

**IMPLEMENTASI YOU ONLY LOOK ONCE v5 (YOLOv5) UNTUK
IDENTIFIKASI GENUS ANGGREK DI INDONESIA BERBASIS MOBILE**

(Skripsi)

Oleh

**MUHAMMAD IRFAN ARDIANSYAH
NPM 1917051034**



**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG
2023**

**IMPLEMENTASI YOU ONLY LOOK ONCE v5 (YOLOv5) UNTUK
IDENTIFIKASI GENUS ANGGREK DI INDONESIA BERBASIS MOBILE**

Oleh

MUHAMMAD IRFAN ARDIANSYAH

Skripsi

**Sebagai Salah Satu Syarat untuk Mencapai Gelar
SARJANA KOMPUTER**

Pada

**Jurusan Ilmu Komputer
Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung**



**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG
2023**

ABSTRAK

IMPLEMENTASI YOU ONLY LOOK ONCE v5 (YOLOv5) UNTUK IDENTIFIKASI GENUS ANGGREK DI INDONESIA BERBASIS MOBILE

Oleh

MUHAMMAD IRFAN ARDIANSYAH

Famili Orchidaceae (Anggrek), sebuah kelompok yang menonjol dalam kerajaan Plantae, menampilkan beragam bentuk, warna, dan ukuran bunga, masing-masing dengan karakteristik yang berbeda. Anggrek menampilkan berbagai bentuk, warna, dan ukuran bunga, masing-masing dengan ciri uniknya. Bentuk dan warna khas bibir atau labellum membedakan bunga anggrek dari tanaman lain. Penelitian ini bertujuan utama untuk mengidentifikasi genus anggrek berdasarkan bunganya menggunakan aplikasi seluler untuk mendeteksi dengan lebih mudah. You Only Look Once (YOLO) adalah jaringan saraf tunggal yang langsung memprediksi kotak pembatas dan probabilitas kelas dari gambar penuh dalam satu evaluasi. Dalam studi ini, YOLO-v5 digunakan untuk mengidentifikasi model terbaik untuk dataset. Beberapa putaran pelatihan dilakukan, dengan variasi ukuran gambar dalam setiap sesi. Setelah dikonversi untuk diintegrasikan ke dalam aplikasi seluler menggunakan perpustakaan TensorFlow Lite, model yang dihasilkan mencapai 86.6% mAP. Kinerja aplikasi dievaluasi menggunakan dataset dari Kaggle, yang menunjukkan tingkat akurasi sebesar 87.5%. Ini memungkinkan deteksi real-time melalui kamera ponsel, bahkan tanpa koneksi internet.

Kata kunci: Anggrek, Kecerdasan Buatan, Convolutional Neural Network, You Only Look Once v5, Android

ABSTRACT

IMPLEMENTATION OF YOU ONLY LOOK ONCE V5 (YOLOV5) FOR ORCHID GENUS IDENTIFICATION IN INDONESIA BASED ON MOBILE PLATFORMS

By

MUHAMMAD IRFAN ARDIANSYAH

The Orchidaceae family, a prominent group in the Plantae kingdom, features diverse flower shapes, colors, and sizes, each with unique characteristics. Orchids exhibit diverse shapes, colors, and sizes in their flowers, each with unique characteristics. The distinctive shape and color of the lip or labellum are unique features that differentiate orchid flowers from those of other plants. This research mainly focuses on recognizing the orchid genus through its flowers using a mobile app for easier detection. You Only Look Once (YOLO) is a single neural network that directly predicts bounding boxes and class probabilities from full images in one evaluation. This research, YOLO-v5 is used to find the best models for the dataset. Multiple rounds of training were conducted using different image sizes in each training. After conversion for implementation in a mobile app using the TensorFlow Lite library, the resulting model is 86.6% mAP. The performance of the application is evaluated by using datasets taken from Kaggle. It is found that the application can achieve an accuracy rate of 87.5%. It can perform real-time detection through a phone camera without an internet connection.

Keywords: Orchid, Artificial Intelligence, Convolutional Neural Network, You Only Look Once v5, Android.

Judul Skripsi : **IMPLEMENTASI YOU ONLY LOOK ONCE v5 (YOLOv5) UNTUK IDENTIFIKASI GENUS ANGGREK DI INDONESIA BERBASIS MOBILE**

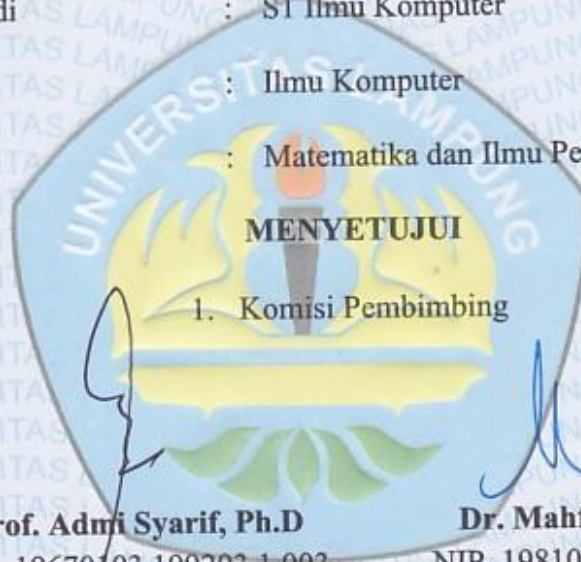
Nama Mahasiswa : **Muhammad Irfan Ardiansyah**

Nomor Pokok Mahasiswa : **1917051034**

Program Studi : **S1 Ilmu Komputer**

Jurusan : **Ilmu Komputer**

Fakultas : **Matematika dan Ilmu Pengetahuan Alam**



MENYETUJUI

1. **Komisi Pembimbing**

Prof. Admi Syarif, Ph.D
NIP. 19670103 199203 1 003

Dr. Mahfut, S.Si., M.Sc.
NIP. 19810909 201404 1 001

2. **Ketua Jurusan Ilmu Komputer**

Didik Kurniawan, S.Si., M.T.
NIP. 19800419 200501 1 004

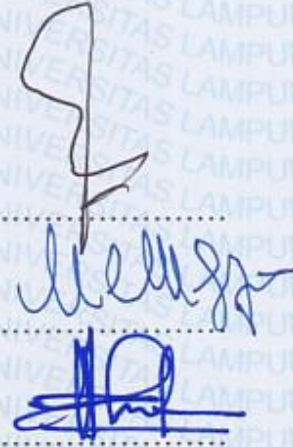
MENGESAHKAN

1. Tim Penguji

Ketua Penguji : Prof. Admi Syarif, Ph.D

Sekretaris Penguji : Dr. Mahfut, S.Si., M.Sc.

Penguji Utama : Dr. rer.nat Akmal Junaidi



2. Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam



Dr. Ing. Heri Satria, S.Si., M.Si.

NIP. 197110012005011002

Tanggal Lulus Ujian Skripsi: **18 Januari 2024**

PERNYATAAN

Saya yang bertanda tangan di bawah ini:

Nama : Muhammad Irfan Ardiansyah

NPM : 1917051034

Dengan ini menyatakan bahwa skripsi saya yang berjudul **“IMPLEMENTASI YOU ONLY LOOK ONCE v5 (YOLOv5) UNTUK IDENTIFIKASI GENUS ANGGREK DI INDONESIA BERBASIS MOBILE”** merupakan karya saya sendiri, bukan karya orang lain. Semua tulisan yang tertulis dalam skripsi ini telah mengikuti kaidah penulisan karya ilmiah Universitas Lampung. Jika dikemudian hari terbukti bahwa karya tulis ilmiah saya terbukti hasil menjiplak karya orang lain, maka saya siap menerima sanksi berupa pencabutan gelar yang saya peroleh.

Bandar Lampung, 13 Februari 2024



Muhammad Irfan Ardiansyah
NPM. 1917051034

RIWAYAT HIDUP



Lahir pada hari Jumat, 1 Juni 2001. Anak kedua dari Bapak Rusli Amd dan Ibu Nuraini ST, Penulis menyelesaikan pendidikan Sekolah Dasar (SD) di SD Negeri 3 Bandarjaya pada tahun 2012, Sekolah Menengah Pertama (SMP) di SMP Negeri 1 Seputih Banyak pada tahun 2016, Sekolah Menengah Atas (SMA) di SMA Negeri 1 Seputih Banyak pada tahun 2019.

Pada tahun 2019, penulis terdaftar sebagai mahasiswa Jurusan Ilmu Komputer Universitas Lampung melalui jalur SBMPTN. Kegiatan yang dilakukan penulis selama menjadi mahasiswa sebagai berikut.

1. Menjadi anggota Adapter Himpunan Mahasiswa Jurusan Ilmu Komputer pada periode 2019/2020.
2. Menjadi anggota Badan Khusus Himpunan Mahasiswa Jurusan Ilmu Komputer periode 2019/2020.
3. Menjadi anggota Badan Khusus Himpunan Mahasiswa Jurusan Ilmu Komputer periode 2020/2021.
4. Menjadi Asisten Dosen Dasar – Dasar Pemrograman Jurusan Ilmu Komputer pada periode 2020/2021.
5. Menjadi Asisten Dosen Pemrograman Terstruktur Jurusan Ilmu Komputer pada periode 2020/2021.
6. Menjadi Asisten Dosen Sistem Operasi Jurusan Ilmu Komputer pada periode 2021/2022.
7. Melaksanakan kegiatan Magang Bersertifikat (MSIB) angkatan 2 di Citiasia Inc. Divisi Apps Programmer.

8. Menjadi Asisten Dosen Pemrograman Mobile Lanjut Jurusan Ilmu Komputer pada periode 2022/2023.

MOTTO

“The end is important in all things.”

(Jhin)

“What you do in this world is a matter of no consequence. The question is what can you make people believe you have done”

(Arthur Conan Doyle)

“Impian manusia tidak akan pernah habis.”

(Marshall D. Teach)

PERSEMBAHAN

Alhamdulillah puji dan syukur kepada ke hadirat Allah SWT atas segala rahmat dan karunia-Nya sehingga dapat menyelesaikan skripsi ini dengan sebaik-baiknya.

Shalawat serta salam selalu saya sanjung agungkan kepada Nabi Muhammad SAW yang syafaatnya selalu senantiasa dinantikan di yaumul akhir kelak.

Kupersembahkan karya ini kepada
Kedua Orang Tuaku Tersayang

Yang selalu mendukung dan mendoakan dalam setiap langkah yang saya lalui. Saya ucapkan terimakasih sebesar-besarnya atas kasih sayang, didikan dan pengorbanan yang telah diberikan kepada saya dan tak akan mungkin bisa terbalaskan.

Seluruh Keluarga Besar Ilmu Komputer 2019

**Jurusan Ilmu Komputer, Fakultas Matematika dan Ilmu Pengetahuan
Alam, Universitas Lampung**

SANWACANA

Alhamdulillah, puji syukur kehadiran Allah SWT, atas berkat rahmat, karunia, hidayah, dan kesehatan yang diberikan sehingga penulis dapat menyelesaikan skripsi yang berjudul “IMPLEMENTASI YOU ONLY LOOK ONCE v5 (YOLOv5) UNTUK IDENTIFIKASI GENUS ANGGREK DI INDONESIA BERBASIS MOBILE”. Dalam penyusunan skripsi ini, penulis mendapatkan semangat, dukungan, dan bantuan dari berbagai pihak. Dalam melaksanakan penelitian dan pembuatan skripsi ini, sehingga pada kesempatan ini penulis ingin menyampaikan ungkapan terima kasih ini kepada

1. Kedua orang tua, Rusli Amd. Dan Nuraini ST. serta keluarga yang telah memberi dukungan dan motivasi kepada penulis untuk menyelesaikan skripsi dengan baik.
2. Kakak Aditya Riyaldie Pratama yang selalu memberikan semangat, dukungan, dan penghibur penulis.
3. Bapak Prof. Admi Syarif, Ph.D. selaku Dosen Pembimbing 1 yang telah membimbing sehingga penulis dapat menyelesaikan skripsi dengan baik.
4. Bapak Dr. Mahfut, S.Si., M.Sc. selaku Dosen Pembimbing 2 yang telah membimbing serta memberi masukan dalam proses pembuatan skripsi.
5. Bapak Dr. rer. nat. Akmal Junaidi, M.Sc. selaku Dosen Pembahas yang telah memberikan masukan dalam penelitian skripsi ini.
6. Bapak Dr. Eng. Suropto Dwi Yuwono, S.Si., M.T. selaku Dekan FMIPA Universitas Lampung.
7. Bapak Didik Kurniawan, S.Si., M.T. selaku Ketua Jurusan Ilmu Komputer Universitas Lampung.
8. Bapak Dr. rer. nat. Akmal Junaidi, M.Sc.. selaku Dosen Pembimbing Akademik.
9. Bapak dan Ibu Dosen Jurusan Ilmu Komputer FMIPA Universitas Lampung yang telah memberikan ilmu serta pengalaman semasa perkuliahan.
10. Ibu Ade Nora Maela dan seluruh staf di Jurusan Ilmu Komputer yang telah sabar membantu segala urusan administrasi di masa perkuliahan.

11. Mas Mohamad Zainudin yang telah memberikan nasihat dan masukan semasa penulis sedang ada masalah dan menjadi kakak yang baik selama dikampus.
12. Mas Ardi Nofalian yang selalu membantu menyediakan dan memberikan tempat untuk melaksanakan seminar serta mengerjakan skripsi.
13. Teman-teman Jurusan Ilmu Komputer Universitas Lampung angkatan 2019 yang senantiasa memberikan dukungan dan telah berjuang bersama menjalankan studi perkuliahan.
14. Anggie Tamara selaku pemberi semangat yang selalu mendukung dan membantu dalam proses pembuatan skripsi.
15. Raymond Faraz Yandika, Faishal Hariz Makaarim Gandadipoera, dan Hans Christian Herwanto selaku rekan seperjuangan di bangku perkuliahan.
16. Teman-teman Squad Awet Muda selaku rekan seperjuangan dan teman-teman yang sangat berarti di bangku perkuliahan.

Bandar Lampung, 13 Februari 2024

Muhammad Irfan Ardiansyah
NPM.1917051034

DAFTAR ISI

	Halaman
DAFTAR ISI	iii
DAFTAR TABEL	vi
DAFTAR GAMBAR	vii
I. PENDAHULUAN	1
1.1. Latar Belakang.....	1
1.2. Rumusan Masalah	3
1.3. Batasan Masalah.....	3
1.4. Tujuan Penelitian.....	4
1.5. Manfaat Penelitian.....	4
II. TINJAUAN PUSTAKA	5
2.1. Penelitian Terdahulu.....	5
2.2. Anggrek	8
2.2.1. Morfologi Anggrek	8
2.3. <i>Convolutional Neural Network (CNN)</i>	14
2.3.1. <i>You Only Look Once v5 (YOLO)</i>	17
2.3.2. <i>Hyperparameter</i>	23
2.4. <i>Confusion Matrix</i>	24
2.4.1. <i>Precision</i>	24
2.4.2. <i>Recall</i>	25
2.4.3. <i>Accuracy</i>	25
2.4.4. <i>F1-Score</i>	26
2.5. <i>Mean Average Precision (mAP)</i>	26
2.5.1. <i>Intersection over Union (IoU)</i>	27

2.6.	Aplikasi <i>Mobile</i>	28
2.7.	Android.....	29
2.8.	Dart.....	29
2.9.	<i>Framework flutter</i>	29
2.10.	Python.....	30
2.11.	Tensorflow.....	30
2.12.	Roboflow	31
2.13.	Visual Studio Code.....	31
III. METODOLOGI PENELITIAN		32
3.1.	Waktu dan Tempat Penelitian	32
3.2.	Perangkat Penelitian	33
3.2.1.	Perangkat Lunak (Software)	33
3.2.2.	Perangkat Keras (<i>Hardware</i>)	33
3.3.	Jenis dan Sumber Data	34
3.3.1.	Data Primer	34
3.3.2.	Data Sekunder	34
3.4.	Bahan.....	34
3.5.	Tahapan Penelitian	35
3.5.1.	Datasets	35
3.5.2.	<i>Pre-processing image</i>	36
3.5.3.	<i>Training and validation YOLOv5 model</i>	37
3.5.4.	<i>Converting Model to Mobile model</i>	38
3.5.5.	<i>Deploy Model to Mobile</i>	38
IV. HASIL DAN PEMBAHASAN.....		43
A.	Datasets.....	43
B.	<i>Pre-processing image</i>	43

C.	<i>Training and Validation YOLOv5 model</i>	44
D.	<i>Converting Model to Mobile model</i>	48
E.	<i>Deploy Model to Mobile</i>	51
V. SIMPULAN DAN SARAN		69
5.1.	Simpulan.....	69
5.2.	Saran	69
DAFTAR PUSTAKA		70

DAFTAR TABEL

Tabel	Halaman
Tabel 1 Penelitian Terdahulu	5
Tabel 2. Confusion Matrix	24
Tabel 3. Jadwal Penelitian.....	32
Tabel 4. Komposisi Dataset	34
Tabel 5. Kombinasi Hyperparameter	37
Tabel 6. Tabel Kebutuhan Fitur Pengembangan Aplikasi	38
Tabel 7. Pengujian Fungsionalitas Halaman Utama	41
Tabel 8. Pengujian Fungsionalitas <i>Realtime Detection</i>	42
Tabel 9. Pengujian Fungsionalitas <i>Image Dtection</i>	42
Tabel 10. Tabel perbandingan <i>precision</i> , <i>recall</i> , dan mAp tiap model berdasarkan perbedaan <i>image size</i>	45
Tabel 11. <i>Precision, Recall, F1-score YOLOv5 model</i>	47
Tabel 12. <i>Average Precision and Average Recall PyTorch model</i>	47
Tabel 13. <i>Average Precision and Average Recall TFlite model</i>	50
Tabel 14. Tabel pengujian deteksi aplikasi pada data test di datasets	56
Tabel 15. Ringkasan tabel pengujian deteksi aplikasi pada data test di datasets ..	60
Tabel 16. Tabel evaluasi pengujian deteksi aplikasi pada data test di datasets	61
Tabel 17. Tabel pengujian deteksi langsung aplikasi di lapangan	62
Tabel 18. Ringkasan tabel pengujian deteksi langsung aplikasi di lapangan	66
Tabel 19. Tabel evaluasi pengujian deteksi langsung di lapangan	66

DAFTAR GAMBAR

Gambar	Halaman
Gambar 1. Morfologi bunga anggrek genus <i>Paphiopedilum</i> (Olahan Pribadi). ...	10
Gambar 2. Morfologi bunga anggrek genus <i>Vanilla</i> (Olahan Pribadi).....	11
Gambar 3. Morfologi bunga anggrek genus <i>Dendrobium</i> (Olahan Pribadi).	11
Gambar 4. Morfologi bunga anggrek genus <i>Vanda</i> (Olahan Pribadi).	12
Gambar 5. Morfologi bunga anggrek genus <i>Cattleya</i> (Olahan Pribadi).	13
Gambar 6. Morfologi Anggrek Genus <i>Phalaenopsis</i> (Olahan Pribadi).....	14
Gambar 7. Arsitektur CNN (O’Shea & Nash, 2015).	15
Gambar 8. Proses Konvolusi CNN (O’Shea & Nash, 2015).	16
Gambar 9. ReLu <i>Activation</i> (O’Shea & Nash, 2015).....	16
Gambar 10. <i>Pooling Layer</i> (O’Shea & Nash, 2015).....	17
Gambar 11. Arsitektur <i>Single-Stage Detector</i> (Jocher <i>et al.</i> , 2020).	18
Gambar 12. Arsitektur jaringan YOLOv5 (Jocher <i>et al.</i> , 2020).	19
Gambar 13. Penerapan CSPNet ke ResNet dan DenseNet (Jocher <i>et al.</i> , 2020). .	20
Gambar 14. Arsitektur modul BottleneckCSP (Jocher <i>et al.</i> , 2020).....	20
Gambar 15. Struktur dari blok SPPF (Jocher <i>et al.</i> , 2020).	21
Gambar 16. Persamaan yang digunakan untuk menghitung kotak pembatas target. (a) Persamaan yang digunakan pada versi sebelumnya (YOLOv2, YOLOv3). (b) Persamaan yang digunakan di YOLOv5 (Jocher <i>et al.</i> , 2020).....	22
Gambar 17. Grafik fungsi aktivasi yang digunakan di YOLOv5 (Jocher <i>et al.</i> , 2020). (a) Grafik fungsi SiLU. (b) Grafik fungsi sigmoid.	22
Gambar 18. <i>Intersection over Union</i> (Padilla <i>et al.</i> , 2020).....	27
Gambar 19. Tahapan Penelitian (Pramestya, 2018).....	35
Gambar 20. Proses <i>image annotation</i>	36

Gambar 21. Prototipe tampilan <i>homescreen</i>	39
Gambar 22. Prototipe tampilan halaman <i>realtime detection</i>	40
Gambar 23. Prototipe halaman <i>image detection</i>	41
Gambar 24. Hasil <i>training model</i> dengan menggunakan <i>image size</i> 160x160	44
Gambar 25. Hasil <i>training model</i> dengan menggunakan <i>image size</i> 320x320	45
Gambar 26. Hasil <i>training model</i> dengan menggunakan <i>image size</i> 480x480	45
Gambar 27 Hasil <i>training model</i> dengan menggunakan <i>image size</i> 640x640	45
Gambar 29. Grafik hasil proses <i>training YOLOv5</i>	46
Gambar 30. <i>Confusion matrix</i> model YOLOv5 Pytorch	48
Gambar 31. Hasil konversi <i>model</i> ke <i>tflite model</i> dengan menggunakan <i>image size</i> 480x480.....	49
Gambar 32. <i>Confusion matrix YOLOv5 Tensorflow Lite</i>	49
Gambar 33. Perbandingan label dan hasil deteksi oleh model YOLOv5	50
Gambar 34. Contoh <i>output</i> deteksi anggrek genus <i>Cattleya</i>	52
Gambar 35. Contoh <i>output</i> deteksi anggrek genus <i>Dendrobium</i>	53
Gambar 36. Contoh <i>output</i> deteksi anggrek genus <i>Paphiopedilum</i>	53
Gambar 37. Contoh <i>output</i> deteksi anggrek genus <i>Phalaenopsis</i>	54
Gambar 38. Contoh <i>output</i> deteksi anggrek genus <i>Vanda</i>	55
Gambar 39. Contoh <i>output</i> deteksi anggrek genus <i>Vanilla</i>	55

I. PENDAHULUAN

1.1. Latar Belakang

Bunga adalah salah satu komponen aspek estetika yang merupakan bagian dari hidup manusia. Anggrek, tanaman hias yang sangat menarik sehingga banyak pula penggemarnya. Berbagai variasi bentuk, warna, dan ukuran dengan ciri-ciri yang unik menjadi daya tarik anggrek. Untuk mengungkapkan dan mengagumi keindahan anggrek, banyak orang yang merelakan waktunya tersita. Tanpa segan, sebagian dari mereka pun memetikinya guna menghias ruangan. Juga karena keindahannya itu pulalah, lingkungan tanpa sadar mengangkatnya menjadi bunga komersial (Darmono, 2008).

Anggrek adalah salah satu famili tumbuhan berbunga yang paling beragam dan tersebar luas serta merupakan salah satu dari sedikit famili tumbuhan yang memiliki profil yang dapat memberikan manfaat bagi konservasi tumbuhan dalam skala luas. Namun terdapat berbagai masalah tentang distribusi, kelangkaan, ancaman dan kepunahan anggrek dibandingkan sebelumnya, dan banyak spesies yang menghadapi ancaman setiap hari termasuk hilangnya habitat dan eksploitasi yang tidak berkelanjutan. Banyak kelompok konservasi tanaman internasional, termasuk *Orchid Specialist Group of the IUCN Species Survival Commission*, menggunakan anggrek sebagai spesies unggulan dalam perdebatan konservasi. Organisasi-organisasi ini mempunyai peran penting dalam melawan tren hilangnya keanekaragaman hayati dengan mempromosikan konservasi anggrek yang efektif, meningkatkan jaringan dan transfer teknologi, berinteraksi dengan pengambil keputusan, dan mendidik komunitas anggrek (Phillip J. et al., 2003).

Anggrek (Orchidaceae), merupakan familia terbesar dalam *kingdom Plantae*

dengan 800-1000 genera dan beranggotakan 20.000-30.000 spesies, 25.000 jenis anggrek diantaranya telah dideskripsikan (Schuttleworth *et al.*, 1970; Godo *et al.*, 2010; Gogoi *et al.*, 2012). Keanekaragamannya sebagian besar terpusat di kawasan tropis dan subtropis. Daerah sebaran anggrek luas, termasuk di Indonesia.

Anggrek memiliki variasi bentuk, warna, dan ukuran bunga yang bermacam-macam dengan ciri-ciri unik tersendiri. Bentuk dan warna bibir atau *labellum* merupakan keunikan bunga anggrek yang menjadi pembeda dengan tanaman lain. Bunga anggrek memiliki struktur dasar tiga sepal (kelopak) dan tiga petal (tajuk bunga). Keindahan bunga anggrek menjadikan bunga ini termasuk komoditas hortikultura yang mempunyai nilai ekonomi tinggi (Lestari & Santoso, 2010).

Untuk membedakan jenis anggrek satu dengan yang lain yaitu dapat dilihat dari warna, tekstur, kelopak bunga anggrek. Dengan mengetahui perbedaan tersebut seseorang dapat mengidentifikasi jenis suatu bunga anggrek. Namun secara umum jenis-jenis bunga anggrek memiliki banyak kemiripan dari segi warna, tekstur dan kelopak bunganya, hal ini yang menyebabkan seseorang dapat mengalami kesulitan dalam mengidentifikasi jenis bunga anggrek.

Metode yang digunakan untuk mengidentifikasi jenis anggrek dapat dilakukan secara manual dan otomatis. Metode manual dilakukan dengan pengamatan langsung dari seorang ahli. Metode ini memerlukan seorang ahli yang mengenal dan dapat membedakan jenis dari suatu tanaman anggrek. Sedangkan pengidentifikasian secara otomatis dilakukan dengan bantuan alat yang dapat mengambil citra suatu tanaman anggrek dan secara otomatis membedakan jenis suatu anggrek. Cara ini lebih efektif dan obyektif dalam upaya pengidentifikasian jenis suatu anggrek.

Pengidentifikasian jenis bunga anggrek pernah diteliti oleh (Pamungkas, 2019) dengan judul Implementasi Metode *Convolutional Neural Network* (CNN) Untuk Klasifikasi Jenis Bunga Anggrek. Penelitian ini dilakukan dengan menggunakan Metode *GLCM* dan *KNN* dengan data 90 gambar, yaitu 30 gambar anggrek

Phalaenopsis, 30 gambar Dendrobium, dan 30 gambar bunga lain, menghasilkan tingkat akurasi sebesar 80% dengan rata-rata 77%

Metode *You Only Look Once* (YOLO) diperkenalkan pertama kali oleh (Redmon *et al.*, n.d.) dalam judul *You Only Look Once : Unified, Real-Time Object Detection*. Dalam penelitiannya, selain arsitekturnya yang sederhana, dikatakan pula bahwa YOLO sangat cepat dalam mengidentifikasi objek dan akurasi rata-rata yang didapatkan mencapai 88% dalam *ImageNet 2012 Validation*.

Penerapan metode YOLO sampai saat ini belum banyak diterapkan dalam pengidentifikasian citra. Padahal, YOLO terbukti merupakan metode yang 2 lebih efisien dibandingkan dengan algoritma machine learning lainnya (Wang *et al.*, 2016). Maka dari itu, penelitian ini menggunakan metode YOLO untuk mendeteksi dan mengklasifikasi citra tanaman anggrek.

1.2. Rumusan Masalah

Berdasarkan latar belakang maka dapat disimpulkan bahwa rumusan masalah yang akan diangkat pada penelitian ini yaitu bagaimana membangun Sistem Pendeteksian Anggrek untuk membantu mengidentifikasi jenis-jenis anggrek di Indonesia melalui citra digital berbasis Android.

1.3. Batasan Masalah

Terdapat beberapa batasan masalah pada penelitian ini yaitu objek yang akan dideteksi dan diklasifikasi yaitu tanaman anggrek yang terdiri dari 6 genus. Citra yang digunakan yaitu citra berupa kondisi anggrek dalam keadaan berbunga. Citra yang digunakan terdiri dari berbagai *image size* yang berbeda.

1.4. Tujuan Penelitian

Tujuan yang ingin dicapai dalam penelitian adalah:

1. Mengevaluasi kinerja model YOLOv5 dengan melakukan perbandingan hasil model dengan menggunakan image size yang berbeda pada saat training.
2. Mengembangkan model identifikasi bunga anggrek dengan menggunakan metode YOLOv5.
3. Mengembangkan model YOLOv5 untuk diimplementasikan ke *mobile Android*.

1.5. Manfaat Penelitian

Adapun manfaat penelitian ini adalah sebagai berikut :

1. Penelitian ini menjadi media peneliti dalam menerapkan ilmu dan pengetahuan yang telah diperoleh selama masa perkuliahan.
2. Penelitian ini akan memudahkan pengguna dalam melakukan identifikasi bunga anggrek secara langsung.
3. Penelitian ini dapat menjadi sumber informasi terkait pengembangan aplikasi mobile Android khususnya untuk Framework Flutter pada Jurusan Ilmu Komputer.

II. TINJAUAN PUSTAKA

2.1. Penelitian Terdahulu

Penelitian terdahulu bertujuan untuk membandingkan penelitian yang sudah ada dengan penelitian yang akan dilakukan. Selain itu, penelitian terdahulu akan dijadikan sebagai acuan dalam upaya tinjauan pustaka terkait dengan penelitian yang dilakukan. Penelitian tersebut antara lain:

Tabel 1 Penelitian Terdahulu

No.	Judul Penelitian	Metode	Hasil Penelitian
1.	Ekstraksi Citra menggunakan Metode GLCM dan KNN untuk Identifikasi Jenis Anggrek (<i>Orchidaceae</i>) (Pamungkas, 2019)	Metode GLCM dan KNN	Software yang menyajikan nilai-nilai <i>Energy</i> , <i>Entropy</i> , <i>Contras</i> , dan Homogenitas dari citra bunga anggrek yang digunakan.
2.	<i>Multilabel Classification of Orchid Features based on Deep Learning</i> (Post, 2020)	Deep Learning	Software yang mendeteksi warna bunga, warna labellum, perkembangan tanaman, tekstur labellum, karakteristik labelum dan jumlah bunga dari sebuah anggrek.

No.	Judul Penelitian	Metode	Metode	Hasil Penelitian
3.	Implementasi <i>Convolutional Network</i> (CNN) Untuk Klasifikasi Jenis Bunga Anggrek (Zainuri & Pamungkas, 2020)	Metode <i>Neural</i>	Metode Waterfall	Software pendeteksian jenis anggrek berdasarkan dataset berjumlah 140 data
4.	<i>A Real-Time Detection Algorithm for Kiwifruit Defects Based on YOLOv5</i> (Yao <i>et al.</i> , 2021)		YOLO	Software deteksi kecacatan pada buah kiwi melalui algoritma YOLOv5
5.	Deteksi dan Klasifikasi Kerusakan Jalan Aspal Menggunakan Metode YOLO Berbasis Citra Digital (Pramestya, 2018)		YOLO	Sistem deteksi dan klasifikasi objek kerusakan jalan menggunakan metode YOLOv1-tiny.
6.	Implementasi YOLO pada Aplikasi Pendeteksi Senjata Tajam di Android (Liunanda <i>et al.</i> , 2020)	Algoritma	YOLO	Aplikasi deteksi dan klasifikasi senjata tajam menggunakan metode YOLOv3 berbasis Android.
7.	<i>Orchid types classification using supervised learning algorithm based on feature and color extraction</i> (Andono <i>et al.</i> , 2021)		Metode Penelitian	Sistem klasifikasi jenis anggrek menggunakan metode support vector machine (SVM), Naïve Bayes, and k-nearest neighbor algorithm berdasarkan ekstraksi warna

No.	Judul Penelitian	Metode	Hasil Penelitian
8.	Pendeteksian Dan Pengenalan Jenis Mobil Menggunakan Algoritma <i>You Only Look Once</i> Dan <i>Convolutional Neural Network</i> (Geraldly & Lubis, 2020)	YOLO	Sistem Pendeteksian Dan Pengenalan Jenis Mobil Menggunakan Algoritma YOLO dan CNN
9.	<i>Plant Disease Recognition Model Based on Improved YOLOv5</i> (Chen <i>et al.</i> , 2022)	YOLO	Sistem pendeteksian penyakit pada tumbuhan menggunakan YOLOv5 yang telah di improvisasi
10.	<i>Machine Learning Object Detection</i> Tanaman Obat Secara Real-Time Menggunakan Metode Yolo (<i>You Only Look Once</i>) (Maulana, 2021)	YOLO	Aplikasi Pendeteksi tanaman obat secara real-time dengan menggunakan algoritma YOLO

2.2. Anggrek

Anggrek (Orchidaceae) merupakan kelompok tumbuhan dengan keanekaragaman jenis yang tinggi. Dilaporkan bahwa dari total 26.000 jenis anggrek di dunia, kurang lebih 5.000 jenis berada di Indonesia. Variasi pada berbagai jenis anggrek secara morfologinya terdapat pada bentuk daun, batang, dan bunga (Mahfut *et al.*, 2021).

2.2.1. Morfologi Anggrek

Anggrek dilihat dari morfologinya mempunyai ciri yang sama, hanya lingkungan hidupnya saja yang berbeda tergantung habitat asalnya. Kondisi lingkungan banyak berpengaruh pada pertumbuhan dan pembungaan anggrek tersebut. Secara morfologi, tanaman anggrek terdiri dari daun, batang, akar, bunga, dan buah.

Daun anggrek terdiri dari bermacam bentuk seperti agak bulat, lonjong, sampai lanset dan ketebalan daun pada anggrek beragam dari tipis sampai berdaging, dan kaku. Batang anggrek mempunyai bentuk yang beraneka ragam mulai dari ramping, gemuk berdaging, dan ada yang menebal di bagian tertentu, dengan atau tanpa umbi sembu (*pseudobulb*), ukuran batang anggrek juga bervariasi dari yang tipis, sangat pendek, sangat besar, hingga memanjang. Akar anggrek pada umumnya berbentuk silindris, berdaging, lunak, mudah patah, ujung akar meruncing, licin, dan sedikit lengket. Bunga anggrek tersusun dalam bentuk karangan bunga, jumlah kuntum bunga pada satu karangan bervariasi dapat terdiri dari satu sampai banyak kuntum.

Terdapat lima bagian utama dari bunga anggrek yaitu sepal (daun kelopak), petal (daun mahkota), stamen (benang sari), pistil (putik), dan ovarium (bakal buah). Buah anggrek berbentuk kapsular, di dalamnya terdapat banyak biji berukuran sangat kecil dan halus seperti tepung. Biji-biji anggrek tersebut tidak memiliki *endosperm* (cadangan makanan) sehingga diperlukan tambahan nutrisi dari luar atau lingkungan sekitar dalam membantu perkecambahannya (Darmono, 2008).

2.2.1.1. Klasifikasi Anggrek

Klasifikasi anggrek menurut (Dressler & Dodson, 1960) adalah sebagai berikut :

Kingdom	: Plantae
Divisio	: Spermatophyta
Sub Divisio	: Angiospermae
Class	: Monocotyledonae
Ordo	: Orchidales
Familia	: Orchidaceae
Sub Familia	: Epidendroideae
Genus	: <i>Paphiopedilum</i> , <i>Vanilla</i> , <i>Dendrobium</i> , <i>Vanda</i> <i>Cattleya</i> , <i>Phalaenopsis</i>

a. Paphiopedilum

Anggrek genus *Paphiopedilum* memiliki bunga relatif besar dengan ukuran tinggi 9-10,5 cm dan lebar 8-10 cm. Secara utuh, bunga anggrek ini memiliki kombinasi warna yang cukup mencolok. Kelopak bunga bagian dorsal berwarna putih krem dengan pola urat kehijauan. Mahkota bunga membentang berwarna hijau hingga hijau-kekuningan dari pangkal hingga tengah, sedangkan pada bagian tengah hingga ujung berwarna merah-jambu keunguan. Bibir bunga berbentuk kantung berwarna coklat kehijauan hingga semburat kemerahan. Benang sari lembaran bulan sabit dengan ujung bagian samping runcing. Anggrek ini tumbuh alami pada habitat berupa hamparan semak hingga hutan berlereng pada ketinggian 1300-1600 m dpl (Mahfut *et al.*, 2021). Selengkapnya morfologi bunga anggrek genus *Paphiopedilum* ditunjukkan pada Gambar 1.



Gambar 1. Morfologi bunga anggrek genus *Paphiopedilum* (Olahan Pribadi).

b. Vanilla

Vanilla merupakan genus anggrek epifit yang memiliki habitus merambat dengan panjang batang mencapai 5 - 15 m. Batang tanaman berbentuk silindris berwarna hijau tua, dengan daun berbentuk bulat hingga lanset, berdaging, tersusun secara berseling (*alternate*). Bunga berwarna kuning kehijauan pucat, memiliki sepal dorsal dengan ukuran dan bentuk yang sama dengan petal, sementara sepal lateral berukuran lebih besar. Labelum (*lip*) berlobus 3, berwarna kuning, dan berbentuk seperti terompet. Bunga hanya mekar sekali dalam setahun, dan hasil penyerbukannya akan menghasilkan buah yang matang dalam jangka waktu 8 - 9 bulan. Buah berbentuk seperti polong, silindris, mengandung ribuan biji kecil berwarna hitam (Mahfut *et al.*, 2021). Selengkapnya morfologi bunga anggrek genus *Vanilla* ditunjukkan pada Gambar 2.



Gambar 2. Morfologi bunga anggrek genus *Vanilla* (Olahan Pribadi).

c. Dendrobium

Dendrobium memiliki daun kecil panjang hingga 5 cm daun pipih dan berwarna hijau kekuningan. Memiliki akar keras panjang dan dipangkal batang terdapat batang yang membuncit besar dan keras. Bunga berwarna putih berbentuk seperti merpati dan hidup menumpang pada pohon pohon besar seperti akasia, pohon, durian dan pohon rambutan (Purnamasari *et al.*, 2011). Selengkapnya morfologi bunga anggrek genus *Dendrobium* ditunjukkan pada gambar 3.



Gambar 3. Morfologi bunga anggrek genus *Dendrobium* (Olahan Pribadi).

d. Vanda

Vanda memiliki batang bundar, panjang dan kokoh. Tinggi tanaman dapat mencapai 2m, daun berbentuk pita agak melengkung dengan ujung daun rumpang bersudut tajam dengan lebar sekitar kurang lebih 3 cm dan panjang mencapai 45 cm, tersusun saling bergantian pada batang yang tumbuh tegak. Tandan bunga bisa mencapai 50 cm yang menyangga 10-20 kuntum bunga yang muncul dari ketiak daun, sepal dan petal berwarna dasar antara putih dan kuning dengan corak totol berwarna coklat hingga kuning, dengan totol-totol merah keunguan. Diameter bunga anggrek *Vanda* bisa mencapai 10 cm, bunga mampu bertahan hingga 20-25 hari. Bunga anggrek *Vanda* berbau harum, aroma harum ini sangat di pengaruhi oleh ketinggian tempat hidupnya, di dataran tinggi aromanya sangat kuat dan semakin turun ke dataran rendah aromanya akan semakin berkurang (Metusala, 2006). Selengkapnya morfologi bunga anggrek genus *Vanda* dapat dilihat pada gambar 4.



Gambar 4. Morfologi bunga anggrek genus *Vanda* (Olahan Pribadi).

e. *Cattleya*

Cattleya merupakan salah satu jenis anggrek yang memiliki banyak variasi, meliputi 113 spesies. Habitat asli *Cattleya* berasal dari Amerika Tengah dan Selatan termasuk Venezuela, Brasil, Peru, Meksiko, Guyana, dan Argentina. Anggrek ini termasuk tanaman epifit dan memiliki pseudobulb tebal yang dapat menyimpan banyak air dan cadangan makanan. Bunga anggrek *Cattleya* terbentuk pada pucuk tanaman. Jenis *Cattleya* berdaun satu memiliki 1–2 kuntum bunga yang berukuran besar, sedangkan jenis *Cattleya* berdaun 2–3 mempunyai 3–8 kuntum dengan ukuran kecil. Panjang tangkai bunga anggrek ini termasuk pendek. Bunga *Cattleya* memiliki diameter 5 hingga lebih dari 16 cm, memiliki daya tahan 1-2 minggu bila tidak dipotong, atau 3-4 hari bila digunakan sebagai bunga potong (Widiastoety, 2004). Selengkapnya morfologi bunga anggrek genus *cattleya* dapat dilihat pada gambar 5



Gambar 5. Morfologi bunga anggrek genus *Cattleya* (Olahan Pribadi).

f. *Phalaenopsis*

Phalaenopsis tersebar luas mulai dari Malaysia, Indonesia, Filipina, Papua, hingga ke Australia. Cara hidupnya secara epifit dengan menempel pada batang atau

cabang pohon di hutan-hutan. Anggrek bulan termasuk dalam tanaman anggrek monopodial yang menyukai sedikit cahaya matahari sebagai penunjang hidupnya. Daunnya berwarna hijau dengan bentuk memanjang. Bunganya memiliki mekar yang lama serta dapat tumbuh hingga diameter 10 cm lebih. Bunga anggrek ini berbentuk khas dan tersusun majemuk, muncul dari ketiak daun. Bunganya simetri bilateral. Helaian kelopak bunga (sepal) biasanya berwarna mirip dengan mahkota bunga (sehingga disebut petal). Helai mahkota bunga termodifikasi membentuk semacam lidah yang melindungi aksesoris yang membawa benang sari dan putik. Benang sari memiliki tangkai sangat pendek dengan dua kepala sari dan terlindungi oleh struktur kecil (Hsu *et al.*, 2018). Selengkapnya morfologi bunga anggrek genus *phalaenopsis* dapat dilihat pada gambar 6.

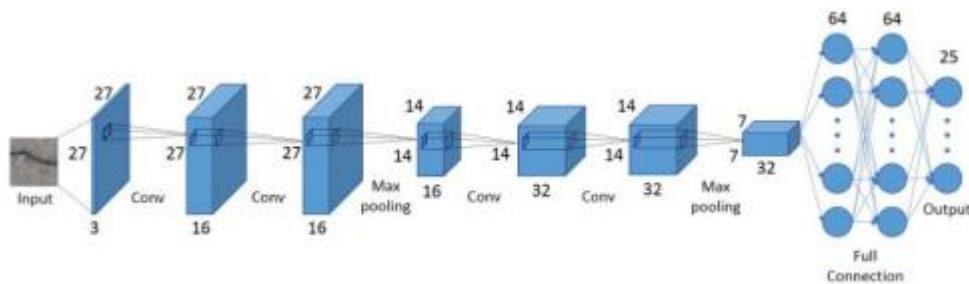


Gambar 6. Morfologi Anggrek Genus *Phalaenopsis* (Olahan Pribadi).

2.3. Convolutional Neural Network (CNN)

Convolutional Neural Network (CNN) adalah subdivisi dari algoritma *Deep learning* yang digunakan dalam visi komputer untuk menyelesaikan kasus atau masalah tertentu seperti klasifikasi objek dan pengenalan foto - foto, atau video (Rahim *et al.*, 2020). CNN terdiri dari tiga jenis layer, yaitu *convolutional layer*

(*Conv*), pooling layer (*Max pooling*) dan *fully-connected layer* (*Full Connection*). Tumpukan lapisan tersebut membentuk arsitektur dari CNN.



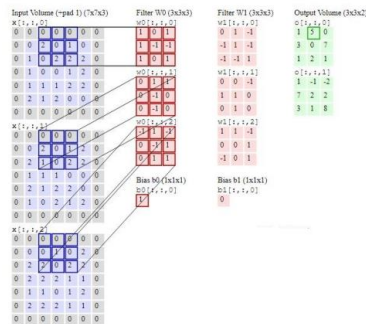
Gambar 7. Arsitektur CNN (O’Shea & Nash, 2015).

CNN dapat melatih dan menguji setiap citra yang masuk melalui beberapa proses dapat dilihat pada gambar 7 yang akan di jelaskan sebagai berikut :

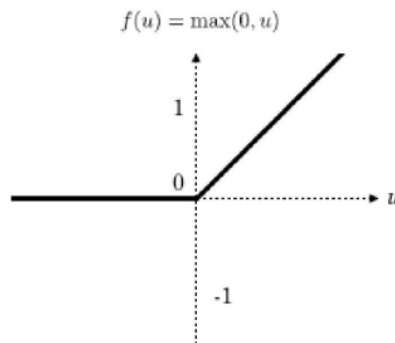
1. *Input layer*. *Layer* yang akan menyimpan nilai piksel dari citra masukan. Ukuran dari data *input* yaitu $27 \times 27 \times 3$, artinya 27×27 merupakan ukuran piksel citra dan 3 merupakan banyak *channel* citra, yaitu *Red*, *Green*, *Blue*.
2. *Convolutional layer*. *Layer* ini akan menentukan keluaran *neuron* yang terhubung ke *input layer* melalui perhitungan skalar produk antara bobot dan daerah yang terhubung dengan *input*. Ilustrasi dari proses dalam *convolutional layer* dapat dilihat pada gambar 8. Pada ilustrasi tersebut digunakan 1 *zero padding*, yaitu penambahan 1 baris nilai nol disepanjang garis batas *input*. Dalam proses konvolusi juga digunakan *Rectified Linear Unit* (biasa disingkat menjadi ‘ReLU’) bertujuan untuk menerapkan fungsi aktivasi ‘elementwise’ seperti sigmoid ke *output* dari aktivasi yang dihasilkan oleh lapisan sebelumnya, selanjutnya grafik ReLu, dapat dilihat pada gambar 9.
3. *Pooling layer*. *Layer* ini akan melakukan *downsampling* di sepanjang dimensi spasial dari input yang diberikan, selanjutnya mengurangi jumlah parameter dalam aktivasi tersebut. *Pooling layer* mengoperasikan peta aktivasi ke seluruh input, dan menggunakan fungsi ”MAX”. Di sebagian besar CNN, *max-pooling layer* menggunakan kernel dengan dimensi 2×2 dengan stride 2 di sepanjang dimensi spasial input, artinya berpindah

sebanyak 2 langkah dalam pergerakan kernelnya. Hal ini mengakibatkan ukuran input turun sampai 25% dari ukuran aslinya. Ilustrasi dari proses ini disajikan pada gambar 10. Pada ilustrasi tersebut digunakan kernel dengan dimensi 2x2 dengan stride 1.

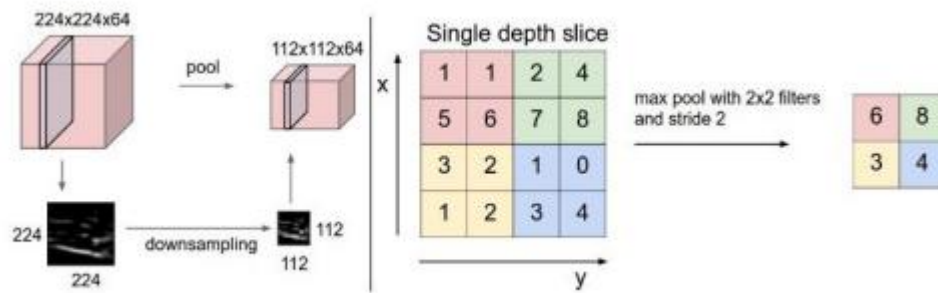
4. *Fully-connected layer*. Lapisan yang digunakan untuk melakukan transformasi pada dimensi data agar data dapat diklasifikasikan secara linear. Layer ini memiliki *hidden layer*, *activation function*, *output layer*, dan *loss function*



Gambar 8. Proses Konvolusi CNN (O’Shea & Nash, 2015).



Gambar 9. ReLu Activation (O’Shea & Nash, 2015).

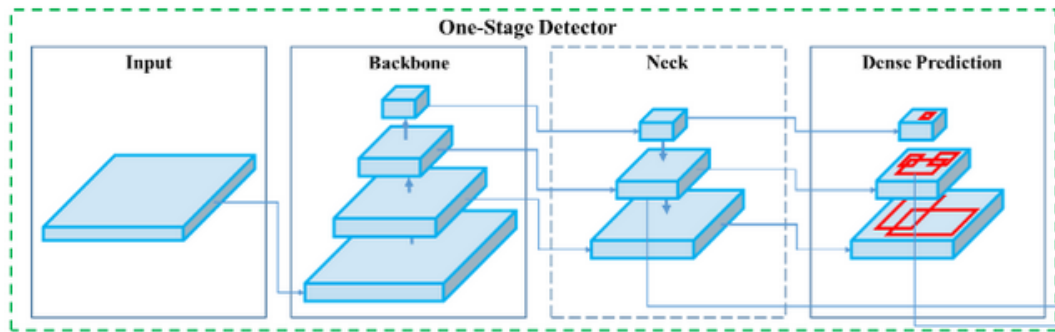


Gambar 10. *Pooling Layer* (O'Shea & Nash, 2015).

2.3.1. *You Only Look Once v5 (YOLO)*

YOLO adalah singkatan dari "You Only Look Once," yang merupakan pendekatan untuk deteksi objek dalam pengolahan citra dan video menggunakan deep learning. Metode ini pertama kali diperkenalkan oleh Joseph Redmon, Santosh Divvala, Ross Girshick, dan Ali Farhadi pada tahun 2016. Pendekatan YOLO memiliki ciri khas dalam mendeteksi objek dengan cepat dan efisien. Pendekatan YOLO berbeda dengan metode konvensional yang memerlukan beberapa tahap dalam proses deteksi objek, seperti ekstraksi fitur dan perhitungan tumpang tindih. Sebaliknya, YOLO menggabungkan semua tahapan ini dalam satu feedforward pass ke dalam jaringan saraf tiruan (neural network) yang dalam satu iterasi dapat menghasilkan daftar koordinat bingkai (bounding box) dan kelas objek yang terdeteksi.

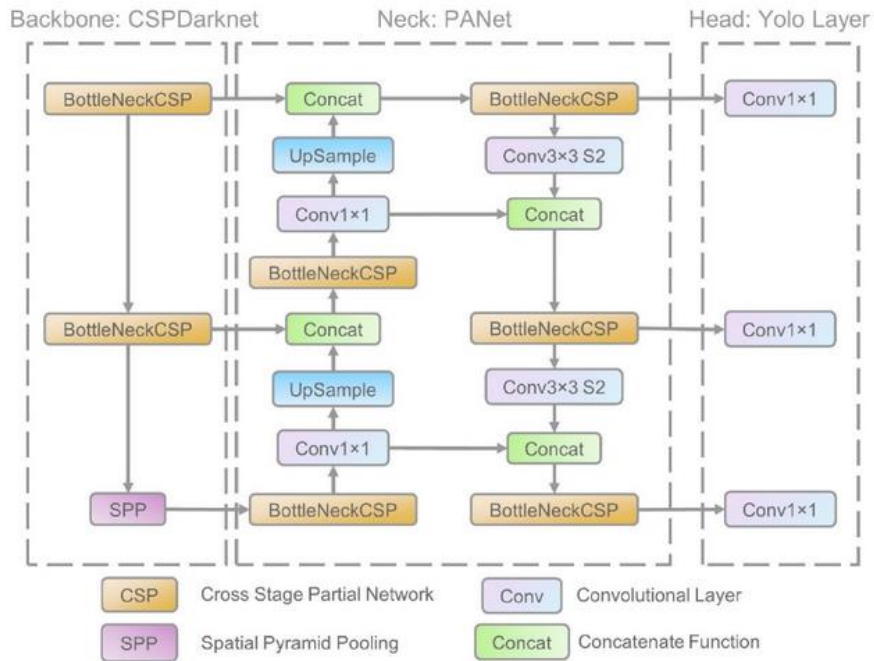
Arsitektur *You Only Look Once (YOLO)* merupakan arsitektur detektor objek satu tahap (*single-stage object detectors*). YOLO terdiri dari tiga komponen: *Backbone*, *Neck*, dan *Head (Dense Prediction)* untuk membuat prediksi padat (Jocher *et al.*, 2020). Arsitektur *Single-Stage Detector* ditunjukkan pada gambar 11.



Gambar 11. Arsitektur *Single-Stage Detector* (Jocher *et al.*, 2020).

1. *Backbone*, adalah jaringan terlatih yang digunakan untuk mengekstrak representasi fitur untuk gambar. Hal ini membantu mengurangi resolusi spasial gambar dan meningkatkan resolusi fitur (channel).
2. *Neck*, digunakan untuk mengekstraksi fitur piramida. Hal ini membantu model untuk melakukan generalisasi dengan baik objek dengan ukuran dan skala berbeda.
3. *Head (Dense Prediction)* digunakan untuk melakukan operasi tahap akhir. Ini menerapkan kotak jangkar pada peta fitur dan merender hasil akhir: kelas, skor objektivitas, dan kotak pembatas .

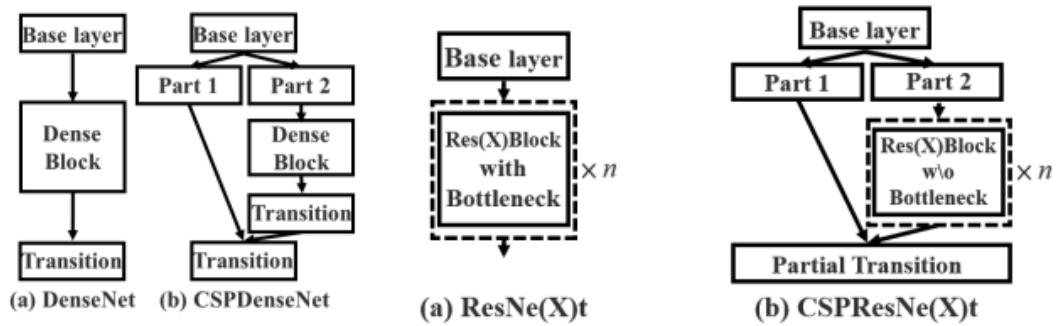
Semua model YOLOv5 terdiri dari 3 komponen yang sama yaitu, CSP-Darknet53 sebagai *backbone*, SPP dan PANet sebagai *neck* dan *head* yang sama yang digunakan pada YOLOv4. Ilustrasi arsitektur jaringan YOLOv5 dapat dilihat pada gambar 12.



Gambar 12. Arsitektur jaringan YOLOv5 (Jocher *et al.*, 2020).

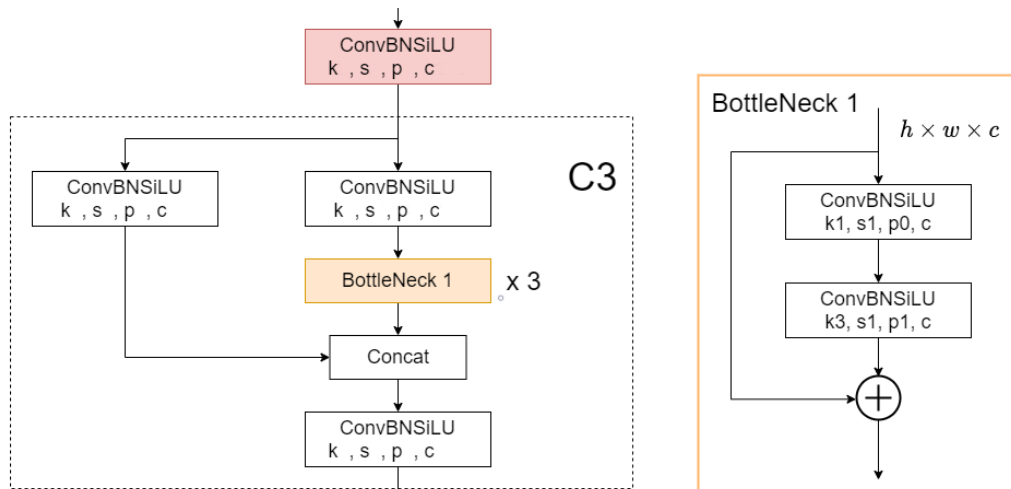
a. *Backbone* (CSP-Darknet53)

YOLOv5 menggunakan CSP-Darknet53 sebagai backbone. CSP-Darknet53 adalah jaringan konvolusional Darknet53 yang digunakan sebagai backbone pada YOLOv3 yang kemudian digunakan untuk menerapkan Cross Stage Partial Network (CSPNet). YOLOv5 menggunakan blok sisa dan padat untuk memungkinkan aliran informasi ke lapisan terdalam dan mengatasi masalah gradien hilang. Namun salah satu keuntungan menggunakan blok padat dan sisa adalah membantu masalah gradien yang berlebihan. CSPNet membantu mengatasi masalah ini dengan memotong aliran gradien (Jocher *et al.*, 2020). Ilustrasi penerapan CSPNet dapat dilihat pada gambar 13.



Gambar 13. Penerapan CSPNet ke ResNet dan DenseNet (Jocher *et al.*, 2020).

YOLOv5 menggunakan CSPNet untuk mempartisi peta fitur lapisan dasar menjadi dua bagian dan kemudian menggabungkannya melalui *cross-stage hierarchy* seperti yang ditunjukkan pada gambar 14.



Gambar 14. Arsitektur modul BottleneckCSP (Jocher *et al.*, 2020).

Penerapan ini memberikan keuntungan besar bagi YOLOv5, karena membantu mengurangi jumlah parameter dan membantu mengurangi jumlah komputasi penting yang mengarah pada peningkatan kecepatan inferensi yang merupakan parameter penting dalam model deteksi objek real-time (Jocher *et al.*, 2020).

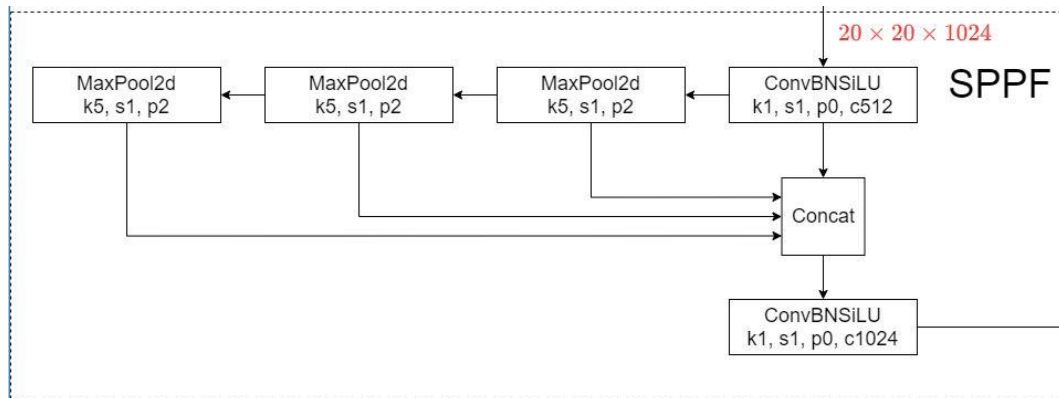
b. Neck (PANet)

YOLOv5 membawa dua perubahan besar pada model *neck*. Pertama yaitu, penggunaan varian *Spatial Pyramid Pooling* (SPP), dan kedua yaitu, *Path Aggregation Network* (PANet) telah dimodifikasi dengan memasukkan

BottleNeckCSP ke dalam arsitekturnya.

Path Aggregation Network (PANet) adalah jaringan piramida fitur, yang telah digunakan di YOLO versi sebelumnya (YOLOv4) untuk meningkatkan aliran informasi dan membantu pelokalan piksel yang tepat. Di YOLOv5, jaringan ini telah dimodifikasi dengan menerapkan strategi CSPNet seperti yang ditunjukkan pada gambar 13.

Spatial Pyramid Pooling (SPP), Blok Spatial Pyramid Pooling melakukan agregasi informasi yang diterima dari masukan dan mengembalikan keluaran dengan panjang tetap. Oleh karena itu, ia memiliki keuntungan dalam meningkatkan bidang reseptif secara signifikan dan memisahkan fitur konteks yang paling relevan tanpa menurunkan kecepatan jaringan. SPP telah digunakan di YOLO versi sebelumnya (YOLOv3 dan YOLO4) untuk memisahkan fitur terpenting dari backbone, namun di YOLOv5 menggunakan SPPF, yang merupakan varian lain dari blok SPP, untuk meningkatkan kecepatan jaringan. Struktur dari SPPF dapat dilihat pada gambar 15.



Gambar 15. Struktur dari blok SPPF (Jocher *et al.*, 2020).

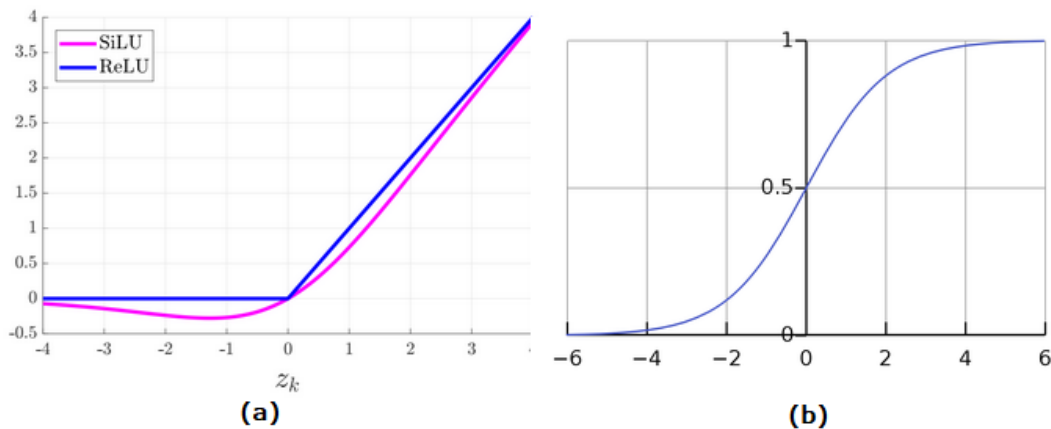
c. Head

YOLOv5 menggunakan *head* sama yang digunakan pada YOLOv3 dan YOLOv4. *Head* terdiri dari tiga lapisan konvolusi yang memprediksi lokasi kotak pembatas (x,y, tinggi, lebar), skor dan kelas objek. Persamaan untuk menghitung koordinat target kotak pembatas telah diubah dari versi sebelumnya, perbedaan dari persamaan yang digunakan ditunjukkan pada gambar 16.

$$\begin{array}{ll}
 b_x = \sigma(t_x) + c_x & b_x = (2 \cdot \sigma(t_x) - 0.5) + c_x \\
 b_y = \sigma(t_y) + c_y & b_y = (2 \cdot \sigma(t_y) - 0.5) + c_y \\
 b_w = p_w \cdot e^{t_w} & b_w = p_w \cdot (2 \cdot \sigma(t_w))^2 \\
 b_h = p_h \cdot e^{t_h} & b_h = p_h \cdot (2 \cdot \sigma(t_h))^2
 \end{array}
 \begin{array}{l}
 \text{(a)} \\
 \text{(b)}
 \end{array}$$

Gambar 16. Persamaan yang digunakan untuk menghitung kotak pembatas target. (a) Persamaan yang digunakan pada versi sebelumnya (YOLOv2, YOLOv3). (b) Persamaan yang digunakan di YOLOv5 (Jocher *et al.*, 2020).

Fungsi Aktivasi, memilih fungsi aktivasi sangat penting untuk model *deep learning* apa pun, untuk YOLOv5 penulis menggunakan fungsi aktivasi SiLU dan Sigmoid.



Gambar 17. Grafik fungsi aktivasi yang digunakan di YOLOv5 (Jocher *et al.*, 2020). (a) Grafik fungsi SiLU. (b) Grafik fungsi sigmoid.

SiLU adalah singkatan dari *Sigmoid Linear Unit* dan disebut juga fungsi *swish activation*. SiLU digunakan dengan operasi konvolusi di *hidden layer*. Sedangkan fungsi aktivasi Sigmoid digunakan dengan operasi konvolusi pada *output layer*.

YOLOv5 mengembalikan tiga keluaran: kelas objek yang terdeteksi, kotak pembatasnya, dan skor objektivitas. Oleh karena itu, YOLOv5 menggunakan BCE (*Binary Cross Entropy*) untuk menghitung *classes loss* dan *objectness loss*. Sedangkan CIoU (*Complete Intersection over Union*) *loss* digunakan untuk menghitung *location loss*.

2.3.2. *Hyperparameter*

Hyperparameter adalah konfigurasi yang berada diluar model. *Hyperparameter* membantu dalam menemukan parameter model yang tidak bergantung pada *data training* (Feurer et al., 2015). *Hyperparameter* yang digunakan yaitu *epoch*, *batch size*, *learning rate* dan *optimizer*.

- *Epoch*

Epoch menandakan satu siklus algoritma belajar dari *dataset* training, Satu epoch berarti sebuah algoritma telah belajar dari data training secara keseluruhan (Keskar et al., 2017).

- *Batch Size*

Batch size merupakan jumlah *training sample* yang digunakan dalam satu *iteration*. *Batch size* digunakan dalam proses *training* untuk menentukan jumlah contoh data yang dilatih dalam satu *forward* atau *backward pass*. *Batch size* mempengaruhi banyak memori yang dipakai (Keskar et al., 2017).

- *Optimizer SGD*

Optimizer SGD merupakan algoritma stokastik yang menggunakan konsep mirip dengan batching dengan membagi data training menjadi beberapa batch. SGD digunakan untuk membantu proses dalam melakukan optimization dan penggunaan resource hardware yang lebih optimum (Keskar et al., 2017).

- *Learning Rate*

Learning rate merupakan salah satu *hyperparameter training* yang ditetapkan untuk menghitung nilai koreksi bobot pada waktu proses *training* yang bertujuan untuk meningkatkan parameter tingkat pembelajaran mencapai hasil yang lebih baik (Astria et al., 2022).

2.4. Confusion Matrix

Confusion Matrix merupakan matriks yang digunakan untuk melakukan evaluasi model klasifikasi dengan memprediksi objek yang benar atau salah. *Confusion Matrix* berisikan informasi tentang klasifikasi aktual dan prediksi. *Confusion Matrix* direpresentasikan dengan tabel yang menjelaskan klasifikasi jumlah data uji yang benar dan klasifikasi jumlah data uji yang salah (Kohavi dan Provost, 1998).

Tabel 2. Confusion Matrix

<i>Correct</i>	<i>Classified as</i>	
<i>Classification</i>	<i>Predicted “+”</i>	<i>Predicted “-“</i>
<i>Actual “+”</i>	<i>True Positives</i>	<i>False Negatives</i>
<i>Actual “-“</i>	<i>False Positives</i>	<i>True Negatives</i>

Keterangan:

True Positives (TP) = data positif yang terdeteksi benar.

False Positives (FP) = data negatif yang terdeteksi sebagai data positif.

True Negatives (TN) = data negatif yang terdeteksi benar.

False Negatives (FN) = data positif yang terdeteksi sebagai data negatif.

Tabel *confusion matrix* bertujuan untuk menghitung kinerja metode klasifikasi dengan menghitung jumlah *recall*, *precision*, *accuracy*, dan *F1-score* sebagai perbandingan rata-rata *precision* dan *recall* yang dibobotkan.

2.4.1. Precision

Precision mengukur proporsi unit yang dikatakan model positif dan sebenarnya Positif. *Precision* memberi tahu kita seberapa besar kita dapat mempercayai model ketika memprediksi data sebagai positif (Grandini *et al.*, 2020).

$$\frac{(TP)}{(TP + FP)} \quad (1)$$

Keterangan:

TP_i = data positif yang terdeteksi benar pada kelas ke-*i*.

FP_i = data negatif yang terdeteksi sebagai data positif pada kelas ke-*i*, *i* = jumlah kelas.

2.4.2. *Recall*

Recall mengukur akurasi prediksi model untuk kelas positif. *Recall* mengukur kemampuan dari model untuk menemukan semua unit positif dalam kumpulan data (Grandini *et al.*, 2020).

$$\frac{(TP)}{(TP + FN)} \quad (2)$$

Keterangan:

TP = data positif yang terdeteksi benar pada kelas ke-*i*.

FN = data positif yang terdeteksi sebagai data negatif pada kelas ke-*i* = jumlah kelas.

2.4.3. *Accuracy*

Accuracy adalah perhitungan yang bertujuan untuk mengevaluasi kinerja dari suatu algoritma. *Accuracy* merupakan tingkat kedekatan antara nilai yang diprediksi oleh sistem dengan nilai yang sebenarnya (Harun *et al.*, 2023).

$$\frac{(TP + TN)}{(TP + FP + FN + TN)} \quad (3)$$

Keterangan:

TP_i = data positif yang terdeteksi benar pada kelas ke- i .

FP_i = data negatif yang terdeteksi sebagai data positif pada kelas ke- i .

FN_i = data positif yang terdeteksi sebagai data negatif pada kelas ke- i .

TN_i = data negatif yang terdeteksi benar pada kelas ke- ii = jumlah kelas.

2.4.4. F1-Score

F1 Score merupakan perbandingan rata-rata presisi (*precision*) dan *recall*. F1-score mencapai nilai terbaiknya pada 1 dan skor terburuk pada 0 (Harun *et al.*, 2023).

$$F1\ Score = 2 \times (Recall \times Precision) / (Recall + Precision) \quad (4)$$

Keterangan:

Precision = parameter untuk mengukur ketepatan dari suatu algoritma.

Recall = parameter yang digunakan untuk mengukur kelengkapan algoritma.

2.5. Mean Average Precision (mAP)

Mean Average Precision (mAP) adalah cara untuk melihat seberapa baik suatu sistem dalam menemukan sesuatu. *Mean Average Precision* melihat seberapa sering hal tersebut benar dan seberapa sering hal tersebut menemukan segala sesuatu yang seharusnya atau ukuran komprehensif yang menggabungkan *accuracy* dan *recall* dari sistem deteksi objek (Harun *et al.*, 2023).

Pencarian *Mean Average Precision* diperlukan, rumus dari *Mean Average Precision* adalah nilai *Average Precision* pada tiap kelas dibagi dengan jumlah kelas. *Average Precision* (AP) hanya mengambil nilai presisi yang relevan dan kemudian hasilnya dibagi dengan jumlah kelas yang dilibatkan (Geraldly & Lubis, 2020).

$$\text{Mean Average Precision} = \frac{1}{n} \sum_{k=1}^{k=n} AP_k \quad (5)$$

Keterangan:

n = jumlah kelas

AP = *Average Precision*

Mean Average Precision memberikan gambaran yang lebih komprehensif tentang performa deteksi objek dibandingkan menggunakan metrik seperti akurasi atau satu IoU saja. Cakup berbagai ambang batas IoU, mengukur bagaimana sistem deteksi berkinerja pada berbagai tingkat kecocokan antara prediksi dan ground truth (Harun et al., 2023).

2.5.1. *Intersection over Union (IoU)*

IoU (Intersection over Union) adalah parameter penting dalam evaluasi performa deteksi objek. IoU mengkuantifikasi tingkat tumpang tindih antara kotak pembatas prediksi *bounding box* yang dihasilkan oleh algoritma deteksi dan kotak pembatas referensi sebenarnya. Ilustrasi IoU dapat dilihat pada gambar 18.

$$IOU = \frac{\text{area of overlap}}{\text{area of union}} = \frac{\text{img alt="Diagram showing two overlapping bounding boxes (green and red) and their union (blue). The intersection area is shaded blue." data-bbox="599 601 714 733}}{\text{img alt="Diagram showing two overlapping bounding boxes (green and red) and their union (blue). The intersection area is shaded blue." data-bbox="599 601 714 733}}$$

Gambar 18. *Intersection over Union* (Padilla et al., 2020).

Evaluasi deteksi objek, IoU *threshold* adalah batas atau ambang yang digunakan untuk menentukan apakah suatu prediksi bounding box dianggap benar atau salah. IoU antara prediksi dan referensi melampaui ambang batas, maka prediksi tersebut

dianggap akurat dan jika tidak dianggap salah. Ambang batas IoU sering digunakan dalam penghitungan metrik evaluasi seperti *Mean Average Precision* (mAP), yang menilai ketepatan deteksi objek (Harun *et al.*, 2023).

Dengan kemampuan untuk menyesuaikan ambang batas Indeks atau IoU (Intersection over Union), pengguna memiliki kontrol atas tingkat tumpang tindih yang diinginkan untuk memandang suatu prediksi sebagai akurat. Biasanya, nilai ambang batas IoU sekitar 0,5 digunakan sebagai standar, mengharuskan setidaknya setengah dari area prediksi dan referensi untuk tumpang tindih agar diklasifikasikan sebagai benar. Namun, dalam praktiknya, nilai ambang batas IoU dapat bervariasi sesuai dengan aplikasi dan persyaratan khusus.

Pemilihan ambang batas IoU yang tepat sangat berpengaruh pada penilaian dan pemilihan algoritma deteksi, dan juga berperan dalam menentukan tingkat kesuksesan dalam pendeteksian objek (Harun *et al.*, 2023).

2.6. Aplikasi Mobile

Aplikasi *mobile* adalah perangkat lunak yang didesain untuk beroperasi di perangkat seluler, seperti *smartphone* dan tablet komputer. Aplikasi *mobile* berjalan pada perangkat seluler dan bertujuan untuk melaksanakan tugas-tugas tertentu bagi penggunanya. Penting bagi aplikasi seluler yang dikembangkan untuk memenuhi standar kualitas tertentu dan sesuai dengan tujuan penggunaannya. Penggunaan aplikasi seluler seringkali terkait erat dengan konteks, lokasi, dan waktu penggunaannya.

Salah satu keunggulan utama aplikasi *mobile* adalah memberikan pengguna kemudahan dalam mengakses informasi secara portabel. Aplikasi seluler memungkinkan pengguna untuk mendapatkan informasi terbaru tanpa terikat oleh batasan waktu dan lokasi, yang membuatnya sangat praktis dan efisien bagi pemilik perangkat seluler (Mazumder *et al.*, 2018).

2.7. Android

Android adalah sistem operasi *mobile* berbasis *Linux* yang dirancang untuk perangkat seluler layar sentuh seperti *smartphone* dan *tablet*. *Linux kernel* merupakan layer terbawah pada sistem operasi *Android stack*. *Kernel* ini mengatur beberapa hal seperti sistem operasi pada umumnya, yaitu manajemen proses, manajemen memori, *file system*, *Input Output*, *driver*, *networking*, serta *protection* dan *security*.

Sistem operasi Android menawarkan lingkungan pengembangan yang sangat fleksibel karena sifat open source-nya. Ini memberi kebebasan kepada para pengembang untuk membuat aplikasi secara gratis agar dapat beroperasi pada perangkat Android. Proses pengembangan aplikasi seluler Android memungkinkan penciptaan aplikasi pihak ketiga yang inovatif dan dinamis.

2.8. Dart

Dart adalah sebuah bahasa pemrograman yang secara khusus dirancang untuk mendukung pengembangan aplikasi *web* dan seluler dengan cepat. Bahasa pemrograman Dart menerapkan berbagai konsep pendekatan pemrograman berorientasi objek, termasuk kelas, pewarisan, abstraksi, enkapsulasi, dan polimorfisme, sehingga memungkinkan pengembang untuk bekerja dengan prinsip-prinsip ini dalam pengembangan aplikasi mereka (Swathiga *et al.*, 2021).

2.9. Framework flutter

Flutter adalah *framework* lintas platform yang menargetkan pengembangan aplikasi *mobile* berperforma tinggi. *Flutter* dirilis secara publik pada tahun 2016 oleh *Google*. Selain berjalan di *Android* dan *iOS*, aplikasi *flutter* juga dapat berjalan di *web*. *Flutter* menggunakan mesin rendering berperforma tinggi untuk merender

setiap komponen tampilan menggunakan miliknya sendiri.

Fitur *hot reload* di *Flutter* yang disebut sebagai *Stateful hot reload* dan merupakan faktor utama untuk meningkatkan siklus pengembangan. *Hot reload stateful* diimplementasikan dengan mengirimkan kode sumber yang diperbarui ke *Dart Virtual Machine (Dart VM)* yang sedang berjalan tanpa mengubah struktur bagian dalam aplikasi, sehingga transisi dan tindakan aplikasi akan terpelihara dengan baik setelah *hot reload*.

2.10. Python

Python adalah bahasa pemrograman yang digunakan untuk membuat aplikasi, perintah komputer, dan melakukan analisis data. Konstruksi *python* memungkinkan pengguna untuk menulis program yang jelas baik dalam skala kecil maupun besar.

Fitur terpenting dari *Python* adalah dukungannya beberapa paradigma pemrograman, termasuk berorientasi objek dan pemrograman imperatif. *Python* mendukung sistem tipe dinamis, manajemen memori otomatis, dan memiliki perpustakaan standar yang besar serta komprehensif. *Interpreters* untuk pemrograman *Python* adalah tersedia untuk banyak sistem operasi.

2.11. Tensorflow

Tensorflow merupakan library *Machine Learning* bersifat *open source* yang di develop oleh *Google* di tahun 2015. *Tensorflow* menggunakan data *graphs* untuk membuat model. Data *graphs* ini terdiri dari beberapa *node* yang merepresentasikan sebuah operasi matematik, dan memiliki koneksi antara *node* yang disebut dengan tensor Model yang dihasilkan oleh *Tensorflow* dapat di *export* dengan format *Flatbuffer* (.tflite) dan dijalankan di *mobile device* dengan menggunakan

Tensorflow Lite. *Tensorflow Lite* yang dikenalkan pada tahun 2017 oleh Google ini memiliki performa yang lebih baik dan binary size yang lebih kecil karena *kernels* yang lebih di *optimized*, *pre-fused activations*, serta jumlah *dependencies* yang lebih kecil.

2.12. Roboflow

Roboflow adalah sebuah aplikasi berbasis web yang menyediakan berbagai fitur yang memudahkan pengolahan data, terutama dalam bidang visi komputer dengan data berupa gambar. Fitur-fitur yang ditawarkan oleh Roboflow mencakup anotasi data, augmentasi gambar, pengecekan kualitas data, dan bahkan melakukan pelatihan data.

Dalam penelitian ini, Roboflow akan digunakan untuk anotasi data, pra-pemrosesan data, dan pemeriksaan kualitas data. Penggunaan Roboflow sebagai alat penelitian dipilih untuk memungkinkan pengolahan dan distribusi data yang otomatis sesuai dengan kebutuhan pengguna.

2.13. Visual Studio Code

Visual Studio Code adalah *software editor* yang *powerful*, tapi tetap ringan ketika digunakan. *Visual Studio Code* bisa dipakai untuk membuat dan mengedit *source code* berbagai bahasa pemrograman.

Fitur yang dimiliki *Visual Studio Code* dapat mempermudah proses pembuatan *source code* aplikasi seperti fitur *intelliSense* atau *auto complete*, *debugging*, *auto save* dan *hot exit*, dan dapat menambahkan ekstensi lainnya sehingga mempermudah proses pembuatan dan pengeditan *source code*.

III. METODOLOGI PENELITIAN

3.1. Waktu dan Tempat Penelitian

Waktu dan tempat penelitian adalah sebagai berikut:

Tabel 3. Jadwal Penelitian

Tahapan	2023																			
	Maret				April				Mei				Juni				Juli			
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
<i>Datasets</i>	█																			
<i>Pre-processing Image</i>	█	█	█																	
<i>Training and Evaluation YOLOv5 Model</i>			█	█	█	█	█	█												
<i>Converting model to mobile</i>									█											
<i>Deploy Model to Mobile Develop Mobile Application</i>											█	█	█	█	█	█				
<i>Deploy Model to Mobile Testing Mobile Application</i>																	█	█	█	█

Penelitian dilakukan pada semester genap 2022/2023 di Gedung Ilmu Komputer Jurusan Ilmu Komputer Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung dan di kediaman Bandar Lampung yaitu di Perumahan Arum Lestari Permai, Sukarame, Bandar Lampung.

3.2. Perangkat Penelitian

Penelitian ini akan menggunakan dua jenis alat, yaitu perangkat lunak dan perangkat keras. Berikut merupakan spesifikasi alat yang akan digunakan selama penelitian.

3.2.1. Perangkat Lunak (Software)

Dalam penelitian ini perangkat lunak yang digunakan adalah:

- a. Sistem Operasi *Windows 10 Pro 64-Bit* digunakan sebagai sistem operasi pada laptop untuk pengembangan model YOLOv5 dan aplikasi *mobile*.
- b. *Python* versi 3.9.16 dengan library *Tensor Flow 2.10* sebagai *tools* untuk melakukan *training* model YOLOv5.
- c. *Visual Studio Code* digunakan sebagai *tools* untuk melakukan proses mengembangkan kode aplikasi *mobile* dengan *Framework Flutter*.
- d. Sistem Operasi *Android* digunakan sebagai sistem operasi pada perangkat *Mobile* saat melakukan *testing*.
- e. *Roboflow* digunakan untuk proses *Image Annotation*.
- f. *Google Colab* digunakan untuk proses *training*, *validation*, dan *export* model.

3.2.2. Perangkat Keras (Hardware)

Dalam penelitian ini perangkat keras yang digunakan adalah:

- a. Laptop Acer E5-475G dengan spesifikasi RAM 8 GB, *Solid-state drive* 256GB, *Processor* Intel i5-7200U, dan *Graphics processing unit* Nvidia GeForce 940MX sebagai alat pendukung penelitian untuk proses pengolahan citra pada bunga anggrek dan implementasi pembuatan aplikasi berbasis *Mobile*.
- b. *Smartphone* Redmi Note 8 Pro digunakan untuk proses *testing* aplikasi.

3.3. Jenis dan Sumber Data

3.3.1. Data Primer

Data Primer didapatkan dari observasi dan wawancara langsung melalui fitur-fitur yang diperlukan dalam pengembangan aplikasi.

3.3.2. Data Sekunder

Data sekunder diperoleh dari studi literatur seperti buku, jurnal, artikel, dan situs resmi. Situs resmi menyediakan informasi mengenai penerapan *framework* Flutter, GitHub, dan studi serupa.

3.4. Bahan

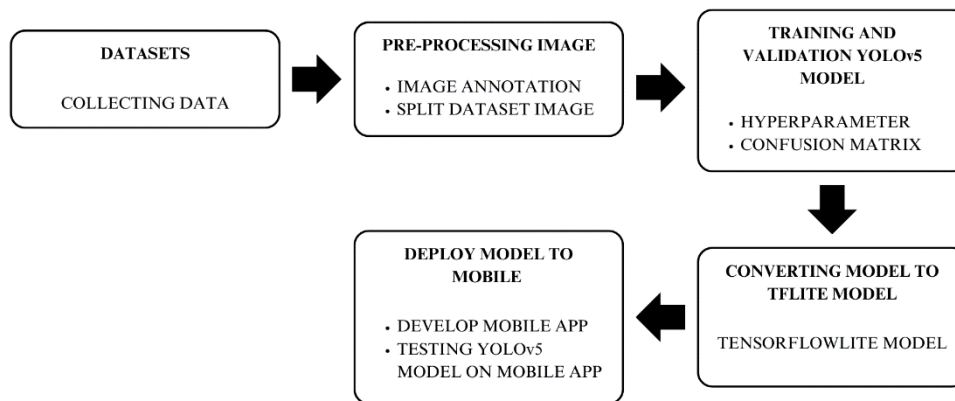
Bahan yang digunakan pada penelitian ini berjumlah 780 citra morfologi bunga anggrek yang didapatkan dari <https://www.kaggle.com/datasets/mikful/orchids>. Data citra tersebut terdiri dari 6 kelas yang berpedoman pada dari hasil eksplorasi dan sumbangan tumbuhan anggrek untuk pengayaan Kebun Raya Liwa. Kelas disesuaikan dengan jumlah anggrek terbanyak di Indonesia pada pedoman, penjabaran ke-6 *dataset* dijelaskan pada tabel 4.

Tabel 4. Komposisi Dataset

No.	Genus Bunga Anggrek	Jumlah Citra
1	Cattleya	130
2	Dendrobium	130
3	Paphiopedilum	130
4	Phalaenopsis	130
5	Vanda	130
6	Vanilla	130

No.	Genus Bunga Anggrek	Jumlah Citra
	Total Data:	780 Citra

3.5. Tahapan Penelitian



Gambar 19. Tahapan Penelitian (Pramestya, 2018)

Tahapan yang dilakukan pada penelitian ini dapat dilihat pada gambar 19 dan dijelaskan sebagai berikut:

3.5.1. Datasets

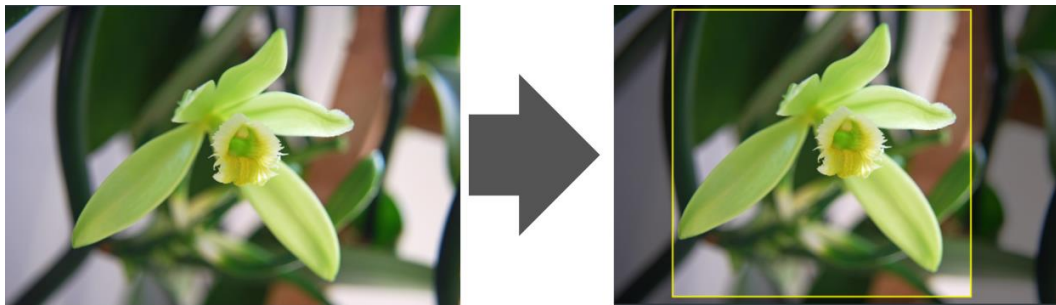
Dataset berguna untuk membuat mesin mengenal apa yang ada didalam gambar tersebut. *Dataset* diambil dan diolah melalui website kaggle yaitu <https://www.kaggle.com/datasets/mikful/orchids>. Dataset terpilih sebanyak 780 citra gambar bunga anggrek berdasarkan genus dengan 6 kelas yang terdapat 130 gambar ditiap kelasnya, kelas tersebut yaitu *cattleya*, *dendrobium*, *paphiopedilum*, *phalaenopsis*, *vanda*, dan *vanilla*.

3.5.2. *Pre-processing image*

Pre-processing adalah proses mengubah data mentah ke dalam bentuk yang lebih mudah dipahami. Tahapan ini dilakukan pada website *Roboflow* dengan melakukan proses *Image annotation* dan diakhiri dengan *Split dataset image*.

- *Image Annotation*

Image Annotation merupakan tahap dimana setiap citra pada dataset diberikan label dengan tujuan untuk menyimpan informasi citra. Proses label dilakukan dengan memberikan bounding box beserta nama kelas pada setiap objek citra. Hal ini bertujuan untuk meningkatkan performa model YOLO dalam pengenalan objek. Proses anotasi gambar dapat dilihat pada gambar 20.



Gambar 20. Proses *image annotation*.

- *Split Dataset Image*

Splitting dataset atau pemisahan dataset dilakukan dengan membagi data menjadi 3 bagian, yaitu data train, data valid, dan data test. Data train diperlukan sebagai bahan utama untuk melatih data. Bagian data train yang akan dipakai sebanyak 70% dari keseluruhan gambar yaitu sebanyak 90 gambar. Data valid dibutuhkan untuk membantu tahapan pelatihan data menggunakan data train. Proses pelatihan menggunakan data train juga memerlukan validasi untuk membuktikan kesamaan data yang dibaca oleh model. Data valid yang digunakan dalam penelitian ini sebanyak 15% dari keseluruhan data gambar yaitu sebanyak 20 gambar. Data testing digunakan untuk melakukan proses pengujian pada model yang ada. Data uji yang digunakan dalam penelitian ini sebanyak 15% dari jumlah keseluruhan data

gambar yaitu sebanyak 20 gambar.

3.5.3. *Training and validation YOLOv5 model*

- *Training YOLOv5 model*

Proses *Training* dilakukan dengan *dataset* sebanyak 780 citra morfologi bunga anggrek, dengan resolusi 160x160, 320x320, 480x480, dan 640x640 *pixel*. *Tools* yang digunakan dalam pelatihan model adalah *Google Colab* dengan menggunakan *runtime* pada T4 GPU dengan *Hyperparameter*. *Hyperparameter* yang akan digunakan dalam penelitian ini dapat dilihat dalam tabel 5.

Tabel 5. Kombinasi Hyperparameter

No	Nama Parameter	Nilai
1	<i>Epoch</i>	100
2	<i>Batch size</i>	16
3	<i>Optimizer</i>	SGD
4	<i>Learning rate</i>	0,01

Keterangan penggunaan *hyperparameter* :

Epoch berjumlah 100 untuk menguji dapatkah untuk menambah akurasi dengan pembelajaran yang lebih banyak. *batch size* utama adalah 16 untuk mempercepat proses *training* model yang akan dilakukan. Penggunaan *optimizer* SGD didasarkan pada *default optimizer* yang digunakan YOLOv5. Penggunaan *Learning rate* didasarkan pada *default learning rate* yang digunakan pada YOLOv5 yaitu sebesar 0.01. *Image Size* atau ukuran citra digunakan 160x160, 320x320, 480x480, dan 640x640 *pixel*, hal ini bertujuan untuk membandingkan tingkat akurasi model berdasarkan *image size* yang berbeda.

- *Validation YOLOv5 model*

Validation YOLOv5 model dilakukan dengan melihat *Confusion matrix*. *Confusion matrix* diperoleh ketika proses pelatihan model terhadap data training dan validasi telah selesai dilakukan. Kemudian pengujian model akan menggunakan data testing

untuk menunjukkan keberhasilan model yang telah dilatih sebelumnya. Hasil dari keberhasilan model dapat dianalisa menggunakan *confusion matrix* dengan menentukan besar *accuracy*, *f1-score*, *recall* dan *precision*.

3.5.4. *Converting Model to Mobile model*

Pada tahap ini akan dilakukan konversi model ke *Tensorflow Lite model*, proses ini dilakukan agar model dapat digunakan pada aplikasi *mobile* di sistem lokal tanpa perantara. Tahapan ini memerlukan *library Tensor Flow* pada *Python*.

3.5.5. *Deploy Model to Mobile*

Pada tahap ini dilakukan pembuatan aplikasi *mobile* yang dapat menjalankan model YOLOv5, aplikasi yang dimaksud adalah aplikasi *Android*. Tahapan ini akan dilakukan 2 proses yaitu *develop mobile application* dan *testing YOLOv5 model on mobile application*.

- *Develop mobile application*

Proses *develop mobile application* dilakukan dengan menggunakan *framework flutter*, proses ini membangun aplikasi *mobile* sederhana dengan fokus utama dapat menampilkan hasil dari deteksi model YOLOv5. Proses ini memerlukan *package flutter_vision* sebagai *plugin* untuk mengakses *Tensorflow Lite API*. Pengembangan diharuskan menjadikan aplikasi dapat dijalankan tanpa internet.

Tahapan ini dimulai dengan menentukan tujuan pembuatan aplikasi yaitu mempermudah pengamatan morfologi bunga anggrek untuk mempercepat proses pengamatan. Kebutuhan fitur pengembangan aplikasi dapat dilihat pada tabel 6.

Tabel 6. Tabel Kebutuhan Fitur Pengembangan Aplikasi

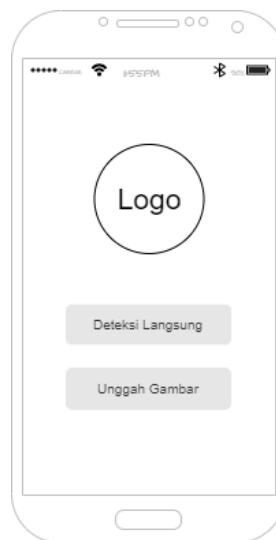
No	Fitur	Deskripsi
1	Deteksi kamera	Menampilkan kamera yang dapat menghasilkan

No	Fitur	Deskripsi
		<i>bounding box</i> pada gambar bunga anggrek
2	Identifikasi gambar	Menambahkan file gambar dan menampilkan hasil identifikasi pada gambar

Prototipe aplikasi dilakukan untuk mempermudah proses *develop mobile application* sehingga lebih terarah dan memiliki gambaran ketika proses dilakukan. prototipe aplikasi dirujuk dari kebutuhan fitur pengembangan aplikasi pada tabel 6.

Prototipe tersebut dapat dilihat pada prototipe tampilan halaman aplikasi sebagai berikut :

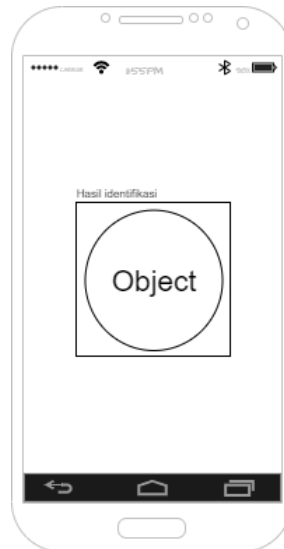
a. Prototipe tampilan halaman *Home Screen*



Gambar 21. Prototipe tampilan *homescreen*

Tampilan *Home Screen* pada gambar 21 merupakan halaman utama dalam aplikasi. Halaman ini berisi daftar menu pendeteksian yang menuju kepada halaman pendeteksian kamera secara *realtime* dan identifikasi melalui gambar.

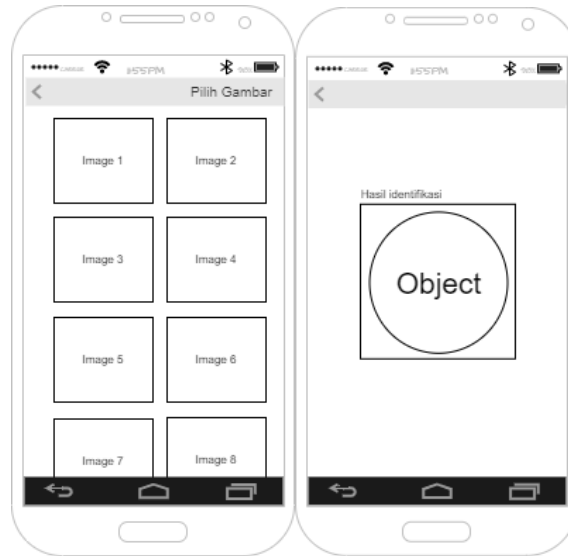
b. Prototipe tampilan halaman *realtime detection*



Gambar 22. Prototipe tampilan halaman *realtime detection*

Tampilan pada gambar 22 merupakan merupakan halaman pendeteksian menggunakan kamera dan *bounding box* pendeteksian secara *realtime*. Model yang akan digunakan pada pedeteksian kamera adalah model YOLOv5 yang telah di *export* menjadi *Tensorflow Lite Model*.

c. Prototipe tampilan halaman *image detection*



Gambar 23. Prototipe halaman *image detection*

Tampilan pada gambar 23 disisi kiri merupakan halaman *upload image* dan disisi kanan merupakan merupakan halaman hasil pendeteksian menggunakan *image* dan *bounding box* pendeteksian. Model yang digunakan pada pendeteksian *image* adalah model YOLOv5 yang telah di *export* menjadi *Tensorflow Lite Model*

- *Testing YOLOv5 model on mobile application*

Proses ini adalah tahap pengujian, pengujian yang dilakukan pada aplikasi akan memfokuskan kepada fungsional keluaran *input* dan *output* dari fungsi aplikasi menggunakan *User Acceptance Test (UAT)*. Tahap *testing* aplikasi dilakukan pada *operation system Android*, proses dilakukan untuk mengetahui apakah model dapat berkerja dengan baik saat melakukan pendeteksian. Skenario pengujian aplikasi dapat dilihat pada tabel 5-7:

Tabel 7. Pengujian Fungsionalitas Halaman Utama

No.	Test ID	Test Scenario
1	A-001	Pengguna masuk ke halaman utama aplikasi dan memiliki tugas untuk memilih menu <i>Image Detection</i> .

No.	Test ID	Test Scenario
2.	A-002	Pengguna masuk ke halaman utama aplikasi dan memiliki tugas untuk memilih menu <i>Realtime Detection</i> .
3.	A-003	Pengguna masuk ke halaman utama dan tidak memiliki tugas untuk memilih menu

Tabel 8. Pengujian Fungsionalitas *Realtime Detection*

No.	Test ID	Test Scenario
1	B-001	Pengguna melakukan deteksi langsung menggunakan kamera pada Android untuk melakukan deteksi pada objek berupa bunga anggrek.
2.	B-002	Pengguna melakukan deteksi langsung menggunakan kamera pada Android untuk melakukan deteksi pada objek selain bunga anggrek..

Tabel 9. Pengujian Fungsionalitas *Image Dtection*

No.	Test ID	Test Scenario
1	C-001	Pengguna melakukan unggah gambar berupa gambar bunga anggrek.
2.	C-002	Pengguna melakukan unggah gambar berupa gambar selain bunga anggrek.

V. SIMPULAN DAN SARAN

5.1. Simpulan

Simpulan yang diperoleh berdasarkan hasil penelitian yang telah dilakukan adalah sebagai berikut:

1. Penelitian ini membuktikan bahwa perbedaan penggunaan *image size* merujuk pada gambar 28. pada saat *training model* dapat mempengaruhi tingkat mAP dari model di YOLOv5.
2. Penelitian ini menghasilkan *model YOLOv5* dengan persentase mAp pada model YOLOv5 PyTorch mencapai 87% dan setelah dikonversi ke bentuk *Tensorflow Lite model* mencapai 86,6% pada 0.5 *threshold* dan mendapat nilai error sebesar 13.4%. Pada saat *testing* dengan *data test* pada *datasets* mendapat nilai akurasi sebesar 0.875 dan pada saat *testing* di lapangan mendapat nilai 0.80.
3. Penelitian ini membuktikan bahwa YOLOv5 dapat diimplementasikan pada aplikasi *mobile* untuk melakukan deteksi morfologi bunga anggrek.

5.2. Saran

Saran yang diberikan untuk penelitian selanjutnya adalah sebagai berikut:

1. Mengambil dataset secara pribadi agar dataset memiliki ukuran pixel yang sama.
2. Mengembangkan model arsitektur lainnya seperti *MobileNet*, *SSD MobileNet*, *EfficientNet*, dan yang lainnya.
3. Mengembangkan aplikasi *mobile* dengan paket lainnya seperti *pytorch* dan *tflite_flutter*.
4. Mengembangkan aplikasi *mobile* dengan *framework* lain seperti *Java framework*.

DAFTAR PUSTAKA

- Andono, P. N., Rachmawanto, E. H., Herman, N. S., & Kondo, K. (2021). Orchid types classification using supervised learning algorithm based on feature and color extraction. *Bulletin of Electrical Engineering and Informatics*, 10(5), 2530–2538. <https://doi.org/10.11591/eei.v10i5.3118>
- Astria, C., Windarto, A. P., & Damanik, I. S. (2022). Pemilihan Model Arsitektur Terbaik Dengan Mengoptimasi Learning Rate Pada Neural Network Backpropagation. *JURIKOM (Jurnal Riset Komputer)*, 9(1), 109–114.
- Bagli, E., & Visani, G. (2020). METRICS FOR MULTI-CLASS CLASSIFICATION: AN OVERVIEW. *A WHITE PAPER*, 1–17.
- Chen, Z., Wu, R., Lin, Y., Li, C., Chen, S., Yuan, Z., Chen, S., & Zou, X. (2022). Plant Disease Recognition Model Based on Improved YOLOv5. *Agronomy*, 12(2). <https://doi.org/10.3390/agronomy12020365>
- Darmono, D. W. (2008). *Agar Anggrek Rajin Berbunga*.
https://books.google.co.id/books?hl=id&lr=&id=DBi9gil14z4C&oi=fnd&pg=PA1&dq=anggrek&ots=ai6nCDx2G0&sig=pY6z4idNf-ZCTqZvqccit0jpa14&redir_esc=y#v=onepage&q=anggrek&f=false
- Dressler, R. L., & Dodson, C. H. (1960). Classification and Phylogeny in the Orchidaceae. *Annals of the Missouri Botanical Garden*, 47(1), 25.
<https://doi.org/10.2307/2394615>
- Feurer, M., Klein, A., Eggenberger, K., Springenberg, J., Blum, M., & Hutter, F. (2015). Efficient and Robust Automated Machine Learning. In C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, & R. Garnett (Eds.), *Advances in Neural Information Processing Systems* (Vol. 28). Curran Associates, Inc.
https://proceedings.neurips.cc/paper_files/paper/2015/file/11d0e6287202fced

- Gerald, C., & Lubis, C. (2020). Pendeteksian Dan Pengenalan Jenis Mobil Menggunakan Algoritma You Only Look Once Dan Convolutional Neural Network. *Jurnal Ilmu Komputer Dan Sistem Informasi*, 8(2), 197.
<https://doi.org/10.24912/jiksi.v8i2.11495>
- Harun, A., Mustakim, & Kharisma, O. B. (2023). Implementasi Deep Learning Menggunakan Metode You Only Look Once untuk Mendeteksi Rokok. *Jurnal Media Informatika Budidarma*, 7(1), 107–116.
- Hsu, C.-C., Chen, H.-H., & Chen, W.-H. (2018). Phalaenopsis. *Ornamental Crops*, 567–625.
- Jocher, G., Changyu, L., Hogan, A., 于力军 L. Y., Changyu98, Rai, P., & Sullivan, T. (2020). *ultralytics/yolov5: Initial Release*. Zenodo.
<https://doi.org/10.5281/zenodo.3908560>
- Keskar, N. S., Mudigere, D., Nocedal, J., Smelyanskiy, M., & Tang, P. T. P. (2017). On Large Batch Training for Deep Learning. *Iclr*, 1–16.
- Kohavi, R., & Provost, F. (1998). On Applied Research in Machine Learning. *In Editorial for the Special Issue on Applications of Machine Learning and the Knowledge Discovery Process*, 30, 271–274.
- LESTARI, D. A., & SANTOSO, W. (2010). Inventory and habitat study of orchids species in Lamedai Nature Reserve, Kolaka, Southeast Sulawesi. *Biodiversitas Journal of Biological Diversity*, 12(1).
<https://doi.org/10.13057/biodiv/d120106>
- Liunanda, C. N., Rostianingsih, S., & Purbowo, A. N. (2020). Implementasi Algoritma YOLO pada Aplikasi Pendeteksi Senjata Tajam di Android. *Jurnal Infra, Vol 8, No.*, 1–7.
- Mahfut, Wahyuningsih, S., Handayani, T. T., Sukimin, Umur, K., Sari, D. A., Angraeni, M., & Tohari, E. N. (2021). *Identifikasi Anggrek Alam di Kebun Raya Liwa*.

- Maulana, F. (2021). *Machine Learning Object Detection Tanaman Obat Secara Real-Time Menggunakan Metode Yolo (You Only Look Once)*.
- Mazumder, T. A., Student, M. S., Light, F., Networking, S., & Players, V. (2018). *Mobile Application and Its Global Impact 1. 06*, 72–78.
- Metusala, D. (2006). *Glancing at Vanda tricolor Orchid Conservation in Merapi* <http://www.angrek.org/melirik-konservasi-angrek>. Vanda- tricolor.
- O’Shea, K., & Nash, R. (2015). An introduction to convolutional neural networks. *ArXiv Preprint ArXiv:1511.08458*.
- Pamungkas, D. P. (2019). *Ekstraksi Citra menggunakan Metode GLCM dan KNN untuk Identifikasi Jenis Angrek (Orchidaceae)*.
- Phillip J., C., Shelagh P., K., Kingsley W., D., & Russell L., B. (2003). Orchid conservation: a global perspective. *Orchid Conservation, May 2014*, 1–24.
- Post, C. A. (2020). *Multilabel Classification of Orchid Features based on Deep Learning*. 1–10.
- Pramestya, R. H. (2018). Deteksi dan Klasifikasi Kerusakan Jalan Aspal Menggunakan Metode YOLO Berbasis Citra Digital. *Institut Teknolgi Sepuluh Nopember*, 91. http://repository.its.ac.id/59044/1/06111650010019-Master_Thesis.pdf
- Purnamasari, L., Yolanda, R., & Karno, R. (2011). Jenis-jenis Angrek Epifit (Orchidaceae) di Desa Koto Tinggi Kecamatan Rambah Kabupaten Rokan Hulu. *Jurnal Penelitian Pertanian Terapan*, 2(2), 150–161.
- Rahim, A., Kusriani, K., & Luthfi, E. T. (2020). Convolutional Neural Network untuk Kalasifikasi Penggunaan Masker. *Inspiration: Jurnal Teknologi Informasi Dan Komunikasi*, 10(2), 109. <https://doi.org/10.35585/inspir.v10i2.2569>
- Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (n.d.). *You Only Look Once: Unified, Real-Time Object Detection*.
- Swathiga, U. U. A. S., Vinodhini, P., & Sasikala, V. (2021). an Interpretation of

Dart Programming Language. *Dogo Rangsang Research Journal*, 11(10), 144–149.

Wang, Q., Rasmussen, C., & Song, C. (2016). Fast, deep detection and tracking of birds and nests. *Advances in Visual Computing: 12th International Symposium, ISVC 2016, Las Vegas, NV, USA, December 12-14, 2016, Proceedings, Part I 12*, 146–155.

Widiastoety, D. (2004). Bertanam Anggrek. *Penebar Swadaya. Jakarta*, 76.

Yao, J., Qi, J., Zhang, J., Shao, H., Yang, J., & Li, X. (2021). A real-time detection algorithm for kiwifruit defects based on yolov5. *Electronics (Switzerland)*, 10(14). <https://doi.org/10.3390/electronics10141711>

Zainuri, M., & Pamungkas, D. P. (2020). Implementasi Metode Convolutional Neural Network (CNN) Untuk Klasifikasi Jenis Bunga Anggrek. *Seminar Nasional Inovasi Teknologi*, 87–92.