapa aspek penting yang menentukan efektivitas dan akurasi sistem dalam menganalisis citra dan video. Beberapa aspek utama dalam *CV* meliputi:

1. Akuisisi Citra (*Image Acquisition*) : Proses pengambilan citra menggunakan perangkat seperti kamera, sensor, atau drone. Kualitas citra yang diperoleh sangat berpengaruh terhadap hasil analisis.
2. Pra-pemrosesan Citra (*Image Preprocessing*) : Tahap ini mencakup Teknik seperti peningkatan kualitas citra (*image enhancement*), normalisasi, pengurangan noise, dan segmentasi untuk mempersiapkan citra sebelum dianalisis lebih lanjut.
3. Ekstraksi Fitur (*Feature Extraction*) : Proses identifikasi dan pemilihan karakteristik penting dalam citra, seperti bentuk, warna, tekstur, dan pola, yang akan digunakan dalam proses klasifikasi atau deteksi objek.
4. Pengenalan dan Klasifikasi Objek (*Object Recognition and Classification*) : Penggunaan algoritma seperti *Convolutional Neural Networks (CNN)* dan

Faster R-CNN untuk mengidentifikasi serta mengkategorikan objek dalam citra berdasarkan fitur yang telah diekstraksi.

5. Segmentasi Citra (*Image Segmentation*) : Pemisahan bagian-bagian penting dalam citra untuk memudahkan analisis, seperti memisahkan objek utama dari latar belakang atau membedakan tingkat kematangan buah kelapa sawit dalam penelitian pertanian.
6. Pelacakan Objek (*Object Tracking*) : Proses memantau pergerakan objek dalam video atau citra berurutan untuk berbagai keperluan, seperti analisis pertumbuhan tanaman atau pemantauan proses panen.
7. Interpretasi dan Pengambilan Keputusan (*Interpretation and Decision Making*) : Tahap akhir di mana sistem *CV* menginterpretasikan hasil analisis citra dan memberikan keluaran yang dapat digunakan dalam pengambilan keputusan, misalnya dalam menentukan waktu panen optimal berdasarkan klasifikasi tingkat kematangan buah.

Aspek-aspek ini saling berhubungan dan membentuk sistem *computer vision* yang efisien dalam berbagai aplikasi, termasuk klasifikasi tingkat kematangan buah kelapa sawit untuk mendukung optimalisasi hasil panen.

2.6 Citra Digital

Citra merupakan representasi visual dari suatu objek yang dapat berbentuk foto optik, sinyal video analog seperti tampilan pada layar televisi, atau format digital yang tersimpan sebagai hasil sistem perekaman data. Dalam dunia digital, citra diolah menggunakan komputer agar dapat dianalisis dan diproses secara digital agar sesuai dengan kebutuhan. Secara matematis, citra adalah fungsi kontinu yang merepresentasikan intensitas Cahaya pada bidang dua dimensi. Namun, agar dapat diproses secara digital, citra harus dikonversi ke dalam bentuk numerik dengan nilai diskrit melalui proses digitalisasi [17].

Citra digital adalah bentuk citra yang telah direpresentasikan dalam format numerik sehingga dapat diolah oleh komputer. Citra ini terdiri dari matriks dua dimensi dengan koordinat spasial (x, y) , dimana setiap elemen dalam matriks disebut piksel (*picture element*). Piksel merupakan unit terkecil dalam citra digital yang

menyimpan informasi mengenai intensitas cahaya atau warna suatu titik pada gambar. Proses digitalisasi citra mencakup dua tahap utama, yaitu diskritisasi koordinat spasial (*sampling*), yang membagi citra kontinu menjadi titik-titik diskrit dalam bentuk grid, serta diskritisasi tingkat kuantisasi (*quantization*), yang mengubah intensitas Cahaya atau warna menjadi nilai numerik yang dapat dikenali oleh komputer [18].

Representasi dengan berbasis piksel, citra digital dapat ditampilkan pada layar komputer dan diproses lebih lanjut menggunakan berbagai Teknik pengolahan citra serta kecerdasan buatan. Hal ini memungkinkan penerapan *computer vision* dalam berbagai bidang, seperti pengenalan objek, analisis citra medis, serta klasifikasi dan deteksi objek dalam sektor pertanian dan industri.

Pada penelitian ini, model *Convolutional Neural Network (CNN)* dan *Faster R-CNN* digunakan, sehingga citra digital yang dipakai harus memenuhi sejumlah kriteria agar mampu menghasilkan klasifikasi dan deteksi objek yang akurat. Adapun karakteristik citra digital yang sesuai untuk kedua model tersebut Adalah sebagai berikut:

1. Format Citra Digital

Citra yang digunakan dalam penelitian ini harus memiliki format yang umum dalam pemrosesan gambar, seperti *JPEG*, *PNG*, atau *TIFF*. Format *PNG* lebih disarankan karena mampu mempertahankan kualitas gambar tanpa kehilangan informasi akibat proses kompresi. Pemilihan format ini penting untuk memastikan data yang digunakan tetap optimal dalam proses pelatihan model. Pada penelitian ini digunakan format citra gambar berupa *jpg/jpeg*.

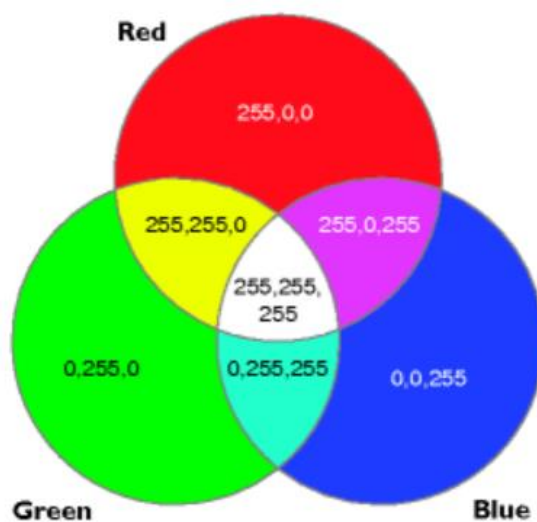
2. Resolusi dan Kualitas Gambar

Resolusi dan kualitas gambar memainkan peran penting dalam kinerja *CNN* dan *Faster R-CNN*. Kedua model ini bekerja lebih baik dengan citra beresolusi tinggi karena detail yang lebih jelas dapat meningkatkan kemampuan dalam mengekstraksi fitur. Resolusi gambar yang direkomendasikan untuk model *CNN* dan *Faster R-CNN* umumnya berkisar antara 224×224 piksel hingga 1024×1024 piksel, tergantung pada kompleksitas objek yang dianalisis. Namun, citra dengan resolusi yang terlalu tinggi dapat meningkatkan beban komputasi, sehingga

diperlukan proses normalisasi ukuran gambar agar seragam dan sesuai dengan kebutuhan model tanpa mengurangi informasi penting dalam gambar.

3. Citra *Red, Green, Blue* (RGB)

Citra *RGB* merupakan jenis citra digital yang terdiri dari tiga komponen warna utama, yaitu merah (*Red*), hijau (*Green*), dan biru (*Blue*). Setiap piksel dalam citra RGB tersusun dari kombinasi ketiga warna tersebut, dengan masing-masing nilai intensitas berada dalam rentang 0 hingga 255. Informasi setiap piksel disimpan dalam format 1 byte per warna, di mana 8 bit pertama merepresentasikan warna merah, 8 bit berikutnya hijau, dan 8 bit terakhir biru. Dengan demikian, setiap piksel dalam citra RGB memiliki kedalaman warna 24 bit, yang memungkinkan hingga 16.777.216 kombinasi warna ($256 \times 256 \times 256$) [19] seperti yang ditunjukkan pada gambar 2.3 berikut:



Gambar 2. 3 Warna RGB

Dalam penelitian ini, citra yang digunakan sebagai dataset untuk klasifikasi tingkat kematangan tandan buah segar kelapa sawit menggunakan model *Convolutional Neural Network* (CNN) dan *Faster R-CNN* berbasis *deep learning*. Citra yang digunakan adalah citra berwarna *RGB* (*True Color*) dengan format *.jpg* atau *.jpeg*, karena format ini umum digunakan dalam pemrosesan gambar serta memiliki keseimbangan antara kualitas dan ukuran file yang efisien untuk keperluan pelatihan model. *Faster R-CNN*, sebagai salah satu model deteksi objek yang canggih, akan memanfaatkan citra RGB untuk mendeteksi

dan mengklasifikasikan objek daun I secara lebih presisi berdasarkan fitur warna dan teksturnya.

2.7 Pengolaan Citra Digital

Pengolahan citra digital (*PCD*) adalah cabang ilmu yang mempelajari berbagai metode untuk meningkatkan kualitas gambar, seperti penyesuaian kontras, perubahan warna, dan restorasi gambar. *PCD* juga mencakup proses transformasi dan augmentasi gambar, termasuk translasi, perubahan skala, dan transformasi geometris seperti *flipping*, rotasi, *cropping*, *scaling*, serta *zooming*. Disiplin ini juga berfokus pada pemilihan fitur gambar yang tepat untuk analisis, ekstraksi informasi, deskripsi objek, dan identifikasi elemen dalam gambar. Selain itu, *PCD* turut berperan dalam kompresi atau pengurangan ukuran data demi efisiensi penyimpanan [20].

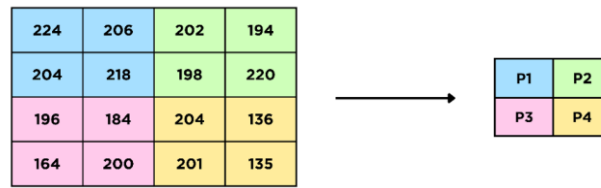
Adapun Teknik pengolahan citra digital yang diterapkan dalam penelitian ini meliputi: 1. *Resizing*

2. Augmentasi

2.6.1 *Resizing*

Resizing adalah proses untuk mengubah ukuran *pixel* suatu gambar menjadi panjang dan lebar sesuai kebutuhan, baik dengan mengecilkan ukuran gambar (*downscale*) maupun memperbesar (*upscale*). Langkah ini diperlukan karena ukuran gambar yang akan diproses tidak selalu seragam [21]. Dalam penelitian ini, *resizing* dilakukan untuk mengecilkan ukuran gambar agar jumlah *pixel* yang diproses lebih sedikit, sehingga waktu komputasi dapat dipercepat. Semakin besar jumlah *pixel*, semakin banyak data yang harus diproses, yang pada akhirnya memperlama waktu komputasi. Proses *resizing* dilakukan dengan menetapkan terlebih dahulu ukuran input yang diinginkan, kemudian menghitung ukuran asli setiap gambar. Selama tahap *resizing*, dilakukan interpolasi atau proses pembuatan ulang data citra dari nilai-nilai antara *pixel* yang telah ditetapkan. Proses *resizing* gambar ini dilakukan melalui *Google Colaboratory*.

Berikut merupakan contoh perhitungan proses *resizing* adalah sebagai berikut :



Gambar 2. 4 Proses Perhitungan *Resizing* [22]

Gambar 2.4 menunjukkan proses perhitungan untuk mengubah ukuran citra 4 x 4 *pixel* menjadi citra 2 x 2 *pixel* melibatkan penghitungan nilai rata-rata dari empat *pixel* yang berdekatan, yang kemudian digunakan sebagai nilai piksel baru dalam citra yang lebih kecil, yaitu 2 x 2. Berikut adalah rincian perhitungan dalam proses *resizing* tersebut.

$$P1 = (224 + 206 + 204 + 218) : 4 = 213$$

$$P2 = (202 + 194 + 198 + 220) : 4 = 203$$

$$P3 = (196 + 184 + 164 + 200) : 4 = 186$$

$$P4 = (204 + 136 + 201 + 135) : 4 = 169$$

213	203
186	169

Gambar 2. 5 Nilai matriks setelah *resizing* [22]

Pada gambar 2.5 menunjukkan nilai *pixel* baru yang dihasilkan oleh perhitungan *resizing* dengan matriks. Pada penelitian ini data citra yang dikumpulkan memiliki ukuran asli 3060 x 4080 piksel dan ukuran file yang bervariasi, dengan rata-rata sekitar 2 MB per gambar. Dalam penelitian ini, ukuran gambar diubah menjadi 300 x 300 *pixel* untuk menghasilkan dataset yang seragam, memudahkan proses pelatihan sistem, dan menghemat sumber daya komputasi. Proses *resizing* gambar ini dilakukan melalui *Google Colaboratory*.

2.6.2 Augmentasi

Augmentasi adalah teknik yang digunakan untuk meningkatkan jumlah dan keragaman data pelatihan dengan menghasilkan variasi dari data yang ada. Proses augmentasi mencakup transformasi citra data asli, seperti memutar gambar pada

interval tertentu, membalik gambar secara *vertical* dan *horizontal*, serta merotasi gambar sebesar 90 derajat. Teknik ini membantu meningkatkan kuantitas dataset. Augmentasi juga mengurangi risiko *overfitting*, yaitu model yang terlalu cocok dengan data pelatihan, dengan memungkinkan model mempelajari fitur-fitur yang lebih umum dan membantu model dalam menggeneralisasi data yang belum pernah dilihat sebelumnya [23]. Teknik flip pada gambar adalah metode yang digunakan untuk membalik atau merotasi gambar terhadap sumbu *horizontal* maupun *vertical*.

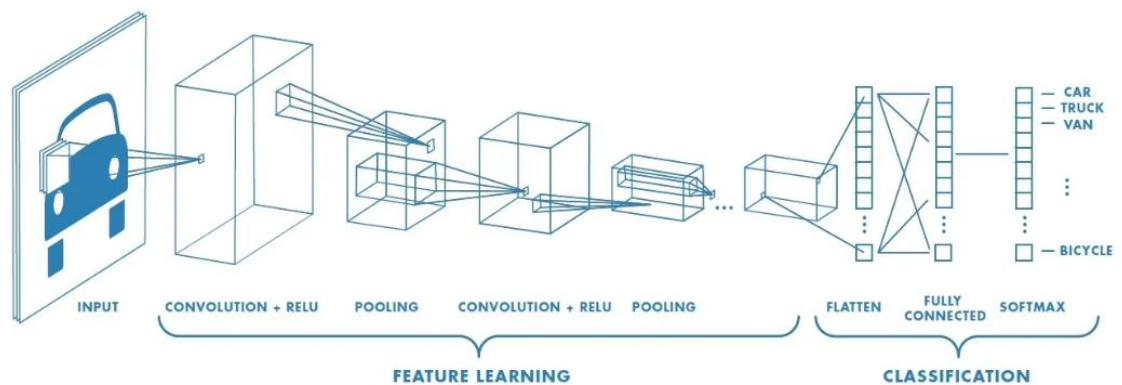
Rotasi 90 derajat adalah proses memutar gambar sesuai dengan sudut 90 derajat, baik searah maupun berlawanan arah jarum jam. Berikut adalah tiga jenis rotasi yang digunakan pada penelitian ini:

Clockwise (Searah Jarum Jam). Citra diputar sebesar 90° ke kanan. Dalam proses ini:

- Sisi kanan citra menjadi sisi bawah.
- Sisi bawah citra menjadi sisi kiri.
- Sisi kiri citra menjadi sisi atas.

2.8 Convolutional Neural Network (CNN)

Convolutional Neural Network (CNN) merupakan salah satu algoritma deep learning yang banyak digunakan. Dikembangkan sejak tahun 1960-an, *CNN* telah menunjukkan performa yang unggul dalam bidang *Computer Vision* dan dikenal sebagai salah satu jenis jaringan saraf yang paling merepresentasikan *deep learning* [24]. *Convolutional Neural Network (CNN)* adalah salah satu metode *deep learning* yang digunakan untuk mengenali dan mengklasifikasikan gambar. *CNN* termasuk dalam kategori neural network yang dirancang khusus untuk memproses data dengan struktur grid. Metode ini dapat diaplikasikan dalam berbagai bidang, seperti pengenalan wajah, analisis dokumen, klasifikasi gambar, dan video. *CNN* bekerja dengan memanfaatkan Kumpulan data berukuran besar yang mengandung jutaan parameter. Data berupa gambar diolah sebagai input dan diproses menggunakan filter untuk menghasilkan *output* sesuai dengan tujuan yang diinginkan seperti pada gambar 2. 6 Arsitektur CNN [25].



Gambar 2. 6 Arsitektur CNN [26]

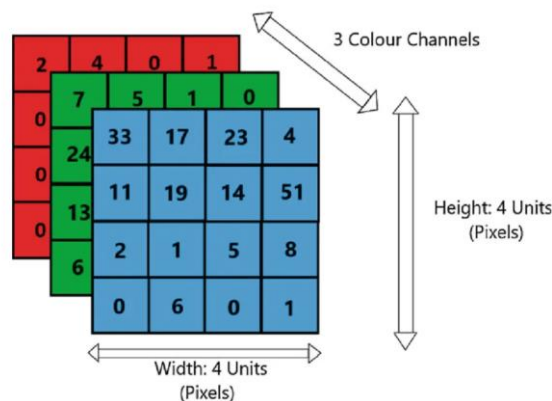
Berdasarkan gambar 2.6, alur kerja *CNN* terdiri dari beberapa tahapan utama. Proses dimulai dengan menerima input gambar (contohnya, gambar mobil), yang kemudian melewati serangkaian lapisan tersembunyi (*hidden layers*). Tahap pertama pertama konvolusi, di mana kernel dengan ukuran tertentu digunakan untuk mengekstrak fitur-fitur penting dari gambar, seperti tepi, bentuk, dan tekstur. Jumlah kernel yang diterapkan harus disesuaikan dengan jumlah fitur yang ingin dihasilkan. Setelah itu, fungsi aktivasi *ReLU* (*Rectified Linear Unit*) diterapkan untuk meningkatkan non-linearitas model. Selanjutnya, proses dilanjutkan dengan *pooling*, yang berfungsi untuk mereduksi dimensi fitur tanpa kehilangan informasi penting. Tahapan ini dapat diulang beberapa kali hingga diperoleh peta fitur yang cukup representatif. Setelah ekstraksi fitur selesai, hasilnya diratakan (*flatten*) dan diteruskan ke lapisan *fully connected* untuk melakukan klasifikasi akhir. Lapisan *output* (*softmax*) kemudian memberikan probabilitas terhadap setiap kelas yang mungkin (misalnya, mobil, truk, van, atau sepeda). Dengan demikian, *CNN* mampu mengenali pola dalam gambar dan mengklasifikasikannya berdasarkan pola tersebut. Penjelasan lebih lanjut mengenai fungsi setiap lapisan akan dibahas dalam sub-bab berikutnya.

2.8.1 Convolutional Layer

Convolutional Layer merupakan komponen dasar dalam arsitektur *CNN* yang berperan dalam proses ekstraksi fitur. Lapisan ini terdiri dari kernel atau filter yang pada awalnya memiliki bobot acak dan akan mengalami perubahan bobot selama proses pelatihan model. Filter tersebut diterapkan dengan melakukan operasi

konvolusi pada seluruh bagian citra, yaitu dengan menggeser dan mengalikannya dari kiri atas ke kanan bawah. Tujuan utama dari *Convolutional Layer* adalah mengekstraksi fitur penting dari citra input, seperti tepi, sudut, dan tekstur. Hasil dari proses ini disebut sebagai *feature map*.

Fungsi konvolusi menghasilkan satu output berupa *feature map* dengan dua argumen input, yaitu citra sebagai data masukan dan kernel atau filter yang digunakan dalam operasi konvolusi. Lapisan ini menentukan dimensi data dalam bentuk Tensor 3D, terutama untuk pemrosesan gambar, seperti ukuran (224, 224, 3) yang menunjukkan tinggi 224 piksel, lebar 224 piksel, dan tiga saluran warna RGB (*Red, Green, Blue*). Fungsi utama layer ini adalah menyiapkan data agar dapat diproses oleh lapisan-lapisan selanjutnya tanpa melibatkan perhitungan yang kompleks. Ilustrasi mengenai layer input dapat dilihat pada gambar 2.7.



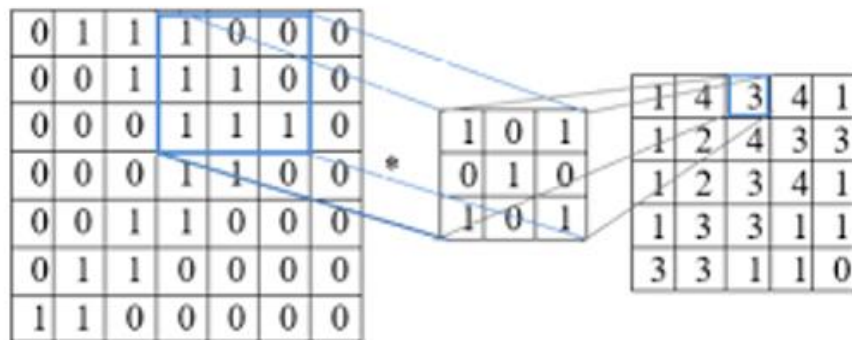
Gambar 2. 7 *Layer Input* [27]

Gambar 2.7 di atas menggambarkan ilustrasi dengan 3 *channel* yang memiliki tinggi 4 piksel dan lebar 4 piksel, meskipun ukuran lebar dan tinggi dapat menyesuaikan dengan gambar yang dimasukkan. Selanjutnya, lapisan konvolusi pada *CNN* memanfaatkan *filter* kernel untuk melakukan operasi konvolusi pada gambar input, dengan tujuan mengekstraksi fitur-fitur dasar dari gambar tersebut. *Filter* ini memiliki dimensi yang sama, tetapi ukurannya lebih kecil dibandingkan dengan gambar input dan memiliki parameter tetap [28].

Pada gambar 2.6, proses konvolusi dilakukan dengan menggeser matriks 3×3 (*filter* atau kernel) secara bertahap pada matriks input (gambar). Pada setiap posisi, elemen-elemen dalam filter dan matriks input dikalikan, kemudian dijumlahkan

untuk menghasilkan satu nilai dalam matriks output. Proses ini terus berulang hingga seluruh bagian gambar terkonvolusi, menghasilkan matriks fitur yang mampu menangkap informasi lokasi dari gambar input.

Lapisan konvolusi memiliki tiga parameter utama yang menentukan cara kerjanya. Ukuran kernel mengacu pada dimensi *filter* yang digunakan dalam pemrosesan. *Stride* atau panjang Langkah menentukan seberapa jauh kernel bergeser sebelum menghitung hasil perkalian untuk setiap piksel *output*. Sementara itu, *padding* adalah Teknik menambahkan bingkai nol di sekitar input guna mempertahankan dimensi *output* yang dihasilkan. Ilustrasi lebih lanjut mengenai proses komputasi ini dapat dilihat pada gambar 2.8.



Gambar 2. 8 *Convolution Layer* [29]

Pada gambar 2.8, proses konvolusi dilakukan tanpa penambahan *padding* pada gambar *input*. Matriks 3x3 digeser dengan *stride* sebesar 1 piksel secara bertahap di sepanjang matriks *input*. Pada setiap langkah, elemen-elemen dalam *filter* dan matriks *input* dikalikan satu per satu, lalu hasilnya dijumlahkan untuk menghasilkan satu nilai dalam matriks *output*, misalnya diperoleh nilai 3. Proses ini terus berulang hingga seluruh area gambar diproses, membentuk matriks fitur yang menangkap informasi Lokasi dari gambar input [30].

Ketika proses konvolusi selesai, kemudian dilakukan aktivasi menggunakan fungsi aktivasi *ReLU* (*Rectified Linear Unit*). *ReLU* adalah fungsi aktivasi *linear* yang umum digunakan dalam *CNN*. Fungsi ini mengubah nilai piksel pada *feature map* yang masuk dengan menetapkan nilai negatif dari layer sebelumnya menjadi 0 [25]. Hasil dari fungsi *ReLU* terhadap hasil konvolusi terdapat pada Gambar 2.9.

Matriks masukan					Matriks setelah ReLU			
-1.228	1.844	0.845	0.333	ReLU →	0	1.844	0.845	0.333
-1.216	2.350	0.723	0.357		0	2.350	0.723	0.357
-1.145	2.538	0.686	0.510		0	2.538	0.686	0.510
-1.004	1.812	0.658	0.616		0	1.812	0.658	0.616

Gambar 2. 9 Aktivasi ReLU [29]

Proses ini dapat direpresentasikan dengan persamaan berikut.

$$F(x) = \max(0, x) \dots \dots \dots (2.1)$$

Dimana:

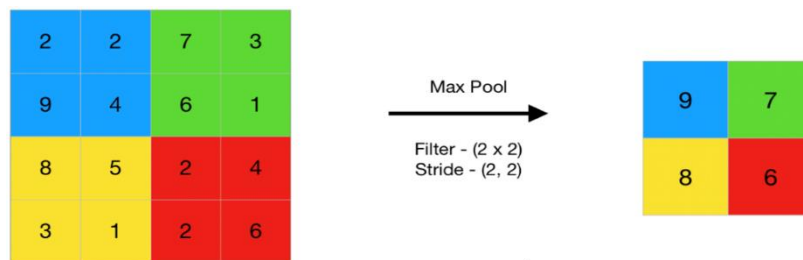
- $f(x)$ 22 adalah output fungsi ReLU.
- Jika $x > 0$, maka $f(x)=x$, yang berarti nilai input tetap diteruskan tanpa perubahan.
- Jika $x \leq 0$, maka $f(x)=0$, yang berarti nilai negatif diubah menjadi nol.

Persamaan tersebut menunjukkan bahwa fungsi ReLU memiliki dua kemungkinan output berdasarkan nilai inputnya: jika input bernilai positif, maka nilai tersebut diteruskan tanpa perubahan; sedangkan jika input bernilai nol atau negatif, maka output menjadi nol. Hal ini tidak hanya menyederhanakan perhitungan dalam jaringan saraf tiruan, tetapi juga mempermudah analisis optimasi, terutama dalam perhitungan gradien selama proses pembelajaran menggunakan algoritma backpropagation.

2.8.2 Pooling Layer

Lapisan ini berfungsi menangkap informasi yang diperoleh melalui proses ekstraksi fitur secara bertahap, dengan tujuan mengurangi ukuran representasi data. Selain

itu, lapisan ini juga mengurangi jumlah parameter untuk menekan kompleksitas komputasi model. Metode *pooling* yang sering digunakan adalah *max pooling*, yang mengambil nilai terbesar, dan *average pooling*, yang menghitung nilai rata-rata [31]. Penelitian ini menggunakan *pooling* dengan matriks berukuran 2×2 dan *stride* 2. Dengan demikian, *pooling* akan bergerak sejauh 2 indeks dan memilih nilai terbesar dari *region pooling*, yang disebut dengan *max pooling*.



Gambar 2. 10 Max Pooling [33]

Pada ilustrasi gambar 2.10 di atas, *feature map* yang awalnya berisi 16 piksel dikompresi menjadi hanya 4 piksel melalui proses *max-pooling*. Meskipun ukurannya berkurang, informasi krusial, terutama lokasi fitur dengan nilai tertinggi, tetap dipertahankan. Proses ini menggunakan *region* berukuran 2×2 , di mana setiap *region* diwakili oleh nilai tertinggi yang ditemukan di dalamnya, seperti nilai 6, yang kemudian menjadi representasi dari area tersebut pada *feature map* yang baru.

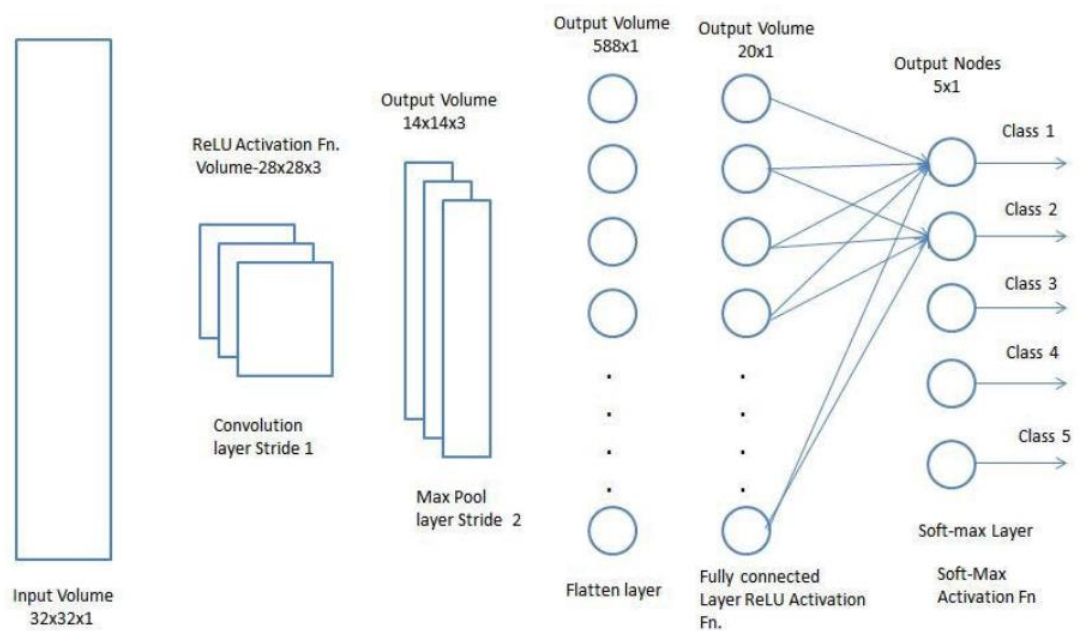
2.8.3 Fully Connected Layer

Fully connected layer merupakan lapisan ketiga yang terdiri dari hasil-hasil yang diperoleh melalui proses konvolusi. Citra akan menjadi input pada *fully connected layer* setelah melewati tahap konvolusi. Proses di *fully connected layer* terjadi di bagian akhir arsitektur *Convolutional Neural Network (CNN)*. Pada tahap ini, prosesnya melibatkan perkalian antara matriks sederhana dengan input, diikuti oleh penambahan vector bias, kemudian diikuti dengan penerapan fungsi *non-linear* [32].

2.8.4 Activation Function

Fungsi aktivasi adalah operasi matematis yang diterapkan pada *output* yang telah difilter untuk memberikan transformasi tambahan [33]. Pada lapisan *output*, fungsi aktivasi yang umum digunakan adalah

softmax. Lapisan ini berperan dalam menghasilkan hasil akhir dari model, terutama dalam tugas klasifikasi, seperti memprediksi kategori suatu objek. Seperti yang ditunjukkan pada Gambar 2.11, lapisan *Fully Connected* dan lapisan *Output* divisualisasikan dengan warna yang berbeda. Lapisan *output* biasanya berbentuk lapisan *dense* yang menggunakan fungsi aktivasi sesuai dengan jenis prediksi yang diinginkan, misalnya *softmax* untuk klasifikasi multikelas.

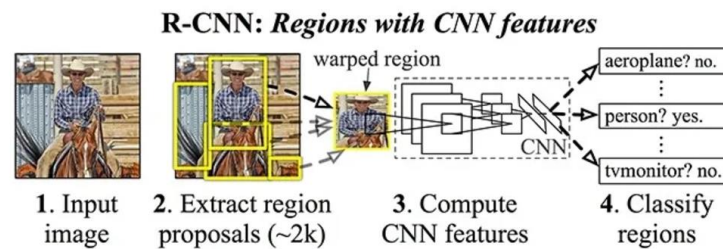


Gambar 2. 11 *Fully Connected Layer* [34]

Gambar 2.11 di atas menggambarkan proses kerja *Fully Connected Layer* dalam arsitektur CNN. Proses dimulai dengan input gambar grayscale berukuran $32 \times 32 \times 1$, yang kemudian melewati *Convolution Layer* dengan *stride 1*, menghasilkan fitur berukuran $28 \times 28 \times 3$ setelah diterapkan fungsi aktivasi *ReLU*. Selanjutnya, fitur tersebut dikurangi dimensinya melalui *Max Pooling Layer* dengan *stride 2*, sehingga ukurannya menjadi $14 \times 14 \times 3$. Hasil dari proses ini kemudian diratakan pada *Flatten Layer* menjadi vektor 1D (588×1), yang selanjutnya diproses dalam *Fully Connected Layer* dengan aktivasi *ReLU* untuk mendeteksi pola yang lebih kompleks. Pada tahap akhir, *Softmax Layer* menghasilkan probabilitas untuk masing-masing dari 5 kelas, di mana kelas dengan probabilitas tertinggi dipilih sebagai hasil klasifikasi.

2.9 Region Convolutional Neural Network (R-CNN)

Region Convolutional Neural Network (R-CNN) adalah metode berbasis *deep learning* yang digunakan untuk mendeteksi objek dalam pemrosesan citra. Dibandingkan dengan *CNN*, *R-CNN* menawarkan perbaikan dalam cara memproses citra input. Berikut merupakan gambar arsitektur dari *R-CNN*.

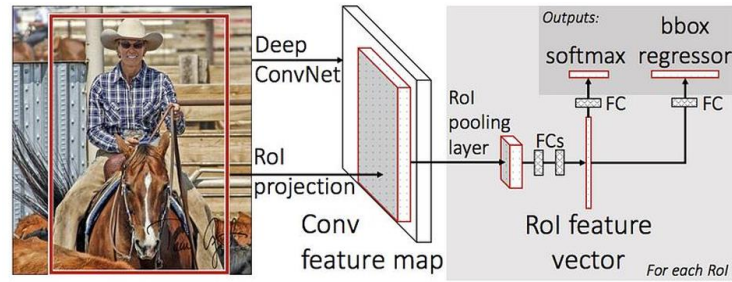


Gambar 2. 12 Arsitektur R-CNN [35]

Gambar 2.12 merupakan arsitektur dari *R-CNN* dimana pada *R-CNN*, citra dibagi menjadi 2000 wilayah (*region*) berdasarkan fitur seperti tekstur, warna, dan intensitas, sehingga proses deteksi menjadi lebih terfokus dan efisien. Sementara itu, pada metode *CNN*, pembagian wilayah citra dilakukan dalam skala besar, yang menyebabkan proses pemrosesan citra menjadi lambat. *R-CNN* menggunakan algoritma yang bertugas menemukan dua wilayah yang memiliki kemiripan tinggi, kemudian menggabungkannya. Algoritma ini diterapkan dalam teknik *selective search* [36].

2.10 Fast Region Convolutional Neural Network (Fast R-CNN)

Fast R-CNN, yang dikembangkan oleh Ross Gershick pada tahun 2015, merupakan pengembangan lanjutan dari metode *R-CNN*. Metode ini mampu melakukan proses *training* dan *testing* data dengan lebih cepat dibandingkan *R-CNN*. Tidak seperti *R-CNN*, *Fast R-CNN* hanya menggunakan satu *CNN*. Fungsi *R-CNN* sebagai *classifier* dengan ROI Pooling dan *fully connected layer* digantikan oleh *SVM* [36].

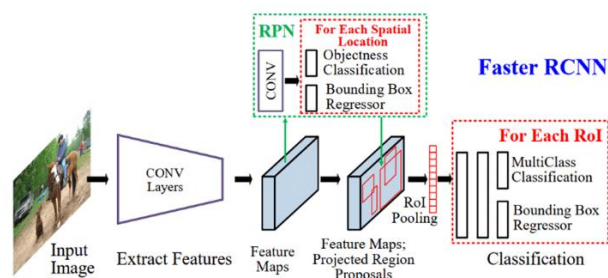


Gambar 2. 13 Arsitektur *Fast R-CNN* [35]

Gambar 2.13 merupakan arsitektur dari *Fast R-CNN*, dalam *Fast R-CNN*, hasil *feature map* dicocokkan dengan *ROI (Region of Interest)* sebelum dilakukan proses klasifikasi kelas. Pendekatan ini menggunakan satu *CNN* pada *feed-forward network* dan *ROI Pooling Layer*, yang memberikan kemampuan tambahan bagi *R-CNN* untuk menjadi *end-to-end differentiable*. Hal ini mempermudah proses pelatihan model.

2.11 *Faster Region Convolutional Neural Network (Faster R-CNN)*

Penelitian ini menggunakan metode *Faster Region Convolutional Neural Network (Faster R-CNN)*, yang merupakan pengembangan dari algoritma *Fast R-CNN*. Baik *R-CNN* maupun *Fast R-CNN* memerlukan waktu tambahan untuk menentukan *region proposal* menggunakan *selective search*. Sebagai solusinya, *Faster R-CNN* menghilangkan kebutuhan akan algoritma selektif tersebut dan memungkinkan jaringan untuk mempelajari proposal wilayah secara langsung [37].



Gambar 2. 14 Arsitektur Umum *Faster R-CNN* [38]

Gambar 2.14 merupakan arsitektur umum dari *Faster R-CNN*, pada *Faster R-CNN*, citra diinput dan diproses melalui lapisan *CNN* menggunakan *pre-trained model ResNet50*. Proses konvolusi menghasilkan *feature map* yang menyediakan peta fitur konvolusional. Proposal wilayah kemudian diproses menggunakan *Region*

Proposal Network (RPN), yang memiliki 9 *anchors* untuk menilai apakah area tertentu merupakan objek atau bukan. *RPN* ini kemudian diprediksi, diolah Kembali melalui lapisan *ROI pooling*, lalu digunakan untuk mengklasifikasikan objek pada *region proposal* dan memperkirakan nilai *offset* pada *bounding box*.

Berikut adalah bagian-bagian dan cara kerja *Faster R-CNN*:

- *Convolutional Layer*

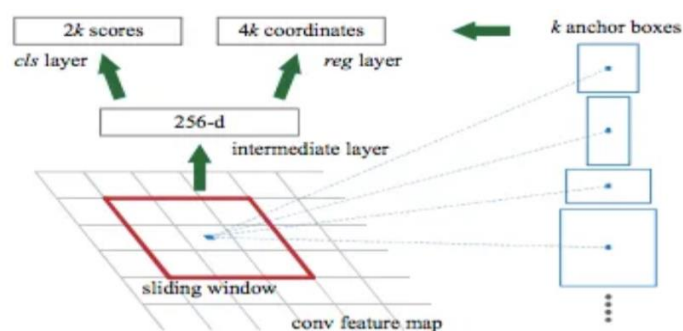
Lapisan ini terdiri dari filter dua dimensi dengan ukuran panjang dan lebar. Fungsinya adalah mempelajari karakteristik yang menjadi ciri khas suatu objek dan menghasilkan feature map dari objek tersebut.

- *Feature Map*

Berisi informasi berupa representasi vektor dari citra yang telah diolah. Komponen ini dihasilkan oleh convolutional layer.

- *Region Proposal Network (RPN)*

Region Proposal Network (RPN) adalah tugas yang cepat dan berguna untuk mencari kemungkinan lokasi objek dalam sebuah gambar. Posisi objek dalam gambar ini biasanya dibatasi oleh area yang disebut *region of interest (ROI)*. *RPN* menerima gambar dengan berbagai ukuran sebagai input dan menghasilkan sekumpulan proposal objek berbentuk persegi panjang, di mana setiap proposal dilengkapi dengan skor yang menunjukkan seberapa besar kemungkinan itu adalah objek yang dimaksud [39].



Gambar 2. 15 Arsitektur *Region Proposal Network (RPN)*

Pada gambar 2.15 Arsitektur *Region Proposal Network (RPN)*, gambar pertama-tama dimasukkan ke dalam jaringan *Convolutional Neural Network (CNN)*. Gambar input kemudian diteruskan ke layer konvolusional terakhir, yang menghasilkan *feature map*. *Sliding window* diterapkan pada setiap bagian dari

feature map. Ukuran *sliding window* biasanya berupa masker nn. Untuk setiap *sliding window*, secara bersamaan dilakukan prediksi untuk beberapa region proposal, dengan jumlah maksimum proposal untuk setiap lokasi dilambangkan sebagai k . *Layer regresi (reg)* menghasilkan output $4k$ yang mengkodekan koordinat kotak, sementara layer klasifikasi (cls) menghasilkan $2k$ yang memperkirakan probabilitas apakah ada objek atau tidak di setiap proposal. Setiap anchor diposisikan di Tengah sliding window. Secara default, digunakan 3 skala dan 3 rasio aspek, yang menghasilkan $k=9$ anchor untuk setiap *sliding window*. Klasifikasi menunjukkan probabilitas 0 atau 1 yang menunjukkan apakah wilayah tersebut mengandung objek atau tidak [43].

- *ROI Pooling*

Lapisan *pooling RoI* menggunakan *max pooling* untuk mengubah fitur dalam *Region of Interest (RoI)* menjadi *feature maps* dengan ukuran spasial tetap $H \times W$ (misalnya, 7×7), di mana H dan W merupakan parameter sistem untuk *RoI* tertentu. *RoI* sendiri adalah jendela persegi yang diambil dari *feature map* hasil konvolusi. Setiap *RoI* didefinisikan oleh empat nilai tuple (r, c, h, w) , yang menunjukkan posisi sudut kiri atas (r, c) serta tinggi dan lebar (h, w) dari *RoI* tersebut. Lapisan *pooling RoI* ini bekerja dengan cara menerapkan *max pooling* pada area yang *valid* dalam *RoI* untuk menghasilkan peta fitur kecil dengan ukuran spasial tetap, seperti $H \times W$.

- *Classification Layer*

Tahap akhir dalam proses klasifikasi, yang bertugas mengidentifikasi objek berdasarkan deteksi *RPN* dan memberikan label pada objek yang telah diklasifikasikan.

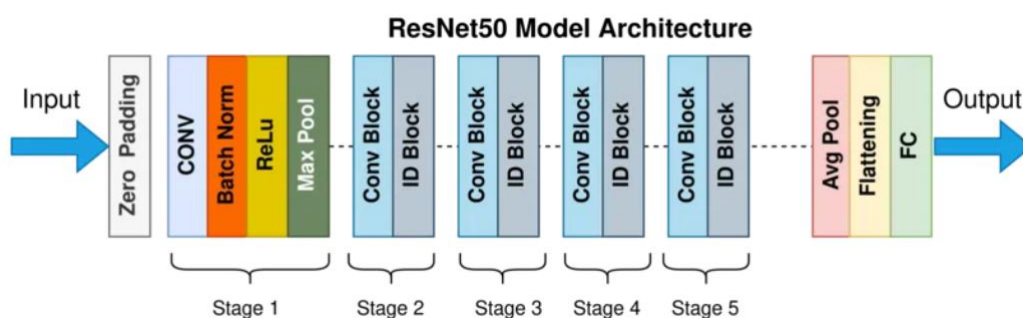
2.12 Resnet-50

Residual Network (ResNet) merupakan salah satu jaringan saraf dalam *deep learning* yang memiliki kontribusi signifikan dalam meningkatkan pembelajaran dan kinerja model. Dalam arsitektur *deep learning*, jaringan dengan kedalaman lebih besar digunakan untuk mengenali pola yang lebih kompleks serta mengekstraksi fitur tingkat tinggi. Namun, peningkatan kedalaman ini sering kali menimbulkan permasalahan *gradien* menghilang saat *backpropagation*, sehingga proses pelatihan menjadi kurang optimal. Untuk mengatasi masalah tersebut, *ResNet* dikembangkan dengan inovasi utama berupa *residual connections*, yang

memungkinkan jaringan untuk melewati beberapa lapisan tertentu guna menjaga aliran gradien tetap stabil [40].

ResNet pertama kali diperkenalkan sebagai solusi atas kesulitan dalam melatih jaringan saraf yang sangat dalam dan berhasil memenangkan tantangan *ImageNet* pada tahun 2015. Model ini membuktikan bahwa dengan adanya residual connections, performa jaringan dapat terus meningkat seiring bertambahnya kedalaman, dengan berbagai varian seperti *ResNet-18*, *ResNet-32*, *ResNet-50*, *ResNet-101*, dan *ResNet-152*. Pada penelitian ini, digunakan *ResNet-50*, salah satu arsitektur dalam keluarga *Residual Networks* yang memiliki 50 lapisan. *ResNet-50* sangat efektif dalam tugas klasifikasi citra, termasuk klasifikasi citra biner, yang sering menghadapi kendala hilangnya gradien, terutama saat menggunakan fungsi aktivasi sigmoid.

Keunggulan utama *ResNet-50* terletak pada kemampuannya dalam mempertahankan aliran gradien yang lebih stabil melalui penggunaan *skip connections*, yang memungkinkan gradien mengalir lebih efisien selama proses *backpropagation*. Dengan struktur ini, *ResNet-50* mampu mengoptimalkan jaringan lebih baik dan mencapai tingkat akurasi yang lebih tinggi dibandingkan model konvensional lainnya. Oleh karena itu, arsitektur *ResNet-50* menjadi pilihan yang tepat dalam penelitian ini untuk meningkatkan efektivitas klasifikasi citra. Gambar berikut menunjukkan arsitektur *ResNet-50* yang digunakan dalam penelitian ini.



Gambar 2. 16 Arsitektur *ResNet50* [41]

Berdasarkan gambar 2.16 secara umum, arsitektur *ResNet-50* terdiri dari lapisan konvolusi awal, diikuti oleh beberapa blok residu. Setiap blok residu mencakup

beberapa lapisan konvolusi dengan *shortcut connection*. Setelah blok residu terakhir, terdapat lapisan pooling global yang bertujuan untuk mengurangi dimensi spasial dari fitur-fitur yang dihasilkan. Fitur-fitur tersebut kemudian dihubungkan ke lapisan *fully connected* yang menghasilkan output untuk klasifikasi [42].

2.13 Google Colaboratory

Google Colab (Colaboratory) adalah sebuah *platform* berbasis *cloud* yang dikembangkan oleh *Google* untuk mempermudah pengguna dalam menjalankan dan berbagi kode *Python*, khususnya dalam pengolahan data dan pengembangan model pembelajaran mesin.



Gambar 2. 17 *Google Colaboratory*

Gambar 2.17 merupakan logo dari *Google Colaboratory*. *Platform* ini memungkinkan pengguna untuk menulis dan menjalankan kode dalam format *notebook* interaktif langsung dari *browser* tanpa perlu menginstal perangkat lunak apa pun. *Colab* menyediakan akses gratis ke sumber daya komputasi seperti GPU dan TPU, yang sangat berguna untuk pelatihan model *deep learning* yang membutuhkan pemrosesan intensif. Selain itu, pengguna dapat menyimpan *notebook* yang dibuat langsung di *Google Drive*, sehingga memudahkan kolaborasi dan penyimpanan data dengan aman *Platform* ini juga mendukung instalasi berbagai *library* eksternal melalui *pip*, serta memungkinkan visualisasi data secara interaktif menggunakan Pustaka seperti *Matplotlib* dan *Seaborn*.

2.14 Python

Python adalah bahasa pemrograman yang bersifat interpretatif, memiliki semantik yang jelas, dan mendukung pemrograman dinamis.



Gambar 2. 18 Logo *Software Python*

Gambar 2.18 merupakan logo dari *software python*. Desainnya dirancang untuk meningkatkan keterbacaan kode, sehingga mempermudah pengembang dalam memahami *source code*. Dengan sintaks yang sederhana, *Python* memungkinkan programmer untuk menciptakan berbagai aplikasi modern, didukung oleh koleksi *library* yang sangat lengkap.

2.15 *TensorFlow*

TensorFlow adalah sebuah *platform open-source* yang dikembangkan oleh *Google* untuk mendukung pengembangan dan implementasi model pembelajaran mesin dan kecerdasan buatan.



Gambar 2. 19 Logo *TensorFlow*

Gambar 2.19 merupakan logo dari *platform TensorFlow*, *platform* ini dirancang untuk memenuhi kebutuhan komputasi numerik dengan menggunakan pendekatan berbasis grafik aliran data (*data flow graph*), di mana setiap node dalam grafik mewakili operasi matematis, dan setiap *edge* mewakili aliran data antar operasi tersebut. *TensorFlow* mendukung berbagai jenis algoritma pembelajaran, seperti jaringan saraf tiruan (*Artificial Neural Networks*), jaringan saraf konvolusional (*Convolutional Neural Networks*), dan jaringan saraf berulang (*Recurrent Neural Networks*). Dengan fleksibilitasnya, *TensorFlow* dapat digunakan untuk membangun model pembelajaran sederhana hingga model berskala besar yang memerlukan komputasi intensif.

2.16 Keras

Keras adalah sebuah *framework* atau *library* yang digunakan untuk pengembangan *deep learning*. *Framework* ini dirancang untuk mempermudah pengguna serta mengurangi tingkat kerumitan dalam membangun model *deep learning*. Keras dibuat dengan tujuan untuk mendukung algoritma seperti *RNN*, *CNN*, dan kombinasi keduanya, yang dapat dijalankan baik pada perangkat GPU maupun CPU. Sejak diluncurkannya *TensorFlow* versi 2.0, Keras telah menjadi API resmi tingkat tinggi dari *TensorFlow*, sehingga Keras kini dapat digunakan sebagai bagian terpadu dalam ekosistem pengembangan *TensorFlow*.

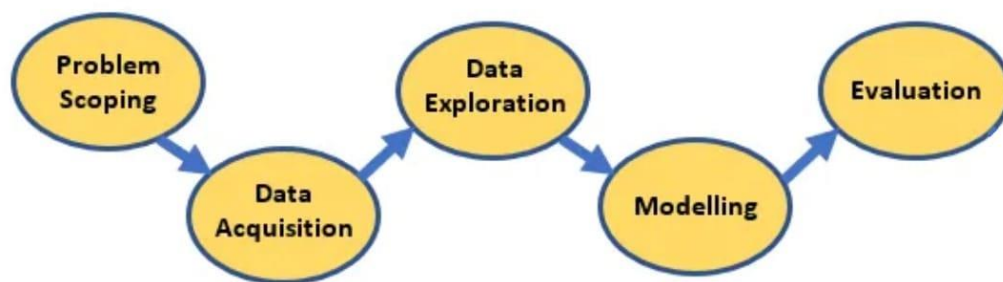
2.17 Optimizer Adam

Optimasi pada *neural network* adalah proses penyesuaian parameter model, seperti bobot dan bias, untuk meminimalkan *loss* selama pelatihan. Tujuan algoritma optimasi adalah menemukan bobot optimal yang mampu meminimalkan *loss* dan memaksimalkan akurasi. Selama pelatihan, parameter model diubah untuk menekan nilai *loss function*, yang mengukur tingkat akuratan model dalam memprediksi data uji. Algoritma optimasi menggabungkan *loss function* dan parameter model untuk melakukan pembaruan model. Secara sederhana, algoritma optimasi membentuk model yang lebih akurat melalui pemanfaatan bobot. Proses ini melibatkan metode seperti *gradient descent*, yang digunakan untuk menghitung gradien *loss* terhadap parameter dan memperbarui parameter tersebut agar *loss* berkurang.

Adam Optimizer merupakan salah satu algoritma optimasi yang banyak digunakan dalam *deep learning*. Algoritma ini memperbarui bobot dengan *learning rate* yang dapat berubah selama proses pelatihan. Adam menggabungkan keunggulan *Momentum* dan *RMSProp* untuk mengatasi masalah gradien jarang (*sparse gradient*) sehingga mencapai kinerja yang baik. Dalam *deep learning*, pemilihan *optimizer* berpengaruh terhadap tingkat akurasi model. Adam, yang merupakan singkatan dari *Adaptive Moment Estimation*, memiliki kemampuan untuk memperbarui bobot dan *learning rate* secara adaptif. Dengan demikian, selama pelatihan menggunakan Adam Optimizer, nilai *learning rate* dapat berubah-ubah sesuai sifat adaptif yang dimilikinya.

2.18 Siklus proyek *AI*

Dalam pengembangan solusi berbasis *Artificial Intelligence (AI)*, prosesnya perlu dibagi ke dalam beberapa tahap untuk memastikan efisiensi dan efektivitas. Kerangka kerja Siklus Proyek *AI* menyediakan pendekatan sistematis dengan membagi tahapan pengembangan proyek *AI*, khususnya dalam domain *Computer Vision (CV)*, sehingga setiap Langkah dapat dilakukan secara lebih terstruktur dan terorganisir.



Gambar 2. 20 Siklus Proyek *AI*

2.18.1 *Problem Scoping*

Tahapan ini bertujuan untuk mengidentifikasi dan memahami permasalahan yang akan diselesaikan menggunakan solusi *AI*. Dalam proses ini, pengguna harus merumuskan permasalahan serta menetapkan tujuan dari proyek agar arah pengembangan menjadi lebih jelas. Dengan pendekatan yang terstruktur, *AI* dapat diterapkan secara efektif dan efisien untuk mengatasi permasalahan yang dihadapi. Pada penelitian ini berfokus dalam bidang bidang perkebunan kelapa sawit.

2.18.2 *Data Acquisition*

Tahapan ini mencakup proses pemilihan sumber data yang akan digunakan dalam melatih model *CNN* dan *Faster R-CNN* dengan *backbound ResNet50*. Proses pengumpulan data dilakukan dengan teknik *web scraping* serta mengambil data langsung dari Perkebunan kelapa sawit untuk pelatihan dan evaluasi dari model yang telah dibuat.

2.18.3 *Data Exploration*

Setelah proses pengumpulan, data perlu dieksplorasi dan dianalisis untuk memastikan keakuratan serta menghindari adanya kesalahan atau kerusakan. Tahap ini mencakup filter guna menjaga kualitas data. Selain itu, visualisasi data sering digunakan untuk mendeteksi pola dan tren yang dapat memengaruhi hasil akhir, sehingga membantu dalam pengambilan keputusan yang lebih tepat.

2.18.4 *Modelling*

Pada tahap ini, pemilihan model *AI* yang sesuai dilakukan untuk memastikan performa yang optimal. Model kemudian dilatih menggunakan data yang telah dikumpulkan dan melalui proses *preprocessing* pada tahap sebelumnya. Proses ini mencakup pemilihan algoritma yang tepat serta optimasi model agar mencapai kinerja terbaik. Dalam penelitian ini, digunakan model *CNN* dan *Faster R-CNN* dengan *backbone ResNet50*, yang dirancang untuk meningkatkan akurasi dalam tugas deteksi dan klasifikasi objek.

2.18.5 *Evaluation*

Setelah proses pelatihan selesai, evaluasi kinerja model dilakukan untuk mengukur efektivitas dan efisiensinya. Tahap ini mencakup pengujian model menggunakan data uji guna memastikan kemampuannya dalam melakukan generalisasi terhadap data baru. Hasil evaluasi akan menjadi dasar dalam menentukan apakah model perlu dilakukan penyesuaian, pelatihan ulang, perbaikan pada dataset, atau sudah siap untuk digunakan. Umumnya, kinerja model dinilai berdasarkan metrik evaluasi tertentu yang digunakan dalam pengembangan *AI*.

2.19 Confussion Matrix

Confusion matrix adalah metode yang digunakan untuk menilai kinerja dalam masalah klasifikasi yang menghasilkan output dengan dua kelas atau lebih. Perbedaan antara nilai prediksi dan nilai aktual pada confusion matrix menghasilkan empat kombinasi. Tabel *confusion matrix* dapat dilihat pada gambar berikut.

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

Gambar 2. 21 Tabel *Confusion Matrix* [43]

Berdasarkan gambar 2.20 tabel *confusion matrix*, hasil dari proses klasifikasi dapat dijelaskan dalam empat istilah, yaitu:

- True Positive* (TP) terjadi ketika model menebak bahwa buah kelapa sawit sudah matang, dan kenyataannya memang benar matang. Misalnya, model memprediksi buah sawit siap panen, dan setelah diperiksa secara manual, buah tersebut memang berada pada tingkat kematangan yang sesuai untuk dipanen.
- True Negative* (TN) terjadi ketika model menebak bahwa buah kelapa sawit belum matang, dan kenyataannya memang benar belum matang. Contohnya, model mengklasifikasikan buah sebagai masih mentah atau mengkal, dan setelah dilakukan pengecekan, buah tersebut memang belum siap untuk dipanen.
- False Positive* (FP) terjadi ketika model menebak bahwa buah kelapa sawit sudah matang, tetapi kenyataannya belum matang. Kesalahan ini dapat menyebabkan panen prematur yang berisiko menghasilkan minyak sawit dengan kualitas rendah dan volume yang lebih sedikit. Misalnya, model memprediksi buah siap panen, namun saat diperiksa, buah masih mentah atau belum mencapai tingkat kematangan optimal.

- d) *False Negative* (FN) terjadi ketika model menebak bahwa buah kelapa sawit belum matang, tetapi kenyataannya sudah matang. Kesalahan ini dapat menyebabkan keterlambatan panen, yang berisiko menurunkan kualitas minyak sawit atau bahkan membuat buah membusuk di pohon. Sebagai contoh, model mengklasifikasikan buah sebagai belum matang, tetapi saat diperiksa, ternyata buah sudah berada pada kondisi terbaik untuk dipanen.

Kesalahan *False Positive* dan *False Negative* harus diminimalkan agar proses panen dapat dilakukan dengan tepat waktu dan menghasilkan minyak sawit berkualitas tinggi.

Confusion matrix juga digunakan untuk menghitung berbagai *performance metrics*, salah satunya adalah:

2.19.1 Accuracy

Accuracy adalah metode untuk mengukur sejauh mana model dapat mengklasifikasikan data dengan benar. Persamaan untuk menghitung *accuracy* dapat dilihat pada Persamaan berikut [44].

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \times 100\% \dots\dots\dots (2.2)$$

Accuracy juga menunjukkan seberapa dekat nilai prediksi dengan nilai aktual. Semakin banyak data yang diprediksi dengan benar, maka semakin tinggi pula akurasi model dalam suatu penelitian.

2.19.2 Precision

Presisi mengukur seberapa akurat model dalam mengklasifikasikan sampel sebagai positif. Persamaan untuk menghitung *precision* dapat dilihat pada Persamaan berikut [44].

$$Precision = \frac{TP}{TP + FP} \times 100\% \dots\dots\dots (2.3)$$

Precision adalah metode yang digunakan untuk menghitung perbandingan antara jumlah data yang benar-benar sesuai dengan hasil yang diperoleh oleh sistem,

dibandingkan dengan jumlah total data yang dibaca oleh sistem, baik yang sesuai maupun yang tidak sesuai.

2.19.3 Recall

Recall mengukur seberapa baik model dalam mendeteksi semua instance yang sebenarnya positif. Persamaan untuk menghitung *recall* dapat dilihat pada Persamaan berikut [44].

$$Recall = \frac{TP}{TP + FN} \dots\dots\dots (2.4)$$

Recall adalah metode yang digunakan untuk menghitung perbandingan antara jumlah data yang benar-benar sesuai dengan hasil yang diberikan oleh sistem, dibandingkan dengan seluruh data yang relevan dalam kumpulan data, baik yang terbaca maupun yang tidak terbaca oleh sistem.

2.19.4 F1-Score

F1-Score merupakan metode yang digunakan untuk menghitung perbandingan dari rata-rata precision dan recall yang dibobotkan. Adapun persamaan dari *F-1 Score* terdapat pada Persamaan berikut [48].

$$F1 - Score = 2 \times \frac{Recall \times Precision}{Recall + Precision} \dots\dots\dots (2.5)$$

F1-Score berguna dalam kasus di mana terdapat ketidakseimbangan kelas dalam dataset. Jika hanya presisi yang tinggi, tetapi recall rendah, atau sebaliknya, *F1-Score* membantu menemukan titik keseimbangan yang optimal antara kedua metrik tersebut. Metrik ini sering digunakan dalam berbagai aplikasi pembelajaran mesin, terutama ketika penting untuk mempertimbangkan baik *False Positive* maupun *False Negative*.

2.19.5 Mean Average Precision

Mean Average Precision (mAP) merupakan metrik yang digunakan untuk mengevaluasi seberapa akurat suatu model dalam mengklasifikasi tingkat kematangan tandan buah segar kelapa sawit dari berbagai kelas dalam sebuah

dataset. Adapun persamaan dari *Mean Average Precision (mAP)* terdapat pada Persamaan berikut [44].

$$mAP = \frac{1}{n} \sum_{i=1}^n AP_i \dots \dots \dots (2.6)$$

Semakin tinggi nilai *mAP*, maka semakin baik kinerja model dalam mengidentifikasi dan mengklasifikasi tingkat kematangan kelapa sawit secara tepat dan menyeluruh. Nilai *mAP* merupakan rata-rata dari *Average Precision (AP)* pada semua kelas. Dimana *AP_i* merupakan *AP* untuk kelas ke-I dan N merupakan total kelas yang dievaluasi.

BAB III METODOLOGI PENELITIAN

3.1 Waktu dan Tempat Penelitian

Adapun penelitian ini dilaksanakan di Laboratorium Telekomunikasi, Laboratorium Terpadu Jurusan Teknik Elektro, Universitas Lampung, dimulai pada bulan November 2024 sampai dengan bulan Mei 2025.

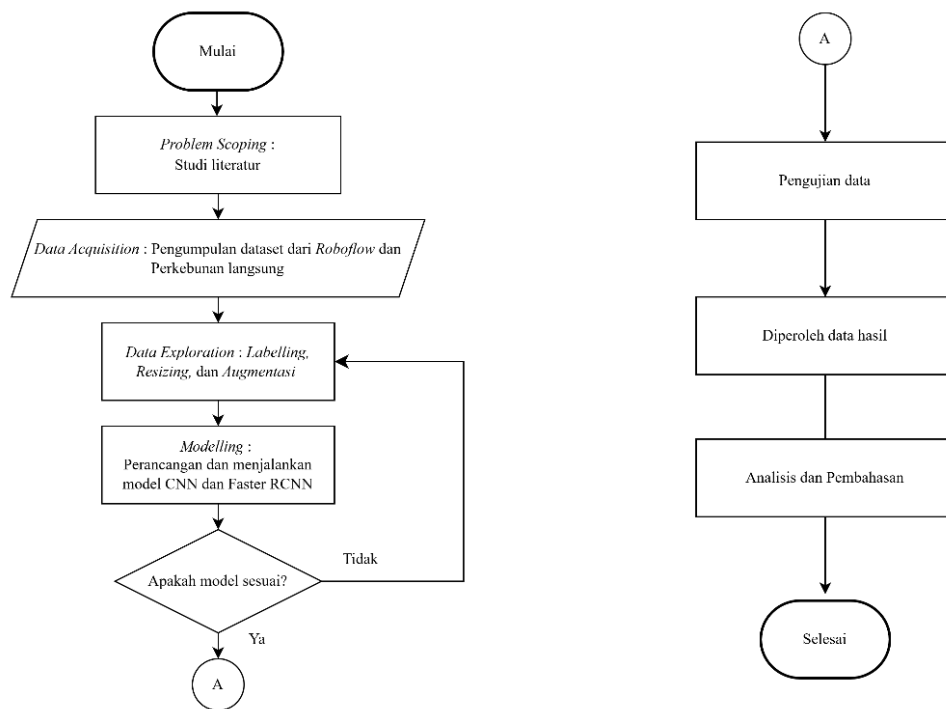
3.2 Alat dan Bahan Penelitian

Adapun alat dan bahan yang digunakan dalam penelitian ini adalah sebagai berikut :

1. Satu buah handphone Android Samsung A05, yang digunakan untuk mengambil citra tandan buah segar kelapa sawit di Pesawaran, Lampung.
2. Satu laptop Lenovo ideapad Slim 5 Pro dengan prosesor Ryzen 7 yang digunakan untuk pelatihan model dan *software* untuk mendeteksi objek.
3. *Software Roboflow* yang digunakan untuk eksplorasi data.
4. *Software Google Colab* yang digunakan untuk melatih model.
5. *CNN* dan *Faster CNN* dengan *backbone ResNet50* yang digunakan untuk mendeteksi dan mengklasifikasi tingkat kematangan tandan buah segar kelapa sawit.

3.3 Diagram Alir Penelitian

Adapun tahapan yang dilakukan pada penelitian ini dijelaskan melalui diagram alir pada gambar 3.1



Gambar 3. 1 Diagram Alir Penelitian

Diagram alir ini menggambarkan tahapan pengembangan sistem klasifikasi tingkat kematangan tandan buah segar kelapa sawit menggunakan model CNN dan *Faster RCNN*. Proses diawali dengan tahap *Problem Scoping*, yaitu studi literatur untuk memahami masalah serta menentukan solusi yang relevan. Selanjutnya, dilakukan *Data Acquisition*, yaitu pengumpulan dataset dari *platform* seperti *Roboflow* maupun dari Perkebunan secara langsung. Data yang telah terkumpul kemudian dieksplorasi melalui tahap *Data Exploration*, yang mencakup pelabelan, pengubahan ukuran (*resizing*), dan augmentasi data untuk meningkatkan kualitas dataset. Setelah itu, tahap *Modelling* dilakukan dengan merancang dan menjalankan model *CNN* serta *Faster RCNN*. Model yang telah dibuat akan dievaluasi untuk memastikan apakah performanya sesuai dengan kebutuhan. Jika performanya belum memadai, proses kembali ke tahap eksplorasi data dan pembangunan model untuk dilakukan perbaikan. Namun, jika model telah sesuai, sistem dilanjutkan ke tahap Pengujian Data, dimana model diuji menggunakan data tertentu untuk mendapatkan hasil yang diharapkan. Data hasil pengujian tersebut kemudian dianalisis pada tahap Analisis dan Pembahasan guna memperoleh wawasan dan

kesimpulan terkait performa sistem. Proses ini diakhiri dengan tahap Selesai, yang menjadi penutup dari pengembangan sistem.

3.4 Prosedur Penelitian

Dalam perancangan sistem pengolahan citra ini dilakukan Langkah-langkah kerja sebagai berikut :

3.4.1 Studi literatur

Tahap awal dari penelitian ini adalah melakukan studi literatur. Studi literatur bertujuan untuk mengidentifikasi masalah, mencari sumber-sumber yang relevan dengan penelitian, serta menyediakan landasan teori sebagai acuan dalam melakukan penelitian. Studi literatur dilakukan melalui tinjauan literatur yang berfokus pada klasifikasi tandan buah kelapa sawit menggunakan algoritma *CNN* dan *Faster R-CNN* dengan data masukan berupa gambar. Pencarian literatur dilakukan melalui platform *Google Scholar* dan *ResearchGate* dengan menggunakan kata kunci yang tepat. Kata kunci tersebut dibagi menjadi lima kategori, yaitu: a) “*classification*”, b) “*CNN*”, c) “*Faster R-CNN*”, d) “*productivity*”, dan e) “*Oil Palm Fruit Bunch*”.

Kriteria pemilihan literatur yang digunakan adalah sebagai berikut:




- a. Literatur merupakan publikasi ilmiah.
- b. Literatur membahas tentang klasifikasi tandan buah segar kelapa sawit.
- c. Literatur berasal dari publikasi ilmiah berbahasa Indonesia dan Inggris.
- d. Literatur menggunakan pendekatan berbasis citra (*image-based*).

3.4.2 Pengumpulan data

Kegiatan pengumpulan data dilakukan dengan mengumpulkan data yang diperlukan dalam pembuatan desain program dan data gambar tandan buah segar kelapa sawit. Data input yang digunakan berupa gambar tandan buah segar kelapa sawit yang didapat dari platform open dataset yaitu *Roboflow*. Data gambar yang diambil dari *Roboflow* akan digunakan sebagai data *training*, dimana seluruh data berekstensi .jpg/.jpeg dengan jumlah total data sebanyak 8.400 gambar. Kemudian data gambar tandan buah segar kelapa sawit yang didapat dari Perkebunan digunakan sebagai data *validation* dengan jumlah total data sebanyak 625 gambar.

Contoh gambar tandan buah segar kelapa sawit yang digunakan pada penelitian ini dapat dilihat pada Tabel 3.1

Tabel 3. 1 Data Citra Tandan Buah Segar Kelapa Sawit

No	Gambar	Tingkat Kematangan
1		Matang
2		Mengkal
3		Mentah

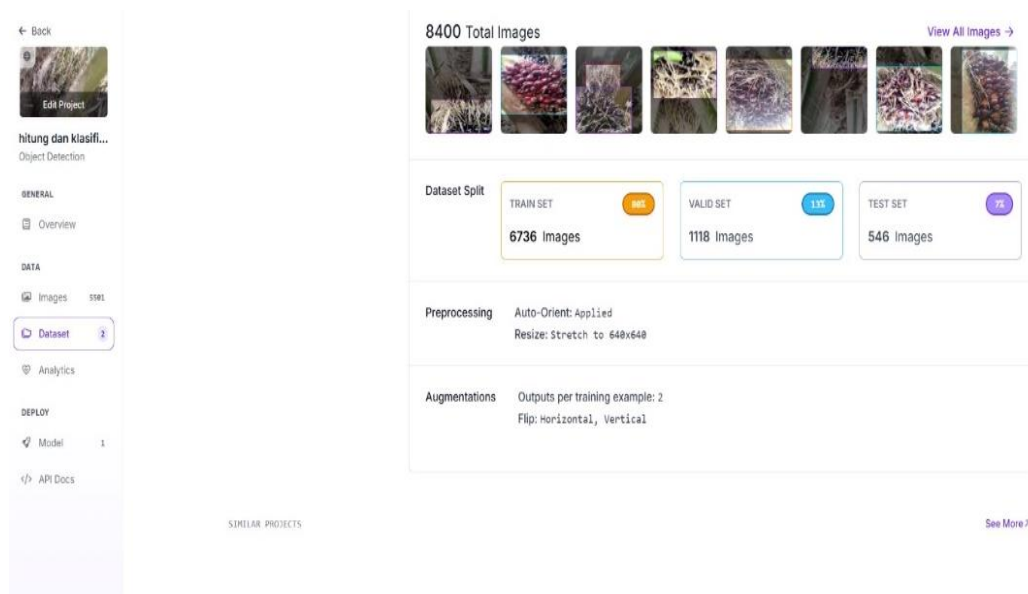
3.5 Perencanaan Program

Proses perencanaan program dilakukan menggunakan *Google Colaboratory*, dengan mengacu pada berbagai referensi seperti *GitHub*, *StackOverflow*, *YouTube*, dan sumber lainnya. Pengembangan program dibagi menjadi dua bagian utama, yaitu implementasi menggunakan metode *CNN* dan metode *Faster R-CNN*. Tujuan dari pengembangan ini adalah untuk mengevaluasi performa kedua metode dalam

mengklasifikasikan tingkat kematangan tandan buah segar kelapa sawit secara otomatis.

3.6 Data Acquisition

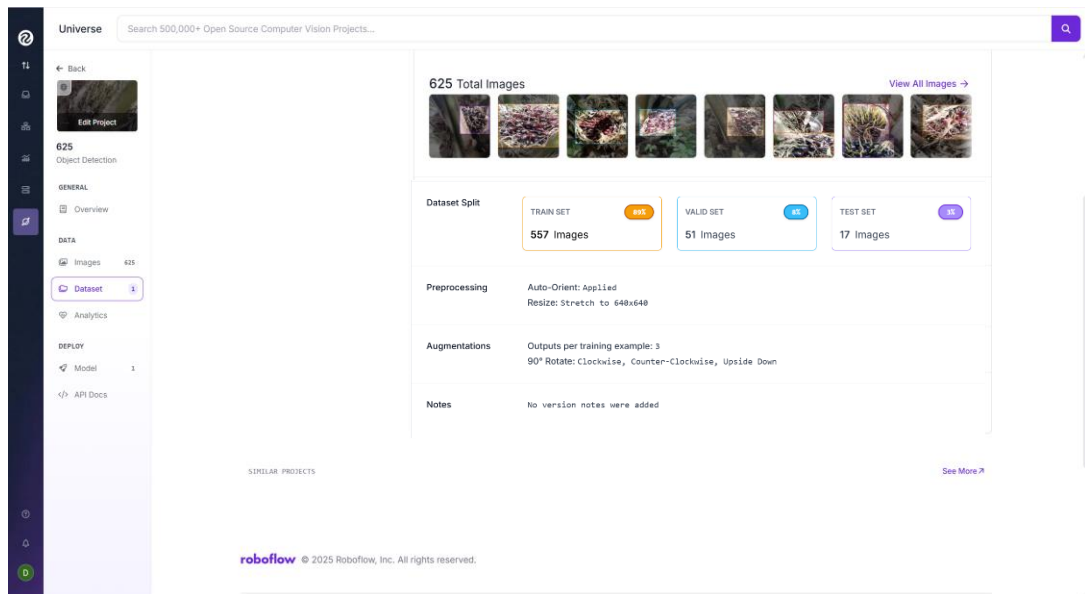
Langkah awal dalam penelitian ini adalah data acquisition. Pada tahap ini, data gambar kelapa sawit dikumpulkan sebagai input awal. Penelitian ini memanfaatkan gambar tandan buah segar kelapa sawit dengan berbagai tingkat kematangan, yaitu mentah, mengkal dan matang. Berikut merupakan dataset *Roboflow* dengan jumlah 8.400 yang telah dikumpulkan.



Gambar 3. 2 Data *Roboflow* 8400

Gambar 3.2 merupakan data yang diperoleh dari *platform open dataset Roboflow* dengan jumlah data 8.400 gambar dengan *splitting* data *train* set 80% dengan jumlah 6.736 data, *valid* set 13% dengan jumlah 1118 data dan *test* set 7 % dengan jumlah 546 data.

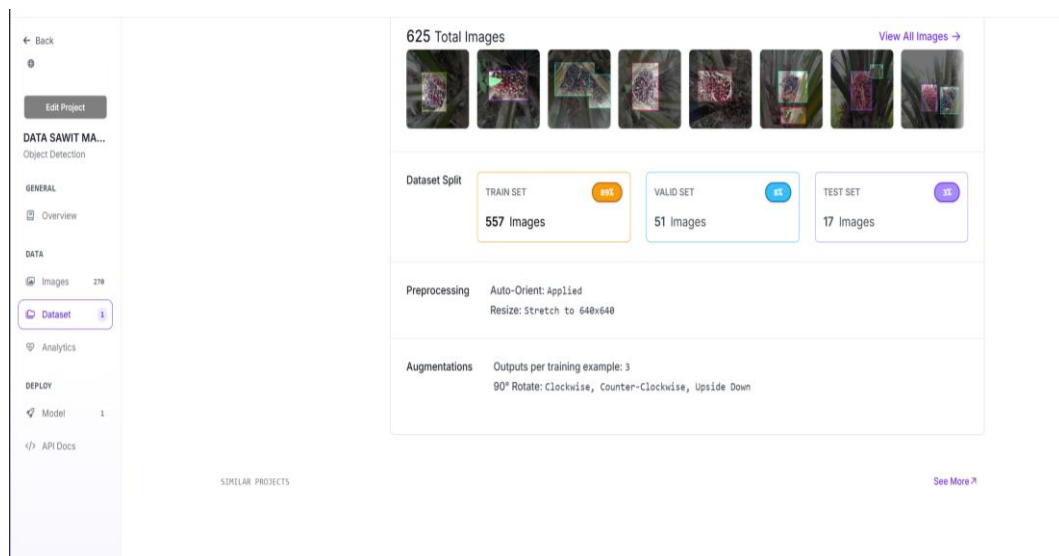
Dataset berikutnya merupakan data yang diperoleh dari *Roboflow* dengan jumlah 625 data, yang merupakan hasil pengurangan dari dataset sebelumnya menjadi 625 data.



Gambar 3. 3 Data Roboflow 625

Gambar 3.3 menampilkan data yang diperoleh dari *platform open dataset Roboflow* dengan total 625 gambar. Dataset tersebut dibagi menjadi *train set* sebesar 89% dengan jumlah 557 gambar, *validation set* sebesar 8% dengan jumlah 51 gambar, dan *test set* sebesar 3% dengan jumlah 17 gambar.

Kemudian data juga diambil dari lahan pertanian kelapa sawit milik seorang petani di Pesawaran. Berikut merupakan *dataset* Perkebunan yang diambil secara langsung dan dikumpulkan.

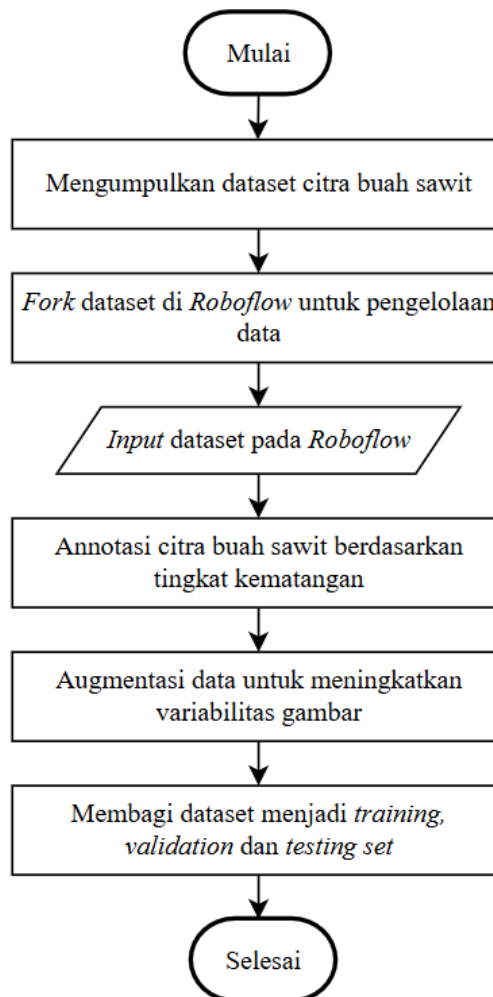


Gambar 3. 4 Data Perkebunan

Gambar 3.4 merupakan data perkebunan yang diambil dari lahan pertanian kelapa sawit milik seorang petani di Pesawaran dengan jumlah 625 gambar dengan *splitting* menjadi data *train* set 89% dengan jumlah data 557 data, *valid* set 8% dengan jumlah 51 data dan *test* set 3% dengan jumlah 17 data. Seluruh gambar berekstensi .jpg/.jpeg.

3.7 Data Exploration

Tahapan ini bertujuan untuk memproses dataset yang telah dikumpulkan. Pengolahan gambar dilakukan agar format setiap gambar memenuhi persyaratan yang dibutuhkan oleh algoritma deteksi yang digunakan. Proses eksplorasi data ini dapat dilihat pada diagram alir pada gambar 3.4 berikut.



Gambar 3. 5 Diagram Alir *Data Exploration*

Proses pengolahan data dalam penelitian ini diawali dengan pengumpulan citra digital buah kelapa sawit. Citra yang telah dikumpulkan kemudian diunggah ke *Roboflow*, di mana dilakukan proses *fork* dataset untuk mempermudah pengelolaan data. Setelah itu, dataset dimasukkan ke dalam *Roboflow* untuk dilakukan *cropping*, yang bertujuan meningkatkan jumlah sampel data yang digunakan dalam validasi. Setelah proses *cropping* selesai, citra diekspor dalam format *.jpg* atau *.jpeg* untuk digunakan dalam tahap selanjutnya.

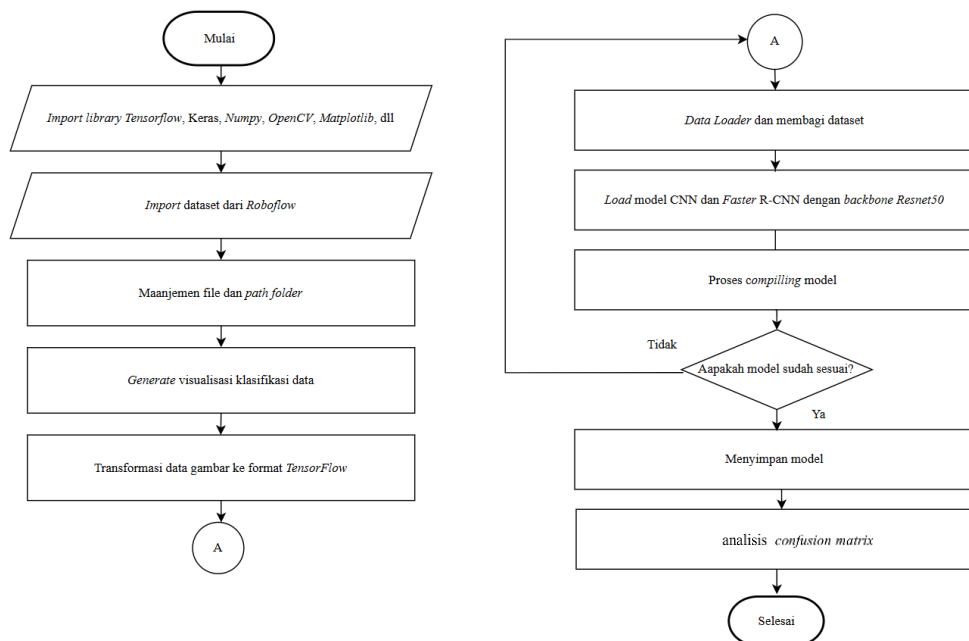
Selanjutnya, citra hasil *cropping* diproses lebih lanjut dalam *Roboflow*, di mana dilakukan anotasi dan pelabelan *bounding box* pada objek buah kelapa sawit. Setelah proses anotasi selesai, dataset dibagi menjadi *training set*, *validation set*, dan *test set* sesuai kebutuhan dalam proses pelatihan model. Untuk meningkatkan jumlah dan variasi dataset, dilakukan augmentasi data dengan menerapkan transformasi citra seperti *flip horizontal*, *flip vertikal*, dan *rotasi 90°*. Teknik augmentasi ini bertujuan untuk mengurangi risiko *overfitting* dan meningkatkan kemampuan model dalam mengenali berbagai variasi objek.

Pada tahap *preprocessing*, citra diubah ukurannya menjadi 640×640 piksel untuk memastikan konsistensi ukuran dataset serta mengoptimalkan efisiensi komputasi selama proses pelatihan. Jika citra yang dihasilkan tidak memenuhi rasio tersebut, latar belakang putih ditambahkan agar tetap proporsional. Proses *resizing* ini dilakukan menggunakan interpolasi citra guna memastikan kualitas gambar tetap optimal.

Dataset yang telah diproses ini kemudian digunakan dalam pelatihan model *Convolutional Neural Network* (CNN) dan *Faster R-CNN* dengan *backbone ResNet50*. CNN digunakan untuk klasifikasi tingkat kematangan buah kelapa sawit berdasarkan fitur warna dan tekstur, sedangkan *Faster R-CNN* diterapkan untuk deteksi serta identifikasi kematangan buah sawit dengan lebih akurat melalui penggunaan *Region Proposal Network* (RPN) dan fitur yang diekstraksi oleh *ResNet50*.

3.6 Modelling

Pada tahap selanjutnya merupakan *modelling*, dimana tahapan *modelling* ini dapat dilihat pada diagram alir berikut.



Gambar 3. 6 Diagram Alir Modelling

Tahapan pemodelan dimulai dengan mengimpor berbagai *library* penting seperti *TensorFlow*, *Numpy*, *OpenCV*, dan *Matplotlib* untuk membaca, memproses, serta melatih model deep learning. Setelah itu, dataset yang telah diimpor dari *Roboflow* dikelola dengan melakukan manajemen file dan path folder guna memastikan data tersusun dengan baik. Selanjutnya, dilakukan visualisasi klasifikasi data untuk memastikan bahwa anotasi dan *bounding box* pada citra buah sawit sudah sesuai sebelum diproses lebih lanjut. Setelah itu, dataset dikonversi ke dalam format *TensorFlow* untuk meningkatkan efisiensi pemrosesan dan mempercepat pelatihan model.

Pada tahap selanjutnya, dataset dibagi menggunakan *DataLoader* agar dapat dikelompokkan menjadi *training set*, *validation set*, dan *test set* sehingga lebih optimal dalam proses pelatihan dan evaluasi model. Model yang digunakan dalam penelitian ini adalah *Convolutional Neural Network (CNN)* dan *Faster R-CNN*

dengan *backbone ResNet50*, yang di-load untuk proses klasifikasi dan deteksi objek buah kelapa sawit berdasarkan tingkat kematangannya.

Proses berikutnya adalah *compiling* model, yaitu tahap di mana model diuji dan dilakukan validasi awal untuk menentukan apakah parameter yang digunakan sudah sesuai. Jika model belum menunjukkan hasil yang optimal, maka dilakukan penyesuaian parameter seperti jumlah *epoch*, ukuran *batch*, dan jenis *optimizer* untuk meningkatkan performa model. Setelah model sesuai dengan kriteria yang diharapkan, model tersebut disimpan untuk digunakan dalam proses prediksi lebih lanjut.

Tahap akhir dari pemodelan adalah analisis error menggunakan *confusion matrix*, yang melibatkan pengukuran metrik seperti *accuracy*, *precision*, *recall*, *F1-score* dan *mAP* guna mengevaluasi kinerja model dalam mengklasifikasikan dan mendeteksi tingkat kematangan buah sawit secara akurat.

3.8 Evaluation

Analisis data mencakup perbandingan performa model berdasarkan metrik yang telah ditentukan, seperti *accuracy*, *precision*, *recall*, *F1-score* dan *mAP*, untuk menilai seberapa baik masing-masing model dalam tugas klasifikasi. Hasil dari analisis ini kemudian divisualisasikan dengan menggunakan *confusion matrix*, yang memberikan gambaran tentang distribusi kesalahan prediksi antara kelas yang berbeda, serta grafik performa seperti kurva *ROC* atau grafik perbandingan metrik untuk memudahkan pemahaman kinerja model secara keseluruhan.

BAB V PENUTUP

5.1 KESIMPULAN

Adapun Kesimpulan pada penelitian ini adalah sebagai berikut:

1. Model *CNN* standar menunjukkan performa terbaik pada dataset perkebunan dengan akurasi 83,87%, *precision* 77,93%, *recall* 75,03%, *F1-score* 75,24%, dan nilai *mAP* sebesar 84,29%. Model ini bekerja efektif dalam klasifikasi satu objek per gambar, yakni satu tandan buah sawit, dan memiliki kecepatan pemrosesan yang tinggi, yaitu 464 FPS menjadikannya sangat cocok untuk implementasi langsung di lapangan dengan keterbatasan komputasi dan klasifikasi tingkat kematangan buah sawit pada sistem yang hanya memerlukan klasifikasi tunggal.
2. Model *Faster R-CNN ResNet50* menunjukkan performa terbaik pada dataset *Roboflow* 8.400 gambar dengan akurasi 92,72%, *precision* 100%, *recall* 91,73%, *F1 score* 95,63%, dan nilai *mAP* sebesar 86,25%. Berkat penggunaan *Region Proposal Network*, model ini mampu melakukan deteksi objek dengan presisi sempurna, bahkan pada variasi tingkat kematangan dan kondisi gambar yang beragam. Meskipun kecepatan pemrosesan hanya 396 FPS, performa akurasi dan presisinya yang tinggi menjadikannya sangat cocok untuk implementasi pada sistem yang memerlukan deteksi tingkat kematangan buah sawit secara detail dan multi-objek.
3. Ketika jumlah data disamakan menjadi 625 gambar, dataset *Roboflow* unggul pada *mAP* (93,94%) dan *precision* (92,12%), sedangkan dataset

4. perkebunan unggul pada *recall* (85,53%) dan kecepatan inferensi (464 FPS). Keunggulan dataset *Roboflow* disebabkan tingkat variasi kematangan buah dan keragaman gambar yang lebih tinggi dibanding dataset perkebunan, sehingga model dapat belajar mengenali pola objek secara lebih detail. Sementara itu, keunggulan dataset perkebunan pada *recall* menunjukkan kemampuan model untuk lebih adaptif pada kondisi lapangan yang nyata meskipun variasi datanya lebih terbatas.
5. Berdasarkan hasil penelitian, dapat disimpulkan bahwa pemilihan model terbaik sangat bergantung pada struktur dan kualitas dataset. *CNN* dasar sangat cocok diterapkan untuk klasifikasi citra dengan satu objek. Sebaliknya, model seperti *Faster R-CNN* sangat efektif pada data dengan banyak objek per gambar dan anotasi yang konsisten, seperti pada dataset *Roboflow*.

5.2 SARAN

Adapun saran yang dapat diberikan untuk pengembangan dan penelitian selanjutnya adalah sebagai berikut:

1. Diperlukan pengumpulan dataset yang lebih besar dan lebih bervariasi untuk meningkatkan generalisasi model terhadap kondisi nyata di lapangan.
2. Penelitian selanjutnya dapat mengeksplorasi arsitektur model lain seperti *EfficientNet*, *MobileNet*, atau kombinasi dengan teknik segmentasi untuk meningkatkan akurasi dan efisiensi sistem.
3. Disarankan untuk mengintegrasikan model klasifikasi ini ke dalam sistem berbasis perangkat keras seperti kamera atau *drone* untuk pengaplikasian langsung di perkebunan.
4. Pengembangan antarmuka pengguna berbasis *web* atau aplikasi *mobile* dapat mendukung penggunaan model ini oleh petani atau pengelola kebun secara lebih praktis.

DAFTAR PUSTAKA

- [1] United States Department of Agriculture (USDA), “Production – Palm Oil,” U.S. Department of Agriculture, Oct. 12, 2024. [Online]. Available: <https://fas.usda.gov/data/production/commodity/4243000>.
- [2] A. Nugroho, *Teknologi Agroindustri Kelapa Sawit*. Banjarmasin, Indonesia: Lambung Mangkurat Univ. Press, 2019, ISBN: 978-602-6483-97-3
- [3] J. Zulkarnain, Kusri, and T. Hidayat, “Klasifikasi tingkat kematangan tandan buah segar kelapa sawit menggunakan pendekatan deep learning,” *JST (Jurnal Sains dan Teknologi)*, vol. 12, no. 3, pp. 748–758, 2024, doi: 10.23887/jstundiksha.v12i3.59140
- [4] M. Subhan and H. Basri, “Klasifikasi Mutu Buah Pala (*Myristica Fragrans* Houtt) Berbasis Pengolahan Citra Menggunakan Metode Deep Learning Arsitektur Faster R-CNN,” *INTEK Jurnal Penelitian*, vol. 6, no. 2, pp. 106–113, 2019, doi: 10.31963/intek.v6i2.1566
- [5] S. K. Singh, V. Kumar, J. Yadav, and M. Sundararajan, “A Comparative Study of Different Architectural Models of CNN for Plant Leaf Disease Detection,” *International Journal of Computing Sciences Research*, vol. 7, pp. 2415–2430, Aug. 2023.
- [6] M. A. Syaharani, T. A. C. Budianto, and R. I. Adam, “Klasifikasi Buah Segar dan Busuk Menggunakan Algoritma Convolutional Neural Network (CNN),” *JATI (Jurnal Mahasiswa Teknik Informatika)*, vol. 8, no. 5, p. 10823, Oct. 2024.
- [7] S. Wan and S. Goudos, “Faster R-CNN for Multi-class Fruit Detection using a Robotic Vision System,” *Computer Networks*, 2019, doi: 10.1016/j.comnet.2019.107036
- [8] E. F. Himmah, M. Widyaningsih, and M. Maysaroh, “Identifikasi kematangan buah kelapa sawit berdasarkan warna RGB dan HSV menggunakan metode K-Means Clustering,” *J. Sains dan Inform.*, vol. 6, no. 2, pp. 193–202, 2020, doi: 10.34128/jsi.v6i2.242 [9] Pardamean, M., *Best Management Practice Kelapa Sawit*, 1st ed. Yogyakarta: ANDI, 2017, pp. xii+276. ISBN: 978-979-29-6609-1.

- [9] M. Pardamean, *Best Management Practice Kelapa Sawit*, 1st ed. Yogyakarta: ANDI, 2017, pp. xii+276, ISBN: 978-979-29-6609-1
- [10] Sulardi, *Budidaya Tanaman Kelapa Sawit*, 1st ed. Bekasi: PT Dewangga Energi Internasional, 2022, pp. 106. ISBN: 978-623-5927-23-7.
- [11] T. Ningsih, H. Gunawan, and J. F. Parhorasan, "Kajian Sistem Panen terhadap Potensi CPO (Crude Palm Oil) di Afdeling I Kebun Tanah Raja PT. Bakrie Sumatera Plantation, Tbk.," *Agro Estate*, vol. 5, no. 2, pp. 1–7, Dec. 2021.
- [12] BPK RI, "Pedoman Budidaya Kelapa Sawit (*Elaeis Guineensis*) Yang Baik," Peraturan BPK RI, Nov. 3, 2024. [Online]. Available: <https://peraturan.bpk.go.id/Details/160405/permentan-no-131permentanot140122013-tahun-20>
- [13] A. Abdullahi, J. K. Alhassan, and S. A. Bashir, "Document Classification in Higher Education Institutions Using Deep Learning: A CNN, RNN, and Hybrid CNN-RNN Approach," *FULafia-FSC Conference Proceedings*, vol. –, no. –, pp. 38–42, Jan. 2025. [Online]. Available: <https://lafiascijournals.org.ng/index.php/fscproceedings>. doi: [10.62050/fscp2024.462](https://doi.org/10.62050/fscp2024.462)
- [14] M. H. Diponegoro, S. S. Kusumawardani, and I. Hidayah, "Tinjauan Pustaka Sistematis: Implementasi Metode Deep Learning pada Prediksi Kinerja Murid," *Jurnal Nasional Teknik Elektro dan Teknologi Informasi*, vol. 10, no. 2, pp. 131–138, Mei 2021.
- [15] D. Sharma and U. Chauhan, "Human Activity Prediction Studies Using Wearable Sensors and Machine Learning," *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, vol. 7, no. 1, pp. 2456–3307, Jan. 2021. [Online]. Available: https://www.researchgate.net/figure/Difference-between-ML-and-DL_fig12_349643727
- [16] M. A. Mulya, N. Z. Arif, and N. Syefudin, "Tinjauan Pustaka Sistematis: Penerapan Metode Gabor Wavelet pada Computer Vision," *Journal of Computer Science and Technology*, vol. 1, no. 2, pp. 83–88, Mei 2023.
- [17] S. R. Sulistiyanti, F. X. A. Setyawan, and M. Komarudin, *Pengolahan Citra: Dasar dan Contoh Penerapannya*, Edisi Pertama. Yogyakarta: Teknosain, 2016, pp. 1–114. ISBN: 978-602-6324-12-2.
- [18] A. Arnita, F. Marpaung, F. Aulia, N. Suryani, and R. C. Nabila, *Computer Vision dan Pengolahan Citra Digital*, Cetakan Pertama. Surabaya: Pustaka Aksara, 2022. ISBN: 978-623-8230-27-3.
- [19] H. Fitriyah and R. C. Wihandika, *Dasar-Dasar Pengolahan Citra Digital*:

- Warna, Segmentasi, dan Filter*. Malang: UB Press, 2021, pp. 1–170. ISBN: 978-623-296-252-1.
- [20] I. Yuadi, *Forensik Digital dan Analisis Citra*. Magetan: CV. AE Media Grafika, 2023. ISBN: 978-623-8384-06-8.
- [21] E. S. Wijaya and Y. Prayudi, “Integrasi Metode Steganografi DCS Pada Image Dengan Kriptografi Blowfish Sebagai Model Anti Forensik Untuk Keamanan Ganda Konten Digital,” *Seminar Nasional Aplikasi Teknologi Informasi (SNATi)*, 2015, pp. D11–D17.
- [22] S. Suwarno and E. Kurniawan, “Multi-Level Pooling Model for Fingerprint-Based Gender Classification,” *Matrik: Jurnal Manajemen, Teknik Informatika, dan Rekayasa Komputer*, vol. 22, no. 2, pp. 195–206, Mar. 2023. [Online]. Available: https://www.researchgate.net/figure/Typical-Pooling-Process-a-Original-Matrix-b-Result-Matrix_fig2_371689457
- [23] M. Toyib, T. D. K. Pratama, and I. Aqil, “Penerapan Algoritma CNN Untuk Mendeteksi Tulisan Tangan Angka Romawi dengan Augmentasi Data,” *Algoritma*, vol. 2, no. 3, pp. 108–120, Mei 2024. doi: 10.62383/algoritma.v2i3.69
- [24] Z. Li, F. Liu, W. Yang, S. Peng, and J. Zhou, “A Survey of Convolutional Neural Networks: Analysis, Applications, and Prospects,” *IEEE Trans. Neural Netw. Learn. Syst.*, pp. 1–21, 2021. doi: 10.1109/TNNLS.2021.3082850
- [25] M. A. Mulya, N. Z. Arif, and N. Syefudin, “Tinjauan Pustaka Sistematis: Penerapan Metode Gabor Wavelet pada Computer Vision,” *J. Comput. Sci. Technol.*, vol. 1, no. 2, pp. 83–88, Mei 2023.
- [26] S. Sena, “Pengenalan Deep Learning Part 7: Convolutional Neural Network (CNN),” *Medium*, Nov. 13, 2017. [Online]. Available: <https://medium.com/@samuelsena/pengenalan-deep-learning-part-7-convolutional-neural-network-cnn-b003b477dc94>
- [27] R. Cadena, “Convolutional Neural Network (CNN),” *Medium*, Jul. 7, 2017. [Online]. Available: <https://medium.com/@raycad.seedotech/convolutional-neural-network-cnn-8d1908c010ab>
- [28] S. A. Suha and T. F. Sanam, “A deep convolutional neural network-based approach for detecting burn severity from skin burn images,” *Mach. Learn. with Appl.*, vol. 9, Apr. 2022, p. 100371.
- [29] A. Prakash, “Machine Learning – Convolution for Image Processing,” *Francium Tech Blog*, May 16, 2018. [Online]. Available: <https://blog.francium.tech/machine-learning-convolution-for-image->

[processing-42623c8dbec0](#)

- [30] J. Moolayil, “Tuning and deploying deep neural networks,” in *Learn Keras for Deep Neural Networks*, 2019, pp. 137–159.
- [31] N. Sabri, “A Comparison between Average and Max-Pooling in Convolutional Neural Network for Scoliosis Classification,” *Int. J. Adv. Trends Comput. Sci. Eng.*, vol. 9, no. 1.4, pp. 689–696, Sep. 2020.
- [32] R. Roncancio, A. El Gamal, and J. P. Gore, “Turbulent flame image classification using Convolutional Neural Networks,” *Energy AI*, vol. 10, p. 100193, 2022.
- [33] O. A. Montesinos López, A. Montesinos López, and J. Crossa, *Multivariate Statistical Machine Learning Methods for Genomic Prediction*, Jan. 2022.
- [34] “Convolution Neural Network,” *Medium*, Feb. 26, 2024. [Online]. Available: <https://medium.com/almabetter/convolution-neural-network-b1b82525adcb>
- [35] “RCNN Review [1311.2524],” *Medium (Analytics Vidhya)*, May 2019. [Online]. Available: <https://medium.com/analytics-vidhya/rcnn-review-1311-2524-898c3148789a>
- [36] C. B. Murthy, M. F. Hashmi, N. D. Bokde, and Z. W. Geem, “Investigations of Object Detection in Images/Videos Using Various Deep Learning Techniques and Embedded Platforms—A Comprehensive Review,” *Applied Sciences*, vol. 10, no. 9, p. 3280, May 2020.
- [37] A. Banerjee, E. Kumar, and R. Megavath, “Learning optimal deep prototypes for video retrieval systems with hybrid SVM-softmax layer,” *International Journal of Data Science and Analytics*, Jun. 2024.
- [38] A. Makone, “Faster R-CNN,” *Medium*, Jun. 4, 2015. [Online]. Available: <https://ashutoshmakone.medium.com/faster-rcnn-502e4a2e1ec6>
- [39] W. Wu et al., “Detection and enumeration of wheat grains based on a deep learning method under various scenarios and scales,” *Journal of Integrative Agriculture*, vol. 19, no. 8, pp. 1998–2008, Aug. 2020, doi: 10.1016/S2095-3119(20)63310-4
- [40] P. C. S. Reddy, G. R. Kumar, and S. Yadala, “Image Recognition using Deep Residual Learning,” *International Journal of Research in Engineering and Science (IJRES)*, vol. 10, no. 5, pp. 115–121, May 2022. [Online]. Available: https://www.researchgate.net/publication/361362336_Image_Recognition_using_Deep_Residual_Learning
- [41] N. Kundu, “Exploring ResNet50: An In-Depth Look at the Model Architecture and Code Implementation,” *Medium*, Jan. 23, 2023. [Online].

Available: <https://medium.com/@nitishkundu1993/exploring-resnet50-an-in-depth-look-at-the-model-architecture-and-code-implementation-d8d8fa67e46f>

- [42] R. P. Astuti et al., “Vegetation classification algorithm using convolutional neural network ResNet50 for vegetation mapping in Bandung district area,” *JURNAL INFOTEL*, vol. 14, no. 2, pp. 146–153, 2022.
- [43] H2O.ai, “What is the Confusion Matrix?”, *H2O.ai Wiki*. [Online]. Available: <https://h2o.ai/wiki/confusion-matrix/>
- [44] J. Terven, D. M. Cordova-Esparza, A. Ramirez-Pedraza, and E. A. Chavez-Urbiola, “Loss Functions and Metrics in Deep Learning. A Review,” *arXiv*, Jul. 5, 2023. doi: 10.48550/arXiv.2307.02694